

PROCEEDINGS OF THE
MULTIDISCIPLINARY DOCTORAL SCHOOL
2009-2010 ACADEMIC YEAR
FACULTY OF INFORMATION TECHNOLOGY
PÁZMÁNY PÉTER CATHOLIC UNIVERSITY
BUDAPEST
2010

Faculty of Information Technology
Pázmány Péter Catholic University

Ph.D. PROCEEDINGS

PROCEEDINGS OF THE
MULTIDISCIPLINARY DOCTORAL SCHOOL
2009-2010 ACADEMIC YEAR
FACULTY OF INFORMATION TECHNOLOGY
PÁZMÁNY PÉTER CATHOLIC UNIVERSITY
BUDAPEST

July, 2010



Pázmány University ePress
Budapest, 2010

© PPKE Információs Technológiai Kar, 2010

Kiadja a Pázmány Egyetem eKiadó
2010
Budapest

Felelős kiadó
Dr. Fodor György
a Pázmány Péter Katolikus Egyetem rektora

Cover image by András Kiss, Mach 3 flow around a cylinder after 1s of simulation time

A borítón Kiss András ábrája látható: Henger körüli szuperszónikus áramlás 1
másodperces szimulálási idő után

HU ISSN 1788-9197

Contents

INTRODUCTION	7
RÓBERT TIBOLD • The Effect of Load on Different Levels of Movement Controlling in Reaching Arm Movements	9
TAMÁS PILISSY • New approaches for improved Functional Electrical Stimulation methods	13
BALÁZS DOMBOVÁRI • In vivo validation of microelectrode arrays with electronic depth control for acute recordings	17
ÁDÁM BALOGH • Application of Phonocardiography on Preterm Neonates with Patent Ductus Arteriosius	21
DÁNIEL KOVÁCS • Theoretical and experimental study of a digital microfluidic chip	25
ANDREA KOVÁCS • Shape detection of structural changes in long time-span image samples by new saliency methods	31
VILMOS SZABÓ • Hierarchical Feature Extraction for Dynamic Feature and Signature Tracking	35
BALÁZS VARGA • GPGPU Accelerated Scene Segmentation Using Nonparametric Clustering	41
ANDRÁS GELENCSÉR • Distortion analysis and possible corrections of the pictures projected by a projector	45
KÁLMÁN TORNAI • Sequence data mining	49
TAMÁS FÜLÖP • Towards moving platform video processing – Object detection on many core architectures	53
MIHÁLY RADVÁNYI • Dynamic object detection in urban environment	57
ATTILA STUBENDEK • Towards recognition-driven semantic shape classification	61
MIKLÓS KOLLER • Wave Computational Abilities of Large Infrared Proximity Arrays	65
LÁSZLÓ LAKI • Investigating the possibilities of processing parallel resources with language statistical methods	69
FERENC OTT • Information-extraction from medical diagnosis and anamnesis with text-mining algorithms	73
ANDRÁS HORVÁTH • Fast computation of particle filters on topographic processor arrays	77
CSABA NEMES • Mapping Mathematical Expressions into FPGA Devices Using Data-Flow Graphs	81
ÁDÁM RÁK • Standard C++ Compiling to GPU with Lambda Functions	85
GÁBOR TORNAI • GPU boosted 2D and 3D level set algorithms	89
TAMÁS ZSEDOVITS • Collision avoidance for UAVs using visual detection	93
LÁSZLÓ FÜREDI • A Redesigned Emulated Digital CNN Architecture for FPGAs	97

ANDRÁS KISS • Emulated Digital Cellular Neural Networks for Accelerating CFD Simulations	101
ZOLTÁN KÁRÁSZ • Realizing large time constant in implantable neural signal recording application	105
LÁSZLÓ KOZÁK • Power Amplifier at Low Frequency with Low Output Power	109
DOMONKOS GERGELYI • Sensing in the terahertz frequency domain	115
BALÁZS KARLÓCAI • Energy optimization in wireless sensor networks using discrete energies	119
GERGELY TREPLÁN • Cooperative Communication Algorithms in Wireless Systems	123
ÁKOS TAR • Object Outline and Surface-Trace Detection Using Infrared Proximity Array	127
JÓZSEF VERES • Novel approach for control authority measurement of a five link planar biped underactuated by one	131

Introduction

It is our pleasure to publish this annual proceedings again to demonstrate the genuine multidisciplinary research done at our Jedlik Laboratories by the many talents working in our Interdisciplinary Doctoral School.. Thanks are also due to the supervisors and consultants, as well as to the five collaborating National Research Laboratories of the Hungarian Academy of Sciences and the Semmelweis Medical School. The collaborative work with the partner Universities, especially, Katolieke Universiteit Leuven, Politecnico di Torino, Technische Universitat in München, University of California at Berkeley, University of Notre Dame, Univetsidad Sevilla, Universita di Catania is gratefully acknowledged..

As an important development of this special collaboration, we were able to jointly complete the second year with the Semmelweis Medical School a new undergraduate curriculum on Molecular Bionics, the first of this kind in Europe.

We acknowledge the many sponsors of the research reported here. Namely,

- the Hungarian National Research Fund (OTKA),
- the Hungarian Academy of Sciences,
- the National Office of Research and Development (NKTH),
- the Gedeon Richter Co.,
- the Office of Naval Research (ONR) of the US,
- IBM Hungary,
- Eutecus Inc., Berkeley, CA,
- Morphologic Ltd., Budapest,
- Analogic Computers Ltd., Budapest,
- AnaFocus Ltd., Seville, and

some other companies and individuals.

Needless to say, the resources and support of the Pázmány University is gratefully acknowledged.

Budapest, July 2010.

TAMÁS ROSKA
Head of the Doctoral School

The Effect of Load on Different Levels of Movement Controlling in Reaching Arm Movements

Robert Tibold

(Supervisor: Dr. József Laczkó)

tibro@digitus.itk.ppke.hu

Abstract—A three-dimensional (3D) arm model to simulate kinematic properties and muscle forces in reaching arm movements is presented. Healthy subjects repetitively performed arm movements grasping an object and replacing it vertically (uplifting or putting down) from one shelf to another one under 2 load conditions: a) with a load; b) without load. 3D Joint coordinates were measured. Muscle moment arms, 3D angular acceleration, moment of inertias of arm segments were calculated to determine 3D joint torques. Variances of hand position, arm configuration and muscle activities were calculated. Ratios of mean variances across all subjects observed in the two conditions were studied. In kinematics there were no significant differences between the 2 conditions except in joint configuration during putting down. Virtual muscle force variances for flexors and electromyogram (EMG) variances for 4 muscles increased significantly by moving the load. The highly increased muscle activity variances didn't imply high increment on kinematic variances. Conclusion: enhancing of synergies helps stabilize the movement at kinematic level if a load is added.

Keywords: musculoskeletal, rehabilitation, motion analysis

I. INTRODUCTION

Special methods are required to control limb movements more properly [1] to improve the effect of medical rehabilitation techniques such as functional electrical stimulation (FES). Graphic based computer models have been developed to discern motor activity patterns of musculo-skeletal systems [2]. For tetraplegics and paraplegics some FES methods have been elaborated for controlling lower limb movements, e.g. cycling movements [3],[4],[5]. However, in the case of hemiplegics (stroke patients) and tetraplegics, we have lack of information whether FES method would be applied to stimulate muscles to move the entire upper extremities. To get the whole arm moved by an artificial control is rather complicated task partly because the complexity of the shoulder mechanism. Three dimensional (3D) inverse kinematic problem must be solved to get muscle forces needed to reach one selected point in the 3D space. FES Hand Grasp System was introduced in Cleveland F.E.S Center, [6] that has been used for controlling mostly the hand and fingers and rarely the forearm [7]. To define proper stimulation patterns, it is required to study the cooperation of muscle groups to be stimulated. There are 2 major objectives of the present study:

- 1) To investigate whether muscle synergies are enhanced in goal directed arm movements by analysis of variance

- 2) To present the computational model that establishes 3D muscle forces required to execute a planned movement.

II. METHODS

A. Experimental Methods

The subjects participated in the study; the applied experimental equipment setup are summarized in [8]. The motor task was executed under 2 conditions corresponding to 2 objects with different masses 1) a light CD case (0.06kg); 2) a load (2kg.). Uplifting (UP) and putting down (DOWN) were repeated ten times under each load condition. Further details on the measurements see [9]. Details on data processing of kinematic parameters (3D coordinates of anatomical landmarks) and EMG's are summarized in [10]

B. Simulation Methods

Input parameters of the model are 3D coordinates of previously measured anatomical landmarks [10]. The time courses of inter-segmental joint angles were computed from these coordinates. Further input parameters were arm segment masses; segment lengths estimated from the height and mass of the body according to [11]. Muscle forces were determined based on [9] when only one muscle was active at a time t :

$$F_m(t) * R_m(t) = \beta(t) * \Theta^{joint}(t) - T_g(t) \quad (1)$$

The calculation of the mechanical parameters of Eq.(1) is written in [9]. $R_m(t)$ (moment arm) was determined from 3D coordinates of muscle attachments based on the study performed on cadavers by Veeger [12]. Using [12] a virtual subject with virtual body heights, segment lengths and muscle attachments were created. However muscle attachments were not assumed as points on the given arm segment as written in [9] but as points on the bone surface of the given segment because:

- 1) muscle attachments are connected to the bone surface via tendons
- 2) Veeger et. al. [12] provided data on muscle insertion and origin areas only when the elbow was fully stretched.

To determine the exact 3D location of muscle attachments on each segment after calculating them at fully stretched elbow Rodrigues' rotation formula (Eq.(2)) was applied for the whole

interval of the movement. Rodrigues' method is a general tool for rotating a given vector in the 3D space ($v \in R^3$) about an arbitrary rotation axis (z is a unit vector of $z \in R^3$) by given angle of rotation (θ). In the case of the modeled movement either uplifting or putting down the vector to be rotated ($v \in R^3$) was a unit vector generated from muscle attachments given in [12]; the rotation axis ($z \in R^3$), about the rotation was done, was the vector that was perpendicular to the plane of rotation; while the rotation angle (θ) was computed from previously determined joint angles [8]. Rodrigues based 3D general rotation was applied for every measured trial for every instant.

$$v_{rot} = v \cos \theta + (z \times v) \sin \theta + zz^T v (1 - \cos \theta) \quad (2)$$

In 1938 Hill [13] proved that the force a muscle can exert is given by the sum of the active $F_a(l)$ and passive force $F_p(l)$. $F_a(l)$ is the force generated by the muscle compartment while $F_p(l)$ is resulted by connective tissues and tendons attaching the muscle to the bone surface. Active and passive force characteristics were determined separately as a function of normalized muscle length. (Figure 1.) Muscle length was computed by using 3D muscle attachments originated on bone surfaces of the actual segment. Active and passive characteristics of muscles were originated based on the study of Woittiez [14] and Yi-Wen Chang [15]. The unique characteristics of a muscle structure can be represented by the index of architecture (ia) defined in [14]. It is calculated as the ratio of a single muscle fiber length to muscle belly length. Muscle belly length is defined as the distance between the proximal and distal tendon of the selected muscle. Active and passive components

$$F_a(\varepsilon, ia) = e^{-\left(\frac{(\varepsilon + 1)(0.96343 * (1 - 1/ia)) - 1}{0.35327 * (1 - ia)}\right)^2} \quad (3)$$

$$F_p(\varepsilon, ia) = 0.0195 * e^{\left(2.933 + \frac{4.911}{ia}\right) * \varepsilon} \quad (4)$$

were computed according to Eq.(3) and Eq.(4) where F_a is the normalized muscle tension; F_p is the normalized passive force; while ε is the muscle strain computed as $(L - L_o)/L_o$. Here L is the muscle length at an instant of time and L_o is the optimum muscle length. L_o were set to be the half of the maximum muscle length. Index of architecture of all investigated muscles were adopted from the literature.(Table I.). BI index was determined from Chang [15]; TR index was applied based on [16]. DA and DP index of architectures were set to be equal according to De Wilde [17], because these two different parts of the deltoid muscle are pretty similar and strap-like. [18]

C. Calculating variances

It has been reported that variances of hand positions and arm configurations are strongly effected by neuromotor diseases

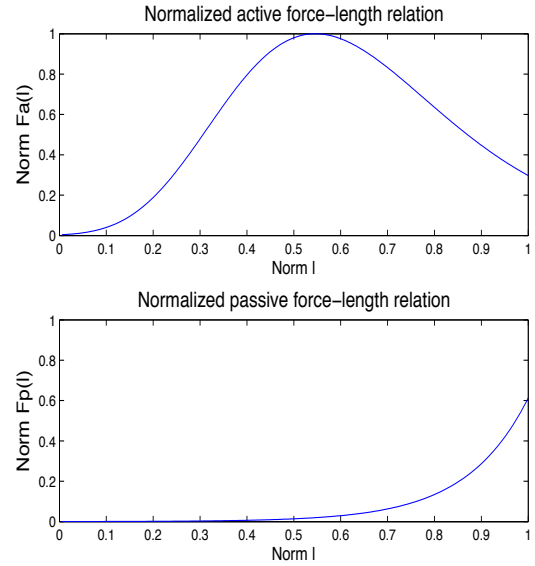


Fig. 1. Normalized active and passive force-length curves of BI. $F_a(l)$ and $F_p(l)$ were determined for all muscles using different index of architectures (Table I.) adopted from the literature.

TABLE I
INDEX OF ARCHITECTURE ADOPTED FROM THE LITERATURE

	BI	TR	DA	DP
ia	0.67	0.4	0.7	0.7

such as stroke and Parkinsons disease (PD) [19]. To show the effect of load on different movement controlling levels time normalized variances of ten repetitively executed trials were calculated for the two load conditions during UP and DOWN. Variances of endpoint (EP) positions; joint configuration (JC); EMG activities and muscle forces (FORCE) were computed for:

- 1) the whole time interval of the uplifting and putting down
- 2) for the interval while the object was in the hand (holding).

Hence, every trial was divided into 3 time intervals. The first part was the time from movement initiation to the instant when the hand reached the object(pre-holding). The second was the time interval in which the object was in the hand(holding). The third part started when the subject released the object and ended when the arm was replaced to the initial position(post-holding). In case 1) variances of ten trials at each percentage of total movement time were calculated. As a result, a variance vector was generated in function of normalized time for all subjects. These vectors were averaged across subjects and then assigned to the actual object condition. This mean variance was computed for all controlling levels. In case 2) the time interval of holding was determined [9]. Holding was started when the distance of the base of the little finger (marker number 7) and the object (marker number 8) was smaller than a threshold. The threshold was set as the minimal distance between the two markers (7-8) plus 25 mm to avoid any

measuring inaccuracy. [9] Holding ended when the distance was greater than the threshold after detecting a start of holding. Because the duration of holdings varied across trials a time normalization of the detected holding was performed in each trial. The variance of ten detected holding were computed as a function of normalized holding time. Variances were generated for EP (upper panel of Figure 2.); JC (lower panel of Figure 2.); EMG of the 4 arm muscles separately (Figure 3.); and for predicted muscle forces (Figure 4.). In the figures only the results of uplifting are presented.

D. Statistics

Two-way t-test according to the investigated two load conditions (*movements executed with load and without load*) was performed to study the effect of both conditions (with load, without load) at a 5% significance level ($p=.05$). Two way t-test's were performed separately for variances computed from ten trials for both uplifting and putting down for all of the investigated movement controlling levels. In the cases of muscle activity levels for both measured (EMG) and predicted (FORCE) ones t-test's were performed separately for all arm muscles. To avoid the typical error of multiple comparisons Bonferroni correction was performed although the effect of load on different movement controlling levels was investigated separately.

III. RESULTS

Statistical analysis of the whole movement interval didn't show significant difference between the load conditions either in measured or in predicted parameters.

In the analysis of holding ratios of variances were computed by dividing the mean variance of with load movements across all subjects by the mean variance of without load movements across all subject. Table II. presents the ratios of mean variances across subjects.

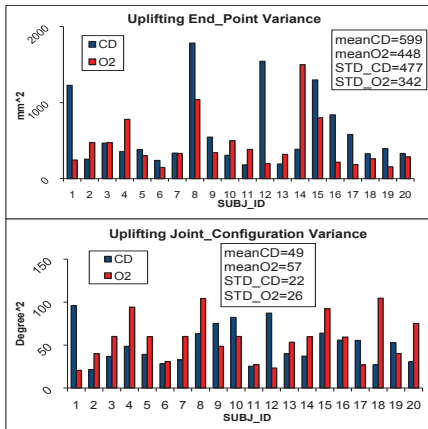


Fig. 2. Subject-by-subject variances; mean variance and standard deviation across all subjects for EP and JC in uplifting.

Holding analysis didn't show significant difference in EP variances (upper panel of Figure 2.) either in UP ($t=0.006$, $p=.05$) or DOWN ($t=1.48$, $p=.05$). The ratio of EP variances

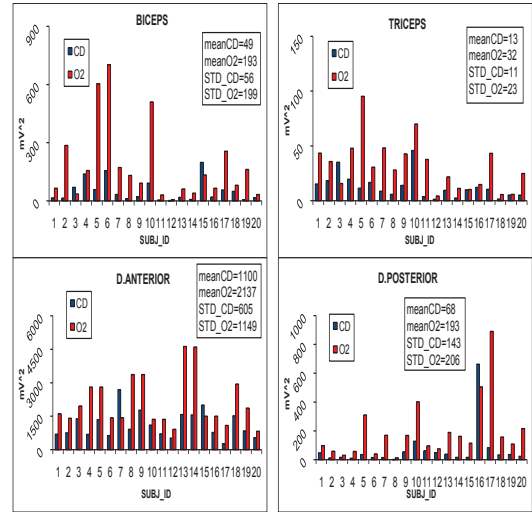


Fig. 3. Subject-by-subject variances; mean variance and standard deviation across all subjects for all muscle EMG's in uplifting.

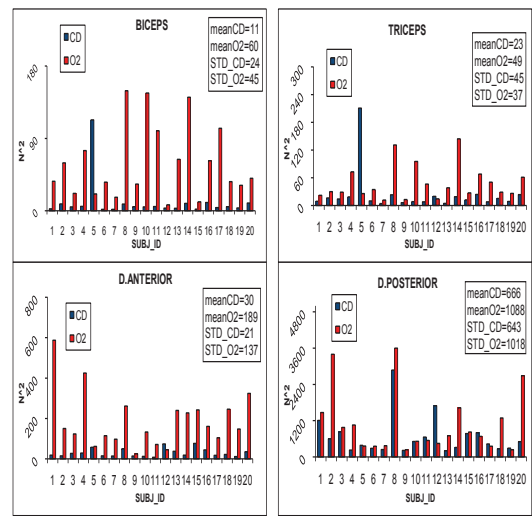


Fig. 4. Subject-by-subject variances; mean variance and standard deviation across all subjects for predicted muscle forces in uplifting.

was smaller than 1 only in UP. (Table II.). In JC (lower panel of Figure 2.) there was no significant difference in UP ($t=1.79$, $p=.05$) but in DOWN the difference was significant ($t=2.31$, $p=.05$). Ratios remained only in half of the subjects less than 1 resulting barely greater than 1 mean variance ratio in UP while in DOWN ratios were greater than 1 for all subjects resulting 1.4 mean variance ratio. This suggests that the load has a smaller effect on JC variability if the movement is executed against gravity. sEMG (Figure 3.) variances in both UP and DOWN showed significant differences for all muscles. Mean variance ratios were much higher than 1. Hence, the load increased muscle activity variances in both directions at a high rate. In computed muscle forces (Figure 4.) in flexors significant differences were observed in UP and DOWN as well. In extensors there were no significant differences in

TABLE II
RATIOS OF MEAN VARIANCES, GENERATED BY DIVIDING THE MEAN VARIANCE OF WITH LOAD MOVEMENTS ACROSS BY THE MEAN VARIANCE OF WITHOUT LOAD MOVEMENTS ACROSS ALL SUBJECTS

	UP		DOWN	
EP	0.7		1.3	
JC	1.1		1.4	
EMG	BI	TR	BI	TR
	3.6	2.5	4.79	3.05
	DA	DP	DA	DP
	1.9	2.8	2.24	3.43
FORCE	BI	TR	BI	TR
	5.9	2	8.02	2.96
	DA	DP	DA	DP
	6.3	1.6	4.85	1.19

UP but TR showed significant difference ($t=3.64$, $p=.05$) in DOWN. Ratios of mean variances were much higher than 1 similarly to the EMG variances concerning all muscles.

IV. DISCUSSION

Statistical analysis of variances for the whole movement interval didn't show any significant difference between the 2 object conditions. This may be because averaged variances in the pre and post-holding parts remained high and reached their top out at about 3 times higher than during holding in both UP and DOWN. This suggests that movement performed with an object varied less than movement performed without object. For holding either in simulated muscle force or in measured muscle activity the larger mass of the object was associated to increased variances in both flexors and extensors. In muscle force variances the range of increment for flexors is higher than for extensors suggesting that flexors generate greater force with higher variability than extensors if the mass of the object is increased. EMG variances however showed less range of increment with lower variability for extensors in the elbow but the opposite was observed for shoulder muscles. This can be explained by that subjects activate more shoulder flexors among which the force of our single virtual flexor is distributed. Based on the ratios (Table II.) we conclude that the load helps to stabilize the movement at kinematic level mostly through by the hand position and less by the joint configuration. Thus, peripheral patterns reflect central processes rather than being separately controlled components of the action. Such findings have been suggested for grip force adjustment [20]. Our finding aligns with the results that external conditions or practice effect joint configuration variances and endpoint variances at a different rate. Practice helps to stabilize hand position by using smaller range of available joint configurations [21]. Our results show that the redundant number of available biomechanical degrees of freedom in the arm movements was restricted by the load at such a way that enhanced joint synergies helped to stabilize hand position during holding. Hence, the relatively small kinematic variances are the result of enhanced muscle synergies rather than increased individual muscle activity. Otherwise the higher muscle activity variances would increase kinematic variances at the same rate.

REFERENCES

- [1] J. Laczko, K. Walton, and H. Watanabe, "Modeling of limb movements to compute and transfer stimulation trains to spinal motoneuron pools," in *12th annual Conference of the Intl.Fes Society*, 2007.
- [2] J. Laczko, A. Pellionisz, J. B. W. Peterson, and T. S. Buchanan, "Multidimensional sensorimotor "patterns" arising from a graphics-based tensorial model of the neck-motor system," *Soc. Neurosci. Abst.*, vol. 13, no. 1, p. 372, 1987.
- [3] J. Szecsi, M. Fiegel, S. Krafczyk, A. Straube, J. Quintern, and T. Brandt, "The electrical stimulation bicycle: a neuroprosthesis for the everyday use of paraplegic patients," *MMW Fortschr Med*, vol. 146, no. 26, pp. 37–38, 2004.
- [4] T. Pilissy, A. Klauber, G. Fazekas, J. Laczko, and J. Szecsi, "Improving functional electrical stimulation driven cycling by proper synchronization of the muscles," *Ideggyogy Sz*, vol. 61, no. 5-6, pp. 162–167, 2008.
- [5] J. Szecsi, C. Schlick, M. Schiller, W. Pollmann, N. Koenig, and A. Straube, "Functional electrical stimulation-assisted cycling of patients with multiple sclerosis: biomechanical and functional outcome—a pilot study," *J Rehabil Med*, vol. 41, no. 8, pp. 674–680, 2009.
- [6] C. F. Center, "http://fescenter.org/index.php," website.
- [7] R. L. Hart, K. L. Kilgore, and P. H. Peckham, "A comparison between control methods for implanted fes hand-grasp systems," *Trans Rehabil Eng*, vol. 6, no. 2, pp. 208–218, 1998.
- [8] J. Laczko, R. Tibold, and G. Fazekas, "Neuromuscular synergy ensures kinematic stability during 3d reaching arm movements with load," in *Program No. 272.2 2009 Neuroscience Meeting*. Soc. for Neuroscience, 2009.
- [9] R. Tibold and J. Laczko, "Non-linear 3d model of muscle forces and kinematic variances in reaching arm movements," *J. Appl. Biom.*, 2010, has been recommended for publication after revision.
- [10] R. Tibold, "Emg," in *Proceedings of the Multidisciplinary Doctoral School Faculty of Information Technology 2007-2008 Academic Year*, pp. 45–48.
- [11] Z. Vladimir, *Kinematics of Human Motion*. Champaign, IL: Human Kinetics, 2008.
- [12] H. E. Veeger, B. Yu, K. N. An, and R. H. Rozendal, "Parameters for modeling the upper extremity," *J Biomech*, vol. 30, no. 6, pp. 647–652, 1997.
- [13] A. V. Hill, "The heat of shortening and the dynamic constants of muscle," *Proc.R.Soc.Lond*, vol. 126, pp. 136–195, 1938.
- [14] R. D. Woititz, P. A. Huijing, H. B. K. Boom, and R. H. Rozendal, "A three dimensional muscle model: a quantified relation between form and function of skeletal muscle," *J Morphology*, pp. 182–195, 1984.
- [15] Y. W. Chang, F. C. Su, and H. W. Wu, "Optimum length of muscle contraction," *Clinical Biomechanics*, vol. 14, no. 8, pp. 537–542, 1999.
- [16] J. Friden and R. L. Lieber, "Quantitative evaluation of the posterior deltoid to triceps tendon transfer based on muscle architectural properties," *Journal of Hand Surgery-American Volume*, vol. 26, no. 1, pp. 147–155, 2001.
- [17] L. D. Wilde, E. Audenaert, and E. Barbaix, "Consequences of deltoid muscle elongation on deltoid muscle performance: a computerised study," *Clinical Biomechanics*, vol. 17, no. 7, pp. 499–505, 2002.
- [18] J. B. Wickham and J. M. M. Brown, "Muscles within muscles: the neuro-motor control of intramuscular segments," *Eur J Appl Physiol*, vol. 78, pp. 219–25, 1998.
- [19] Z. Keresztesy, P. Cesari, G. Fazekas, and J. Laczko, "The relation of hand and arm configuration variances while tracking geometric figures in parkinson's disease: aspects for rehabilitation," *Int J Rehabil Res*, vol. 32, no. 1, pp. 53–63, 2009.
- [20] J. P. Scholz and M. L. Latash, "A study of a bimanual synergy associated with holding an object," *Human Movement Science*, vol. 17, no. 6, pp. 753–779, 1998.
- [21] D. Domkin, J. Laczko, M. Djupsjobacka, S. Jaric, and M. L. Latash, "Joint angle variability in 3d bimanual pointing: uncontrolled manifold analysis," *Exp Brain Res*, vol. 163, no. 1, pp. 44–57, 2005.

New approaches for improved Functional Electrical Stimulation methods

Tamás Pilissy

(Supervisor: Dr. József Laczkó)

piltom@ieee.org

Abstract — In the first years of our Functional Electrical Stimulation (FES) cycling project we achieved good results by applying the average muscle activities of healthy people as stimulation patterns for FES-cycling of spinal cord injured (SCI) patients. If we want to define more scientific and personalized stimulation patterns, a reasonable possibility is to compute (or at least approximate) muscle forces of lower limb arising during cycling motion. In this way individual characteristics (such as gender, weight and body segment parameters) can be taken into account. The new stimulation patterns require more sophisticated hardware, but this is not a problem since the stimulator that has been developed in our university is capable to generate practically any stimulation pattern that we define. In the end of this study, results of a rapidly developing FES-cycling subject will be presented.

Index Terms — biomechanics, 3D lower limb model, muscle force, muscle stimulation

I. INTRODUCTION

Last year we made important steps toward the implementation of a biomechanical lower limb model. Starting with the kinematical part, we have defined muscle attachment sites and lengths of knee extensor and flexor muscles applying an algorithm (based on two studies by Brand [1] and Hoy [2]) to our kinematical database of healthy cycling movement patterns [3]. It must be emphasized that despite the fact that cycling movement is basically a planar movement, we have computed muscle attachment sites and each step of further algorithms in three dimensions (3D) because of several reasons:

- (1) the line of a muscle spanning a joint is possibly not located in the plane that is defined by the two segments connected in the actual joint.
- (2) the plane of the adjacent limb segments (the instantaneous plane of motion) is changing in time.
- (3) during the cycling motion each of the measured markers (on the pedal, ankle, knee and hip) defined a different and furthermore unparallel plane.

It is an important point that for the calculations we did not use EMG data, the input of our algorithms were solely the kinematic data with some anthropometric information that was also registered during the measurements. From the viewpoint of the current task, important anthropometric features were the total body mass, height and gender, from which we could approximate the mass of lower limb segments (thigh, shank and foot) for each subject based on a formula by Zatsiorsky [4].

After all needed kinematic parameters had been acquired we started to deal with the computation of 3D muscle forces.

II. FORCE ALGORITHM

A. Computation of torques arisen in the knee joint

Computation of muscle forces were based on the simple relationship between torque, force and moment arm ($T = r \times F$). First of all the total measured torque (T_t) was computed as the product of the three dimensional angular acceleration vector (AK_{acc}) of the knee joint and the combined moment of inertia of shank and foot (I_{SF}).

$$T_t = AK_{acc} I_{SF} \quad (1)$$

where AK_{acc} vector was defined by the second derivative of the angle of the knee joint multiplied by the unit vector that was perpendicular to the plane defined by the midline of the thigh and shank, while I_{SF} was the combined moment of inertia of the shank-foot complex rotating around the knee. In the approximation of moment of inertia, shank and foot were regarded as two cylinders. Inertia of the shank (I_s) while rotating around the knee was computed as follows:

$$I_s = \frac{1}{4} m_s r_s^2 + \frac{1}{3} m_s l_s^2 \quad (2)$$

where m_s , r_s and l_s are the mass, radius and length of the shank, respectively.

To get the inertia of foot (I_F) during the rotation about the knee joint, we had to compute the distance between the knee and the center of mass of the foot (3) to use the modified formula for the computation of inertia (4) according to the parallel axis theorem [5].

$$d_{FcmK} = \left\| \frac{A+P}{2} - K \right\| \quad (3)$$

where A , P and K are the three dimensional coordinates of ankle, pedal and knee.

$$I_F = \frac{1}{4} m_F r_F^2 + \frac{1}{12} m_F l_F^2 + m_F d_{FcmK}^2 \quad (4)$$

The ankle joint was free during cycling, therefore the distance between the knee and the center of mass of the foot varied during cycling. For this reason, combined inertia of shank and foot was recalculated in every position of the lower limb (actually in every millisecond). Based on Steiner's law [5], combined inertia of two segments rotating about the same axis is the sum of the distinct inertias.

$$I_{SF_K} = I_{S_K} + I_{F_K} \quad (5)$$

As it was mentioned above, Equation 1 yielded the total measured torque. Besides this, we had to take into account the effect of gravity (T_g) and pedal resistance (T_p).

In our case the gravitational torque arose from the gravitational force acting on the shank and foot. Hence we have computed the combined center of mass of the shank and foot (SF_{cm}) considering the mass of these segments as well, by weighting the center of mass of both segments by its mass.

$$SF_{cm} = \frac{m_F}{m_F+m_S} \frac{A+P}{2} + \frac{m_S}{m_F+m_S} \frac{A+K}{2} \quad (6)$$

After the definition of this point, the gravitational moment arm was computed as follows:

$$R_g = SF_{cm} - K \quad (7)$$

Since the gravitational force was also known, only one step was missing to acquire the gravitational torque:

$$T_g = R_g \times ((m_F + m_S) * 9.81 * (-Z)) \quad (8)$$

where Z was the base vector that pointed upwards. The computation of pedal-torque was more difficult because we could not measure the force required to drive the pedal with constant angular velocity. Thus it was assumed that 100N tangential force was required to overwhelm pedal-resistance and drive the pedal with constant angular velocity. This 100N was multiplied by a normalized vector that was perpendicular to the plane defined by the rotating pedal (X) and to the vector pointing from the axis of the pedal to the actual position of the pedal.

$$F_p = 100 * \frac{X \times (P - P_{Axis})}{\|X \times (P - P_{Axis})\|} \quad (9)$$

This force generated a torque (T_p) in the knee that was computed as the cross product of the moment arm vector (R_p) pointing from the pedal to the knee and F_p .

B. Definition of moment arms of the thigh muscles

At this point we had all the required torques that were needed to compute muscle forces, only the corresponding moment arms remained missing. This required the spatial coordinates of muscle attachment sites during the recorded cycling motion samples. Since it was not possible to measure these anatomical points with expensive imaging techniques (CT, MR), we searched for approximation methods in the literature. We have found two studies [1,2] that matched with our expectations and helped us to approximate muscle attachment sites. They had measured anatomic locations of muscle origins and insertions on cadavers, placed these points in specific coordinate systems (based on the endpoints of limb segments) and determined equations that can be used to approximate muscle attachment sites.

The line of muscles breaks at least once, at the joints, that we called pulley point. In our model it was approximated with the point that is on the bisector of the

inner angle of the knee joint at a distance of 45mm from the knee in the case of biceps femoris (knee flexor muscle). Pulley point of vasti (knee extensor muscle) was defined on the same line, but in the opposite direction at 35mm from the knee. In both cases, computation of moment arm was performed on the longer segment of the broken line; hence in this step only the endpoints of the thigh, muscle origin and pulley point were used (Figure 1). Actually we had to find the height vector of the pulley-origin-knee triangle which points from the knee and is perpendicular to the origin-pulley line.

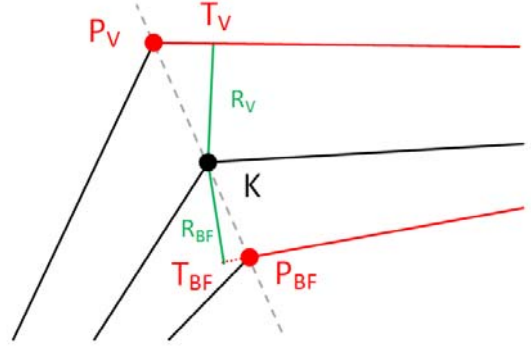


Fig. 1. Moment arm of the vasti (R_V) and biceps femoris (R_{BF}). P_V and P_{BF} were defined by the two segments connecting in the knee; hence they were in the plane of A, K and H. On the contrary, the line of muscles and thereby the moment arms were not in this plane.

For this purpose, we computed the cosine of P-O-K angle from the dot product of two unit vectors with OP and OK direction. Then, this was multiplied by the length of OK to get the length of OT, where T is the endpoint of the height vector of O-P-K triangle. Spatial coordinates of T were acquired by adding the product of the unit vector with OP direction and length of OT to spatial coordinates of O. As a final step, KT vector was defined which yielded the moment arm of the actual muscle. The algorithm was the same in the case of biceps femoris and vasti.

Two dimensional representation of this three dimensional problem can be seen on Figure 1. Note that muscle origins and insertions are not on the line that represents the thigh and shank.

C. Muscle force computation

Cycling is a very special movement, because the muscle work of left and right lower limbs are not independent. The problem here is that we measured only the left side of our subjects during the cycling. To overwhelm this issue, we introduced two new variables $T_1 = T_t - T_g - T_p$ and $T_2 = T_t - T_g$.

T_1 was the torque at knee that was generated by the vasti during cycling. T_2 was different because biceps femoris could not act against the pedal resistance (feet were not fastened to the pedals). T_1 and T_2 were three dimensional vectors and approximately orthogonal to the plane defined by the thigh and shank. To give appropriate directions for these computed torques, we defined muscle vectors (pointing from the origin to the pulley point) and

moment arm vectors (pointing from the knee to the nearest point of the line of muscle). We used the cross product of these two vectors of vasti according to the right hand rule to define the “positive direction”.

Most muscles can rotate a joint only in one direction, therefore in the case of vasti, negative torques were neglected while in the case of biceps femoris only negative torques were taken into account (i.e. were used for muscle force computation). After plotting T_1 and T_2 , it was clearly seen that both T_1 and T_2 changed direction twice during one full circle of cycling. Almost each of the three vector-component changed sign at the same millisecond. Since the first component was the most significant (with highest values), it was used to decide whether the torque is positive or negative.

We assumed that if the first component of T_1 was positive then the left vasti was active and its three dimensional force vector was computed from T_1 and R_v . If T_1 was less than zero, F_v was assumed to be zero.

As it was mentioned earlier, torque of biceps femoris was different, because our recumbent stationary bike did not allow pulling of the pedals. Therefore torque of biceps femoris was T_2 and when the first component of this torque vector was negative, force vector of biceps femoris was computed from T_2 and R_{BF} . If the above condition was not true we regarded F_{BF} as zero.

Muscle force was computed from torque and moment arm in two steps according to Tibold et al. [6]. The torque vector was divided by the length of the moment arm, which resulted in a vector that had the same direction as the torque vector, but with the length of the muscle force vector in question. Then, cross product of this and the moment arm vector was divided by the length of the moment arm, to acquire the final muscle force vector which was perpendicular to the torque and moment arm vectors.

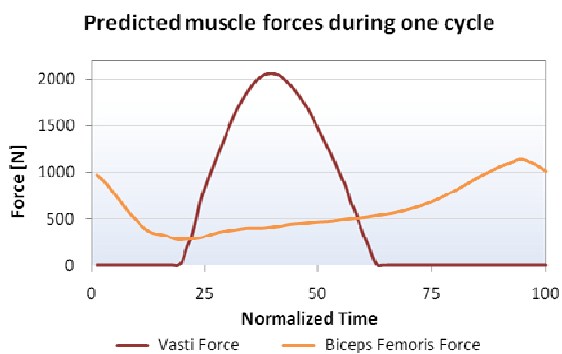


Fig. 1. Force of vasti and biceps femoris are presented as functions of normalized time.

Figure 2 shows an example of the computed muscle forces. As it can be seen, biceps femoris is active during the whole cycle, thus in a certain range, it works against its antagonist, but this is normal, because it has a regulatory effect on vasti. (The measured muscle activities (EMG) showed similar results.) As for the muscles of the right side, it was assumed that left and right muscles performed equal amount of work with a

half circle shift in pedal angle.

III. STIMULATOR DEVELOPMENT

The new stimulator patterns that we will generate from the computed muscle force will be much sophisticated than the ones our Motionstim8 stimulator can accept, but fortunately we develop a new stimulator. The first prototype had an aluminium case but later, when some parts was changed to smaller ones, the stimulator got a much more compact plastic case (Figure 3) with prepared place for six AA batteries. Operational time from batteries is about 3 hours. At the current stage, it can display a simple menu on the LCD, in which stimulation patterns can be selected and on-line parameters (such as maximal stimulating current) can be set by the knob and the channel-selector buttons. Reading of the stimulation patterns from the memory card and angular information from the pedal-angle encoder are also working. To help fast implementation of new stimulation patterns, I have written a program in MATLAB that can be used to generate the lookup tables (i.e. the main parts of the stimulation pattern) from some simple parameters.



Fig. 2. The stimulator that was developed by Attila Tihanyi and his students. This highly customizable stimulator provides improved functionality in a smaller case compared to the commercially available Motionstim8 stimulator.

IV. CHRONIC SCI AND FES-CYCLING PERFORMANCE: A CASE STUDY

During the first four years of the FES-cycling project we had more than a dozen spinal cord injured patient who attend this kind of training for at least several months [7]. Now I present here the performance increase of a subject who had attended FES-cycling only for a hundred days but reached outstanding result on the course of trainings. This subject was a chronic SCI patient, but it seems that it did not cause any disadvantage for him compared to fresh SCI patients. In a shorter period of time it is much easier to maintain the regularity of trainings thus, with a weekly average number of 1.5 trainings, he was the closest to the recommended two trainings per week. He was a fairly chronic SCI patient (with 44 post-injury months) and his injury was at C6.

During the hundred days of FES-cycling, he presented an unexpected development due to his outstanding

physical condition and frequent trainings (Figure 4). He managed to increase his average performance from the initial 5W above 20W in just three months. Linear regression line was fitted to the performance graph, which showed that average increase was 0.96Watt/week.

It seems possible that his spinal cord injury was not complete which could be a reason of his excellent condition.

Our opinion is that his good results was not related with the chronic nature of his SCI, since we had more chronic patients who did not show as good results at all.

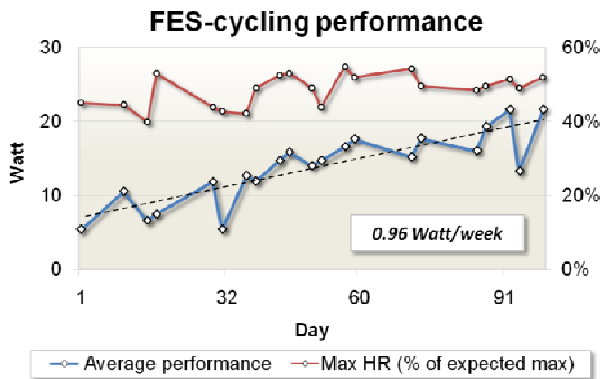


Fig. 3. Performance graph of our most rapidly developing SCI patient. Linear regression of the performance and heart rates as functions of expected maximum are also presented. Higher performance required only slight increase in heart rates.

Our results revealed that the success of trainings does not depend on the time passed since the spinal cord injury. It was interesting to see that the highest performance increase was achieved by a chronic SCI patient. This fact, with many other experiences with SCI patients during the course of FES trainings, has strengthened our notion about the successful feasibility of FES. First of all, muscles have to be in relatively good condition which is possible in the case of spastic muscles. However, it must also be known that high grade of spasticity can make the functional electrical stimulation process impossible since spasms ankylose the lower limb.

V. CONCLUSION

In our algorithm we managed to model geometry and force of lower limb muscles during cycling, based on simple kinematic measurements of four markers.

Earlier we believed that best cycling performance can be achieved by fresh SCI patients, but the results of our above introduced subject showed us that there are much more important parameters that can predetermine the success. The time passed since the injury has practically no affect if we speak about spastic SCI patients. As we saw, besides the level of injury, its completeness is also determinative.

VI. FURTHER PLANS

In the near future we would like to finish the force-computing algorithm by correcting its faults. As for the

stimulator, it requires only some minor development to be ready to exchange the currently used stimulator in the National Institute for Medical Rehabilitation. We have more plans to develop not only new stimulation patterns but further new hardware as well.

ACKNOWLEDGEMENT

We express our thanks to Attila Tihanyi, Dávid Kormos and György Nagy for building the stimulator, to Dr. András Klauber and Dr. Gábor Fazekas for providing the clinical guidance and environments, to the subjects for their participation and to Imréné Szanyi and Györgyi Stefanik for their help in the measurements. Our research is supported by a Grant of the Scientific Council of Healthcare No. ETT 363/2006.

REFERENCES

- [1] Brand R A, Crowninshield R D, Wittstock C E, Pederson D R, Clark C R, van Krieken F M: A model of lower extremity muscular anatomy, *J. Biomechanical Engineering*, Vol. 104, pp. 304-310, 1982
- [2] Hoy M G, Zajac F E, Gordon M E: A musculoskeletal model of the human lower extremity: the effect of muscle, tendon, and moment arm on the moment-angle relationship of musculotendon actuators at hip, knee, and ankle, *J. Biomechanics*, Vol. 23, No. 2, pp. 157-169, 1990
- [3] Katona P, Pilissy T, Fazekas G, Klauber A, Laczko J: A non-invasive method for the examination of muscle geometry to the exploration of the context of the muscle activities and muscle length changes. 4th Hungarian Conference on Biomechanics, CD attachment of the *Biomechanica Hungarica III./1.* pp. 101-105, 2010
- [4] Zatsiorsky W: *Kinematics of Human Motion*, Human Kinetics, Champaign, IL, 647-652, 1998
- [5] Marion JB, Thornton ST: *Classical Dynamics of Systems and Particles* (4th ed.), Thomson, ISBN 0-03-097302-3, 1995
- [6] Tibold R, Poka A, Borbely B, Laczko J: The effect of load on joint- and muscle synergies in reaching arm movements. 7th Conference on Progress in Motor Control, 2009, Marseille, France.
- [7] Pilissy T, Klauber A, Fazekas G, Laczko J, Szecsi J: Improving functional electrical stimulation driven cycling by proper synchronization of the muscles, *Ideggyogy Sz.* 61(5-6) pp. 162-167, 2008

In vivo validation of microelectrode arrays with electronic depth control for acute recordings

Balázs Dombóvári

(Supervisors: Dr. George Karmos and Dr. István Ulbert)
dombaga@digitus.itk.ppke.hu

Abstract—The NeuroProbes consortium were developed a new type of active CMOS based microprobe arrays. These devices implement an electronic depth control system coordinating a large number of electronically switchable recording sites which are separated by 40 μ m from each other. The systems enable the precise positioning of each contact with respect to individual neurons and in vivo measurements of local field potentials (LFP), single (SUA) and multiple unit activity (MUA). We evaluated the stability of the microprobe arrays with electronic depth control from insertion to recording. Single shaft CMOS probe was used with a length of 4 mm and 188 electrode contacts. The input and output lines of the probe were wire bonded to a printed circuit board (PCB). The electrode selection is sent to the probe via a hardware controller using the NeuroSelect software which provides a graphical user interface for managing all versions of NeuroProbes microarrays with electronic depth control. We successfully tested the performance of the recording sites and circuitry in saline solution. After the in vitro testing, the CMOS probes were inserted in the primary motor cortex and S1 trunk region of Wistar rats under ketamine/xylazine anesthesia. The probe assembled on the PCB was fixed to a micro manipulator which was used for the insertion through the intact dura. First successful in vivo experiments have shown that electronic depth control is an advanced technology for neuroscientists to either find specific neuron locations initially after probe insertion or to track unit activity during long- term recording. Each contact of the probe was scanned by switching electronically between the recording sites. LFP, MUA and SUA were collected from the cortex indicating appropriate functioning of the device. Besides good quality LFP and MUA, well separable SUA was also recorded.

Index Terms—electronic depth control; neural recording; silicon microprobes

INTRODUCTION

The European Project NeuroProbes aims to achieve a system platform for the scientific understanding of cerebral systems [5]. Single and multiple shank silicon- based NeuroProbes microprobes are developed and combined with chemical sensors [1], [6], [4].

Neural recordings with high spatial resolution are required for a basic understanding of neural processes. Currently this goal can only be achieved using silicon based MEMS arrays with slender probe shafts comprising multiple electrodes. Despite recent advances in the development of these systems, current microelectrode arrays for intracortical applications comprise a comparably small number of electrodes per shaft only. To overcome the restrictions of existing systems, i.e. the limited number of electrodes on MEMS-based probes and the desirable position control of the recording sites, a novel system concept has been presented, which is called electronic depth control [1], [2], [7]. The integrated CMOS circuitry on the probe shaft

reduces the number of connecting lines and allows to select a subset of recording sites from an unequaled number of electrodes [2], [7].

In the case of multielectrode arrays with a large number of recording channels the task of finding high quality neural signals with the aid of oscilloscopes and loudspeakers is tedious and rather impracticable. Despite, (semi-) automatic selection is required that aims to identify the best recording channels out of a set of electrodes with a special software. The NeuroSelect software was developed to control the CMOS-based multielectrode arrays, which has been presented in [3].

In this paper our goal was to test the stability and the recording capability of the microprobe arrays with electronic depth control in vivo from insertion to recording through the cortical slow wave activity.

MATERIALS AND METHODS

A. CMOS-based Neural Probe Arrays

The 100 μ m thick and 4 mm long active single probe shafts were prepared using deep reactive ion etching of silicon. Two columns of 92 electrodes with a pitch of 40 μ m and four electrodes in the tip region are distributed along the probe shaft resulting in a total number of 188 recording sites. The first CMOS-based neural probe shaft provides seven, the remaining two eight analog output channels. The probes are configurable in any combination of two tetrodes (2 x 2 electrodes) or certain combinations of eight single electrodes. The input and output lines of the probe were wire bonded to a printed circuit board (PCB) and encapsulated with EPO-TEK. The fabrication process of these probes is detailed in [2].

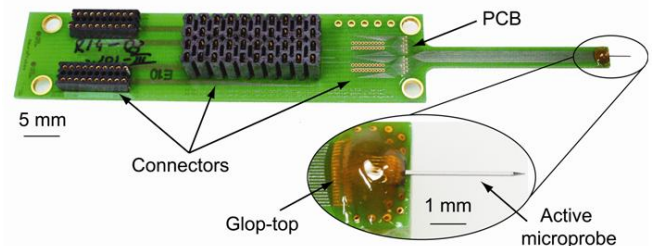


Fig. 1. Photo of active probe shaft with a length of 4 mm

B. Implantation procedure

The stability of NeuroProbes single shaft CMOS probes with a length of 4 mm and 188 electrode contacts were tested in vivo, in the neocortex of Wistar rats (n=5). Prior implantation electrode selection was checked using oscilloscope.

Impedance measurement: The impedances of the electrodes were measured before and during acute

implantation. A two electrode setup was used in vitro in physiological saline solution (0.9% NaCl) in which the working electrode was connected to the Pt electrode and the counter electrode was short-connected to the reference (stainless steel). The same measurement was also performed in vivo with the electrodes positioned in the cortex of the rat. The impedance was measured at 1 kHz. In all cases the impedances were measured between 0.5- 1 M Ω in on state and bigger than 2 M Ω in off state. For further details of in vitro impedance measurement of this type of neural probes, see [2].

Surgery: After every in vitro test, 3 CMOS probes were inserted in the neocortices of rats, one was implanted in the primary motor cortex (2 mm in lateral direction, aiming M1/M2) and 2 were implanted in the S1 trunk region. Surgeries were performed under ketamine/xylazine anesthesia (ketamine: 75 mg/kg, xylazine: 5mg/kg) and consisted of making independent craniotomies for the 3 probes in each rat, one probe was successfully used 3 times in different experiments. The probe assembled on the PCB was fixed to a micro manipulator which was used for the insertion through the intact dura with no significant dimpling. The data was preamplified (g=10 gain, bandpass filtered between DC and 100 kHz) and amplified (g=100 gain, bandpass filtered between 0.1 Hz and 5 kHz) with a total gain of 1000.

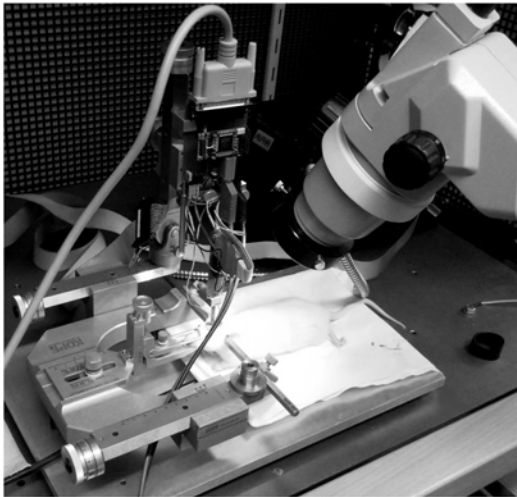


Fig. 2. In vivo experimental setup on a rat

The electrode selection is sent to the probe via a hardware controller (Xilinx Spartan XC3S200) using the NeuroSelect software which provides a graphical user interface for managing all versions of NeuroProbes microarrays with electronic depth control [3].

Signals are digitized with an A/D card at 16-bit resolution and 20 kHz sampling rate per channel (National Instruments PCIe 6259, voltage range ± 10 V).

RESULTS

The recorded data was analyzed off-line in order to test the signal quality. We used DataView and KlustaWin [11] software to separate SUA and custom-made Matlab software for Hilbert transformation, time-frequency analysis and to measure the signal-to-noise ratio (SNR). According to the expectation, on those channels, where the SNR value was higher, the possibility to find and separate SUA was also higher.

In deep sleep and anesthesia, cortical neurons oscillate between two states [9]. One is when the neuron's membrane is in hyperpolarized state (down-state) and the other is when it is in depolarized state (up-state). Under up state most of the neurons are firing, followed by a firing silence through down-state. Therefore we analyzed the relationship between LFP and SUA in down- and up-states with perievent time histogram (PETH) to confirm additionally the appropriate functioning of the probe. We used time- frequency maps to verify each separated states.

Under each experiment, we recorded 32 to 59 sessions. All electrodes were switched with the manual selection option of NeuroSelect software and LFP, MUA and SUA activity were collected from the cortex. Our first experience was the prime quality of MUA signal after bandpass filtering between 0.5-5 kHz in different layers of cortex (Fig. 3).

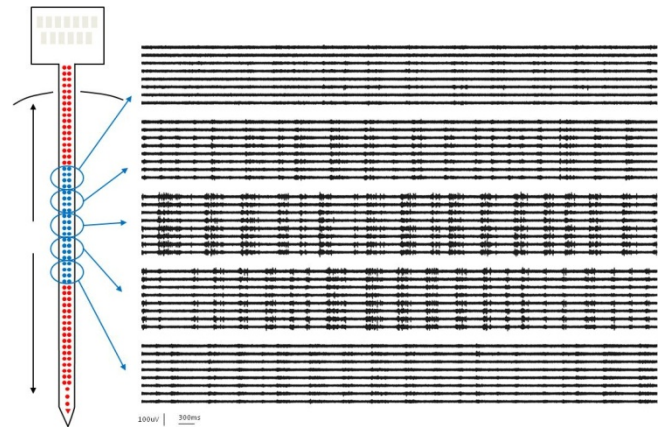


Fig. 3. MUA in cortex. Five different recording sessions following each other from distinct electrode configurations. The various MUA patterns are clearly visible in different layers of the cortex. The original raw data was bandpass filtered between 500-5000 Hz.

A. State detection

In the first step our aim was to retrieve information about the phase of LFP. In addition to spectral analyses, slow wave activity cycle detection was based on phase and amplitude information, extracted from the narrow band filtered (0.3-3 Hz, 24 dB/octave, zero phase shift) LFP data. Instantaneous phase of the filtered signal was calculated by the Hilbert transformation. In our implementation, a single slow wave activity cycle was defined between -180° and $+180^\circ$ phase. The -180° phase value corresponded to the trough of the negative half wave (up state) preceding the 0° phase, which corresponded to the peak of the positive half wave (down state) and finally the $+180^\circ$ phase corresponded to the following negative half wave trough (up state). Waves with non-monotonic phase runs were excluded. For more details, see [8].

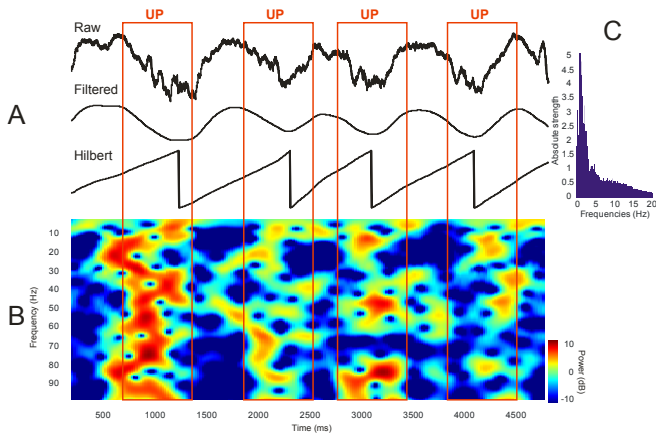


Fig. 4. State detection with Hilbert transformation. (A) Upper: ‘raw’ traces of single sweeps containing slow wave activity; broadband (0.1-300 Hz) LFP. Middle: ‘filtered’ traces after band pass (0.3-3 Hz) filtering. Lower: ‘Hilbert’ traces showing the phases of ‘filtered’ trace above derived by the Hilbert transform. Red rectangles indicate up states (deep negative half waves) (B) Color map of LFP spectral power during the slow wave activity shown in (A). (C) FFT frequency power of (A).

B. Time-frequency analysis

From the raw data and the results of valid Hilbert phase cycles, the single sweep and average time- frequency content of the slow wave activity signal was also computed using sinusoidal wavelet transforms to confirm in which state the cortex was. To compute a baseline normalized sinusoidal wavelet transform, we used the modified ERSP computing function of EEGLAB software [10]. When the cortex is in down- state, the time- frequency spectrogram show a significant decrease in power, in up- state a significant increase according to the selected state (Fig. 4 & 7).

C. Single unit activity analysis

All recordings yielded good quality single unit activity. Single units were separated from the multiple unit activity range (300-5000 Hz) data. The signal was further filtered (500-5000 Hz, zero phase shift, 48 dB/octave) and decimated at 2 kHz, applying a 0.5 ms sliding average rectangular window.

Putative single units were analyzed by conventional threshold detection and clustering methods using Dataview and Klustawin [11] and custom made Matlab software. After threshold recognition (mean $\pm 3-5$ SD) [12] at a given channel three representative amplitude values were assigned to each unclustered spike waveform. These triplets were projected into 3D space and a competitive expectation-maximization based algorithm [13] was used for cluster cutting [11]. If the autocorrelogram of the resulting clusters contained spikes within the 2 ms refractory interval, it was reclustered. If reclustering did not yield a clean refractory period, the cell was regarded as multiple units and omitted from the single cell analysis (Fig. 5).

We found on the average 4 to 7 spikes in each experiment through all electrode selection. To prove the real advance of this type of probes, if once a nice SUA was found, we reconfigured the electrode to the original configuration with a certain elapsed time. In all cases we got back the original single unit activities on same channels, which are also showing the stability of this type of probe.

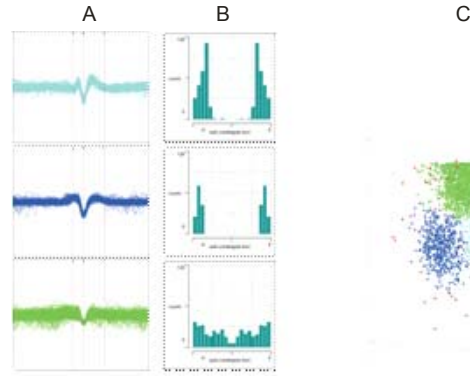


Fig. 5. Spike sorting. (A) Two different spikes (light and dark blue) and MUA error cluster (green) sorted from one electrode of probe. (B) Corresponding autocorrelograms. No firing found within the 2 ms refractory period in the case of blue colored spikes. (C) Three dimensional view of clusters. Red dots: error cluster.

D. SNR for spike quality estimation

The SNR was computed off-line as the relative power of the detected spikes compared with the background noise (1). The SNR value was calculated in Matlab with an algorithm similar to which was integrated in NeuroSelect software [3]. The SNR value was calculated as:

$$SNR_{dB} = 20 \times \log_{10} \frac{\frac{1}{N} \sum_{k=1}^N RMS(spike_k(t))}{RMS(noise)} \quad (1)$$

For threshold detection we used only the thresholding basic unsupervised detection algorithm [3]. As we found in SUA analysis, if we recognized a spike activity and later on we switched back to the same configuration, the SNR values were very similar (Fig. 6).

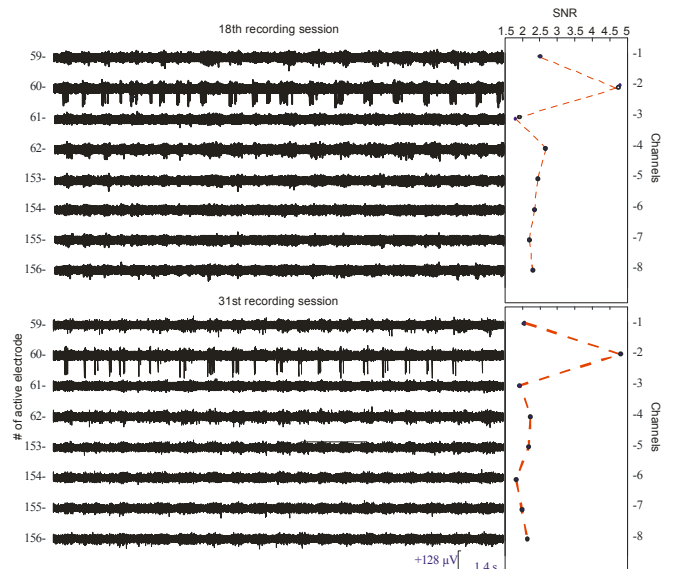


Fig. 6. Two different recording sessions with the same electrode configuration and corresponding SNR values (right side). Note: approximately 1 hour elapsed between the two sessions.

E. PETH

After phase of slow wave activity and spike detection, we further analyzed the relationship between up and down states of cortex in LFP and simultaneously recorded spike activity. PETH was calculated as single unit activity with

reference to the middle of up states. It is clearly visible, that most of the cells are firing under cortical up states, according to other human and animal studies [14] (Fig. 7).

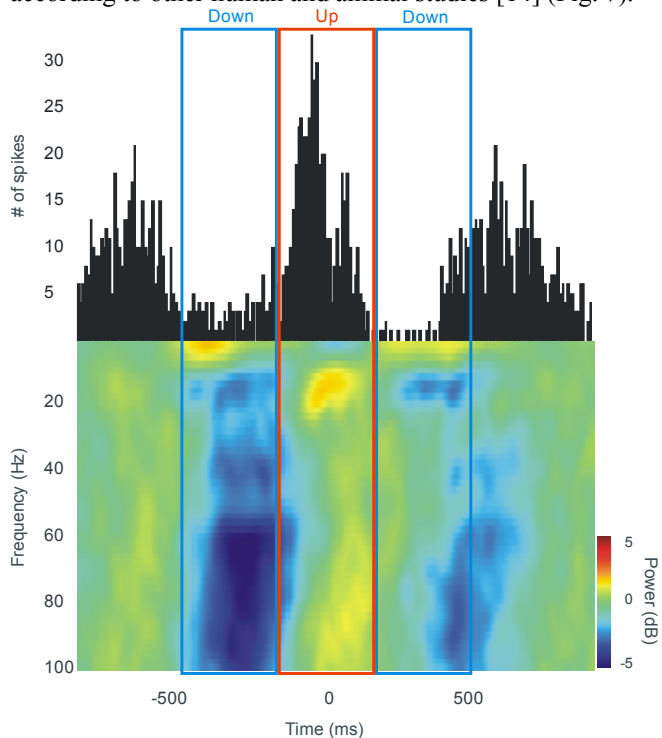


Fig. 7. PETH and time-frequency map of a channel. Upper: the relation of each spike to up state of slow wave activity. Rectangles indicate down (blue) and up (red) states. Lower: broadband spectral activity during slow oscillations: up state locked, averaged, relative spectrogram of LFP.

Nearly all cells showed non-uniform spiking over the slow wave activity cycle. The increased firing activity under the up state phase of slow wave activity is clearly visible in Figure 8.

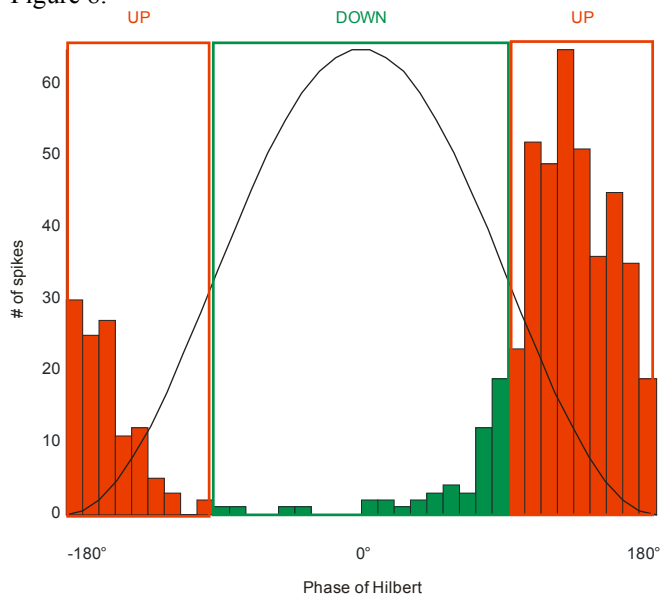


Fig. 8. Single unit firing in slow wave activity. Firing rate versus phase histogram (from -180 to 180, in 10 bins) of clustered cell. Red columns: cells firing under up states. Green columns: cells firing under down states.

CONCLUSIONS

The 4 mm long single shaft probe arrays with electronic depth control exhibit good electrode impedances for recording, both in physiological solution and in vivo. First successful in vivo experiments have shown that electronic

depth control is an advanced technology for neuroscientists to either find specific neuron locations initially after probe insertion or to track unit activity during long-term recording. Here we have shown the capability of probe arrays to record good quality LFP, MUA and SUA activities in different cortical regions of rats via cortical slow oscillation. Therefore this new device significantly increases the amount of information that can be obtained from a single experiment as compared to passive multi-electrode arrays.

ACKNOWLEDGMENT

This work was performed within the framework of the Information Society Technologies (IST) Integrated Project NeuroProbes of the 6th Framework Program (FP6) of the European Commission.

REFERENCES

- [1] H.P. Neves, T. Torfs, R.F. Yazicioglu, et al. The NeuroProbes project: a concept for electronic depth control. In: 30th International IEEE EMBS Conference, Vancouver, Canada, August 20–24, 2008. Piscataway, NJ: IEEE 2008: 1857.
- [2] K. Seidl, S. Herwik, Y. Nurcahyo, et al., “CMOS-based high density silicon microprobe array for electronic depth control on neural recording”, In: Proceedings of the International MEMS Conference, Pisa, Italy, January 25–29, 2009. Piscataway, NJ: IEEE 2009: 232–235.
- [3] K. Seidl, T. Torfs, P. A. De Mazière, G. V. Dijck, R. Csercsa, B. Dombovári et al., Control and data acquisition software for high-density CMOS-based microprobe arrays implementing electronic depth control. In: Biomedical Engineering. Volume 55, Issue 3, Pages 183–191, 2010
- [4] Aarts, A. A. A.; Neves, H. P.; Ulbert, I.; Wittner, L.; Grand, L.; Fontes, M. B. A.; Herwik, S.; Kisban, S.; Paul, O.; Ruther, P.; Puers, R. P.; Van Hoof, C.; “A 3D slim-base probe array for in vivo recorded neuron activity”, Proceedings of the Engineering in Medicine and Biology Society, 2008, EMBS 2008, 30th Annual International Conference of the IEEE, Page(s): 5798 – 5801
- [5] The NeuroProbes project: <http://www.neuroprobes.org/>
- [6] P. Ruther, A. Aarts, O. Frey, S. Herwik, S. Kisban, K. Seidl et al. The NeuroProbes Project: Multifunctional probe arrays for neural recordings and stimulation. In: Proc. Annual Conference of the IFESS; 2008, 238-240.
- [7] T. Torfs, A. Aarts, M. A. Erismis, C. V. Hoof, H. P. Neves, I Ulbert, B. Dombovári et al. Multi-channel neural probes with electronic depth control. In: IEEE BioCAS Conference, 2010, submitted for publication
- [8] R. Csercsa, B. Dombovári, D. Fabó, L. Wittner, L. Eröss, L. Entz et al. Laminar analysis of slow wave activity in humans. In: Brain, 2010, to be published
- [9] M. Steriade, A. Nunez, F. Amzica, A novel slow (< 1 Hz) oscillation of neocortical neurons in vivo: depolarizing and hyperpolarizing components. In: J. Neurosci 1993b, 13: 3252- 3265.
- [10] A. Delorme and S. Makeig, EEGLAB: an open source toolbox for analysis of single trial EEG dynamics including independent component analysis. In: J Neurosci Methods, 2004, 134: 9-21.
- [11] J. W. Heitler, Dataview v5: software for the display and analysis of digital signals in neurophysiology. <http://www.st-andrews.ac.uk/~wjh/dataview/2006>
- [12] J. Csicsvari, H. Hirase, A. Czurko, G. Buzsaki, Reliability and state dependence of pyramidal cell-interneuron synapses in the hippocampus: an ensemble approach in the behaving rat. In: Neuron, 1998, 21: 179-189
- [13] K. D. Harris, D.A. Henze, J. Csicsvari, H. Hirase, G. Buzsaki, Accuracy of tetrode spike separation as determined by simultaneous intracellular and extracellular measurements. In: J. Neurophysiol 2000, 84: 401- 414.
- [14] S. Sakata and K. D. Harris, Laminar structure of spontaneous and sensory evoked population activity in auditory cortex. In: Neuron 2009, 64: 404-418.

Application of Phonocardiography on Preterm Neonates with Patent Ductus Arteriosus

Ádám T. Balogh,

(Supervisors: Dr. Ferenc Kovács and Dr. Tamás Roska)

baladta@itk.ppke.hu

Abstract—The patent ductus arteriosus (PDA) is a prevalent disease in preterm newborns. The assessment of some parameters of the PDA with simple and noninvasive measurement equipment, such as phonocardiography (PCG), would be of great significance. In this paper the assessment of the closure of the PDA is investigated by estimating the splitting interval of the second heart sound (S2). For this, a heuristic method was applied and verified on a model, which was derived from the measured data. Application on clinical data showed promising results. The estimated splitting time increased by 30 % around the time of the closure of the PDA in case of pharmacologically treated preterm neonates.

Index Terms—biomedical signal processing, phonocardiography, patent ductus arteriosus, preterm infants, S2 split

I. INTRODUCTION

While during pregnancy the ductus arteriosus is an essential fetal vascular structure, after birth it should close because it loses its purpose, moreover, the persistence of the ductal patency is abnormal. In fetuses, the ductus arteriosus connects the main pulmonary artery with the descending aorta and shunts the blood coming from the right ventricle into the aorta due to the high resistance of the pulmonary circulation. Closure during pregnancy may lead to right heart failure. In case of normal neonates, with the first intake of breath the lungs expand and the resistance of the pulmonary circulation decreases greatly allowing the development of the normal human circulation. Under normal conditions functional complete closure occurs within the first day after birth [1].

If the ductus arteriosus remains open after birth a left-to-right shunt evolves due to the higher pressure in the aorta. This means an increased pulmonary fluid volume which may cause respiratory problems. Also the left atrium and ventricle have to compensate the increased fluid volume returning from the lungs and the "pressure leakage" in the aorta which may cause hypertrophy of the left atrium and ventricle, but the physiological impact and clinical significance of a PDA depends above all on its size and the state of the underlying cardiovascular system.

Some symptoms for physical examination are murmur, located at the upper left sternal border, overactive precordium, tachycardia and bounding peripheral pulses due to the rapid decrease of the diastolic pressure through the ductus. That means that there is a greater difference between the systolic and diastolic blood pressure (e.g. in case of neonates 2:1 ratio instead of 3:2).

The closure of the PDA may occur spontaneously or due to a surgical or transcatheter intervention. In case of preterm infants pharmacological closure is also possible.

In case of preterm neonates the risk of PDA is clearly much greater which is due to physiological factors related to prematurity [2]. Because the assessment of hemodynamical significance and the decision about the treatment is not obvious [3], in a recent work the usefulness of phonocardiography in assessing the hemodynamics of the PDA was investigated [4]. In that pilot study several parameters were found which could be related to the state and the closure of the PDA. One of them, which was found in case of several infants, was the splitting of the second heart sound.

Normally, the heart sounds are made up of the closure sounds of the valves on the left and right side of the heart. In case of the S2 sound these valves are called the aortic and the pulmonary valves. The pressure ratios between the arteries and the ventricles determine the exact closure time of these valves. For example in case of pulmonary hypertension, when there is an increased blood pressure in the pulmonary artery, the closure of the pulmonary valve is delayed with respect to the closure of the aortic valve, causing a wide splitting of the S2 sound.

Obviously, the presence of the PDA will have an influence on the pressure ratios between the arteries and the ventricles, thus also on the second heart sound. During the closure of the PDA the pressure rates will apparently change, which could be reflected in the time interval between aortic (A_2) and pulmonary (P_2) components of the second heart sound. In this study this was investigated.

This study is an extension of the research on fetal phonocardiography performed at the Pázmány Péter Catholic University, Faculty of Information Technology, Budapest [5].

II. MATERIALS AND METHODS

A. Measurements

In this study 25 preterm newborns have been examined, with an average of 3 measurements per infant, but with large deviations: only those newborns were examined several times which were diagnosed with PDA, those without PDA or with other cardiac malformation only once. Preterms without PDA were measured as a control group. Hemodynamically significant PDA was verified by echocardiography in case of 15 infants but only 8 of those were examined over several days because the others had either also some other malformation

or some other circumstances made further measurements not possible. The diagnostic parameters of the PDA acquired with echocardiography (diameter of the ductus, maximal velocity through the ductus, the left atrial to aortic root ratio) were all collected for later comparison with phonocardiographic parameters. In case of the 8 newborns mentioned above, the PDA was closed by means of pharmacological treatment (4 infants) or surgical intervention (4 infants).

These infants, except one, all weighed less than 2300 g at birth, with an average weight of 1400 g. Except one, all of them were less than 33 weeks of gestation, with an average of 29. They were examined on average on their 6th day after birth and those with PDA then every day until the closure of the PDA, which was verified by echocardiography (the maximum was 9 measurements on one infant). Three measurements had to be posteriorly excluded from the study because of the poor quality of the records since the measuring equipment was also developed during the study.

The measurements were made with a self-made electronic stethoscope (an electret microphone capsule, connected to a laptop, was joined together with a stethoscope head for infants). Each measurement consisted of about three 30 seconds long phonocardiographic records which were recorded at 48 kHz, with a resolution of 16 bits. According to our observation the main components of the heart sounds lie in the low frequency range, thus after prefiltering the data was resampled at 3000 Hz and only the useful part of the record (at least 10 secs) was kept for further analysis.

It should be noted that the clinical environment introduces a lot of noise in the records. Many of the disturbances can be filtered out by simple bandpass filtering. The most serious problem is the noise coming from the breathing machine because this noise lies in the same frequency bands than the heart sounds. Unfortunately practically all of the preterms need breathing aid.

B. Analyzing methods

Although there have been several methods introduced for the estimation of splitting interval [6]–[8] this problem is not solved for very short splitting intervals (SI) and for noisy signals like in case of this study. In this paper we suggest a heuristic, derivative based method for the assessment of SI.

Because the estimated SI cannot be verified in case of most of the records, a model-based validation was performed. In case of some of the measurements, the splitting of the S2 sound could be assessed by visual inspection and the components of the S2 sound could be separated in the time domain (Fig. 1). These records were selected and the method similar to [6] was applied for obtaining a S2 sound model valid for preterm neonates.

This analysis revealed that in case of this study a slightly different model should be used than in [6] which might be explained by the fact that here preterm neonates have been examined. We found that the instantaneous frequency ($IF[t]$) should be an exponentially decreasing function for both the aortic and the pulmonary component in case of preterm infants. For the instantaneous amplitude ($A[t]$) of both of the

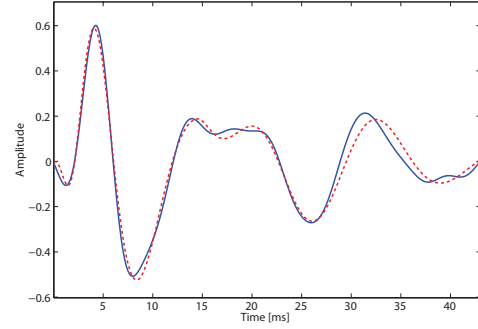


Fig. 1. Original S2 sound (blue) and the synthesized one (red). NRMSE is 15.86 %

components we used the same function as in [6] with two small modifications, namely the $k1$ and $k2$ exponents over t :

$$IF[t] = F_1 e^{-\frac{t}{\tau_1}} + F_2 \quad (1)$$

$$A[t] = a \left(1 - e^{-\frac{tk1}{\tau_2}}\right) \sin\left(\frac{\pi t}{\tau_3}\right) e^{-\frac{tk2}{\tau_4}} \quad (2)$$

where F_1 , F_2 are normalized frequencies, and τ_1 , τ_2 , τ_3 , τ_4 , a , $k1$ and $k2$ are free parameters.

The final parametric signal model is:

$$S_2[t] = A_A[t] \sin(\varphi_A[t]) + A_P[t - t_s] \sin(\varphi_P[t - t_s]) \quad (3)$$

where $A_A[t]$, $\varphi_A[t]$ and $A_P[t]$, $\varphi_P[t]$ are the envelope and the phase functions of the aortic and the pulmonary component of the second heart sound, respectively, and t_s is the SI.

The calculation of the phase functions was achieved in the following manner:

$$\begin{aligned} \varphi_A[t] &= \sum_{\tau=0}^t 2\pi IF_A[\tau] \\ \varphi_P[t] &= \sum_{\tau=0}^t 2\pi IF_P[\tau] \end{aligned} \quad (4)$$

where $IF_A[\tau]$ and $IF_P[\tau]$ are the instantaneous frequency functions of the aortic and the pulmonary components, respectively.

When the SI is short, the A_2 and P_2 components overlap greatly also in the time-frequency domain, thus separation becomes very difficult. We tried also some bound constrained global optimization with the model described above, unfortunately with no promising results, due to the great degree of freedom in the model.

Motivated by the fact, that at the start of the A_2 and P_2 components their frequency is the greatest we investigated the high-pass filtered version of the original signal. Since the initial instantaneous frequency of the components is unknown, and its value can vary greatly, filtering was performed with low order FIR filters. In this way the beginning of the components can be emphasized (Fig. 2).

The estimation of the split based on the difference signal was achieved by a heuristic heartbeat detection method developed for fetal phonocardiography [9]. The outline of this algorithm is as follows:

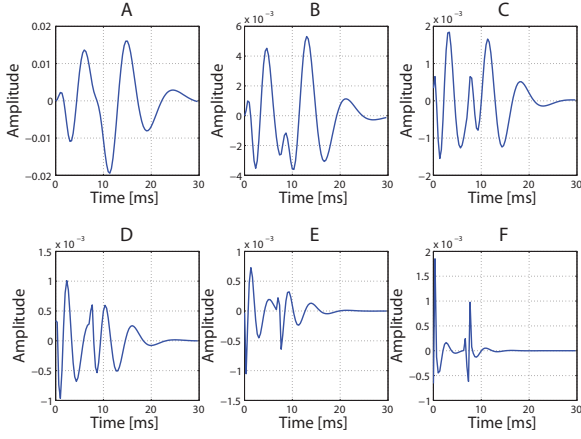


Fig. 2. (A) A synthesized S2 sound and (B-F) its high-passed filtered versions, filtered with FIR filters of order one to five. The cutoff frequency was 345, 414, 455 and 476 Hz, respectively

For a given signal $s[t]$ lets define a kind of contrast enhancement by summing up the signal on a short time-window of length a and taking the difference of neighboring windows:

$$I_1[t] = \sum_{i=t}^{t+a} s[i] - \sum_{j=t-a}^t s[j] \quad (5)$$

Contrast enhancement is achieved by adding this local intensity difference $I_1[t]$ to the original signal $d[t]$, and a secondary local intensity difference is calculated based on this resultant, like in (5) but with a greater time-window of length A :

$$I_2[t] = \sum_{i=t}^{t+A} (d[i] + I_1[i]) - \sum_{j=t-A}^t (d[j] + I_1[j]) \quad (6)$$

Finally, the differences have to be computed in the same way as in (5) regarding the signal $I_2[t]$:

$$V[t] = \begin{cases} -\left(\sum_{i=t}^{t+A} I_2[i] - \sum_{j=t-A}^t I_2[j]\right) & \text{if } > 0 \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

Based on the values a and A , the positive parts of the resultant signal $V[t]$ show the cardiac cycles or the heart sounds. The detection of the S2 sounds was also accomplished with this method.

By applying this method on an S2 signal or on one of its high-passed versions and by decreasing the length of the time windows a and A the assessment of the beginning of the aortic and pulmonary components can be achieved (Fig. 3)

The application of the above described heuristic method showed promising results. Based on the derived model 1000 simulated S2 sounds were generated with random parameter values in the range of real S2 sound parameters. Average error was 4.46 ms, with a standard deviation of 3.82 for unfiltered S2 signals. With high-pass filtering the average error could be decreased to 3.06 ms, standard deviation was 3.01. The error

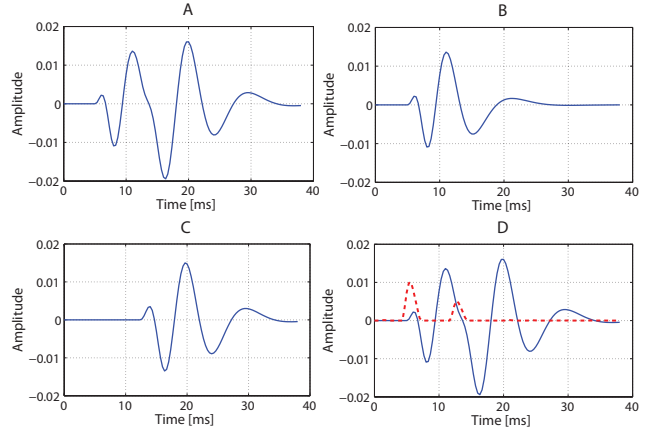


Fig. 3. (A) A synthesized S2 sound with an SI of 7 ms, (C) the aortic and (D) pulmonary components and (B) the result of the heuristic method, $V[t]$ (red), which was applied to synthesized S2 signal. The time difference between the local maxima of $V[t]$ correlate well with the SI.

was significantly higher for small SI values, and the method showed to be reliable for SI values greater than 7 ms.

III. RESULTS AND DISCUSSION

The method described in section II was applied to phonocardiographic records of preterm newborns with PDA. These signals were recorded in a clinical setup: the neonates always lay in an incubator and usually they were connected to a breathing machine. Even with filtering the complete suppression of noise was not possible. This is important to know because by high-pass filtering the usually high frequency noise is also accentuated.

Even though the application of the above described heuristic method on the original or on one of its high-passed versions promising result were achieved (Fig. 4).

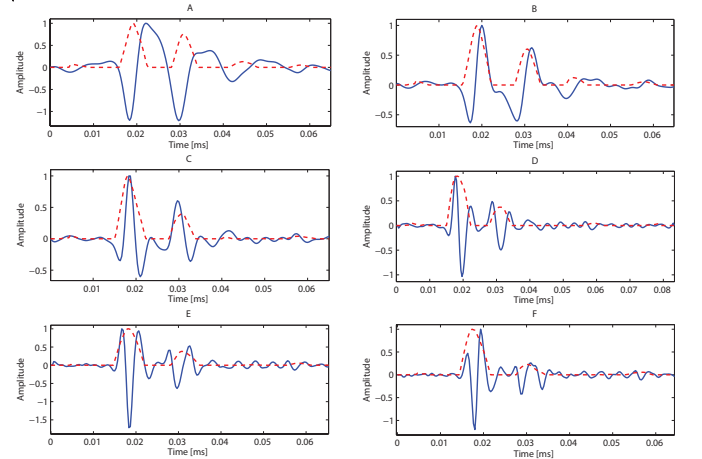


Fig. 4. (A) The S2 sound of a preterm infant recorded after the closure of the PDA (blue) and the output of the heuristic method, $V[t]$ (red). The estimated SI is the time difference between the local maxima of $V[t]$. (B-F) The high-passed filtered versions of the signal, filtered with FIR filters of order one to five, and the the calculated $V[t]$ -s, respectively

An SI estimate for a given record was calculated by finding the two greatest local maxima of the output of the heuristic

algorithm, $V[t]$, in a 33 ms long time window fitted to the S2 sounds. Only those maxima were taken into consideration which were greater than a given threshold (in this study 10 % of the maximum of $V[t]$ was used). The split was assumed to be the time difference between the local maxima described before. If only one maximum was found, than the split was regarded as 0. $V[t]$ was calculated with $a = 1$ ms and $A = 5$ ms.

For a given record the mean value ($m(t_s)$) and the standard deviation ($std(t_s)$) was calculated from all the split estimates for each S2 sound. Those values were rejected, which were outside the interval of $[m(t_s) - std(t_s), m(t_s) + std(t_s)]$. From the remaining values the mean, the standard deviation and the median was computed.

In Fig. 5 the estimated splitting times of those infants can be observed who were treated pharmacological. In case of those neonates who needed surgical intervention the splitting time estimation proved not to be reliable enough because the amplitude of their S2 sounds decreased in a great manner after the ligation. This symptom can be explained by the sudden changes in the circulation due to the operation.

In case of the pharmacologically treated infants, the median of the estimated SI values always increased around the time of the closure of the ductus arteriosus, except in one case. In that case unfortunately it was not possible to make measurements earlier than one day before the closure and four days after the closure. Consequently earlier changes and possible changes one or two days after the closure could not be assessed. The estimated SI increased in average with 11 ms in case of the other three preterms, which is an average change of 85 %. This was computed by calculating the average relative difference between the first estimated SI value after the closure and the local minimum estimated SI before the closure (e.g. one or at most two days before the closure). In case of one infant the estimated SI increased already one day before the clinically verified closure. This might be explained with the dynamical nature of the closing process. In all cases, except one, the estimated SI decreased after the closure, in average after 3.7 days.

As described earlier, the SI depends on the pressure relations between the left and the right side of the heart. These circumstances can be influenced also by other factors, thus for using the estimated SI for diagnostic purposes it is reasonable to take into account other easily measurable parameters, such as the systolic-diastolic pressure ratio, the presence and some parameters of murmur, etc. In further studies we would like to include also these aspects.

IV. CONCLUSION

This paper presents the assessment of the closure of the patent ductus arteriosus (PDA) in preterm infants using phonocardiography. The assessment of closure was based on the estimation of the time interval (SI) between the onset of the aortic and the pulmonary components of the second heart sound. This was achieved by a heuristic method which was also applied to the high passed versions of the of the phonocardiographic signal. This procedure was verified on simulated

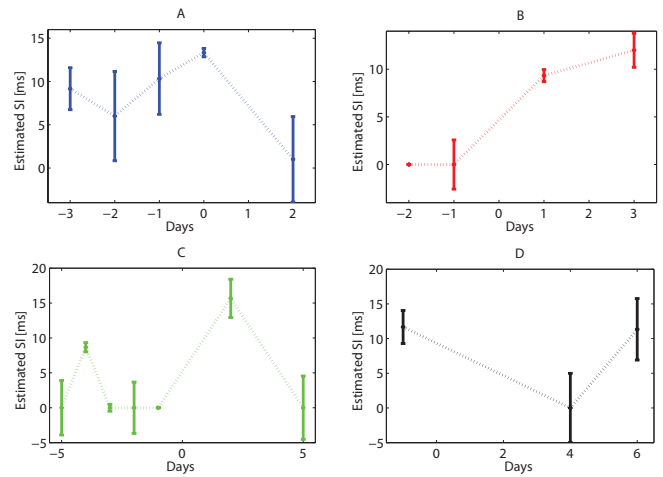


Fig. 5. The estimated SI over several days for preterm infants. All the four figures are drawn in a way that the closure of the PDA lies on Day 0. An increase of the estimated SI around the time of the closure is observable.

data for which a model was derived from the measured data. From the measurements so far we found that the estimated SI increased by 30.33 % around the closure of the PDA in case of pharmacologically treated preterms. For assessing also other aspects of the hemodynamics of the PDA further measurements and analysis are necessary.

ACKNOWLEDGMENT

The author wish to acknowledge Dr. Zoltán Molnár and Dr. Miklós Szabó from the 1st Department of Paediatrics, Semmelweis University, Budapest, for the measurements and their assistance and advise.

REFERENCES

- [1] H. Allen, D. Discoll, R. Shaddy, and T. Feltes, *Moss and Adams' Heart Disease in Infants, Children, and Adolescents: Including the Fetus and Young Adults*, 7th ed. Philadelphia: Lippincott Williams & Wilkins, 2008, vol. 1.
- [2] D. J. Schneider and J. W. Moore, "Patent ductus arteriosus," *Circulation*, vol. 114, pp. 1873–1882, 2006.
- [3] A. Chiruvolu, P. Punjwani, and C. Ramaciotti, "Clinical and echocardiographic diagnosis of patent ductus arteriosus in premature neonates," *Early Human Development*, vol. 85, no. 3, pp. 147–149, 2009.
- [4] Á.T. Balogh, F. Kovács, and Z. Molnár, "Phonocardiography in preterm newborns with patent ductus arteriosus," in *Biomedical Engineering 2010, The 7th IASTED International Conference on*, Innsbruck, 2010, vol. 1.
- [5] F. Kovács, N. Kersner, K. Kádár, and G. Hosszú, "Computer method for perinatal screening of cardiac murmur using fetal phonocardiography," *Comput. Biol. Med.*, vol. 39, no. 12, pp. 1130–1136, Dec. 2009.
- [6] J. Xu, L.-G. Durand, and P. Pibarot, "Nonlinear transient chirp signal modeling of the aortic and pulmonary components of the second heart sound," *IEEE Trans. Biomed. Eng.*, vol. 47, no. 10, pp. 1328–1335, 2000.
- [7] J. Xu, L. Durand, and P. Pibarot, "Extraction of the aortic and pulmonary components of the second heart sound using a nonlinear transient chirp signal model," *IEEE Trans. Biomed. Eng.*, vol. 48, no. 3, pp. 277–283, 2001.
- [8] I. Yildirim and R. Ansari, "A robust method to estimate time split in second heart sound using instantaneous frequency analysis," *29th Conf. Proc. IEEE Eng. Med. Biol. Soc.*, pp. 1855–1858, 2007.
- [9] E. Kósa, A. T. Balogh, B. Úveges, and F. Kovács, "Heuristic method for heartbeat detection in fetal phonocardiographic signals," *Signals and Electronic Systems, ICSES'08. International Conference on*, pp. 231–234, 2008.

Theoretical and Experimental Study of a Digital Microfluidic Chip

Dániel Kovács

(Supervisor: Dr. Kristóf Iván)

kovda@digitus.itk.ppke.hu

Abstract — Microfluidics as a new branch of MEMS technology has appeared in the last decades and has already common applications in inkjet printers or fuel dispensers in spacecrafts. Lately, it has been more and more extensively used in biotechnology. In digital microfluidics individual droplets of biological liquids are manipulated in order to test samples for diseases or the presence of special analytes. Our minimal goal is to devise an open EWOD system which is capable of droplet moving and maybe other droplet manipulation steps. Now I present preliminary numerical models as preparations for a more precise study of such a system and the experimental background that we develop to control an EWOD chip.

Index Terms — microfluidics, digital microfluidics, sessile droplet, lab-on-a-chip, MEMS, EWOD system

I. INTRODUCTION

Microfluidics and digital microfluidics are relatively novel and emergent interdisciplinary technologies on the border of many branches of engineering and science such as electrical engineering, biology, medicine, physics and mathematics. Microfluidics deals with continuous biological liquid flows in miniature channels etched mainly in glass or PDMS (polydimethylsiloxane), a frequently used silicon elastomer, while digital microfluidics, as a subfield of microfluidics, studies the behavior of individual droplets placed on electrodes and exposed to electric field and makes use of it in handling biological samples mostly in medical examinations performed by autonomous and portable devices. Both disciplines, as a part of the lab-on-a-chip (LOC) and MEMS technologies, constitute a bridge over the gap between commercial inorganic silicon microelectronic technology and organic biological systems and seek for the possibility of miniaturizing known medical and biological measurements by downscaling these processes onto a single chip and thereby lowering their cost, time and substance requirement, and making them easier to use – let the user be either a professional or a non-professional.

In this report I would like to summarize the theoretical and experimental work that me and my supervisor have done since the fall semester in digital microfluidics.

II. FUNDAMENTAL PRINCIPLES AND THEORETICAL BACKGROUND

The principal field of study in this area is EWOD (electrowetting on dielectrics) systems, which are devices for liquid droplet manipulation such as droplet moving, mixing, division and merging. These droplets may

contain biological substances or are themselves made of physiological liquids like blood, sweat, tears, urine etc. that are markers of a disease or contain a certain molecule that needs to be amplified. If manipulation of these materials and detection of chemical substances are confined to such small volumes as a liquid drop then all actions of a usual laboratory measurement that earlier required milliliters of liquids can now be done on a portable device which produces a reliable result from only a drop, which is undoubtedly much cheaper and simpler. Our minimal goal is to devise a prototype of a similar digital microfluidic chip which is able to demonstrate the principle of droplet moving.

In such a device, a sessile droplet is placed on one electrode of an electrode array buried under a dielectric layer. The latter assures that the circuit is not closed when the electrodes are activated and so no current can flow through the liquid that could cause it to heat up and evaporate. The usual setup is shown in Fig. 1 representing an open EWOD system, in which the droplet is surrounded by an unconfined medium, mostly air. The

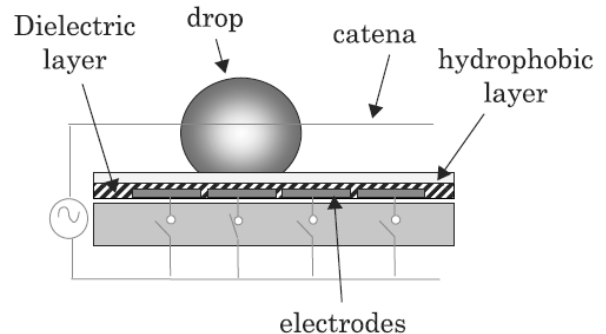


Fig. 1. The structure of a common EWOD system. Adapted from [1].

catena serves as a ground electrode and helps droplet movement along the electrode array, but is not necessarily present in every system. The size of the droplet should be large enough so that a small portion of it overflows to the adjacent electrodes. This is also promoted by the interdigitating shape of the electrodes, as seen in Fig. 2. As an electrode is activated next to another one that is deactivated (or grounded) at the same time the wetting of the droplet is changed due to the electric field which causes the droplet to change its contact angle on the activated electrode and start flowing onto it because of the increased capillary force. This is called electrowetting [1]. The capillary force arises from the fact that every interface tends to minimize its energy (interfacial or surface energy) which in this case generates a force along

the contact line. The surface tension can be interpreted as this force per unit length.

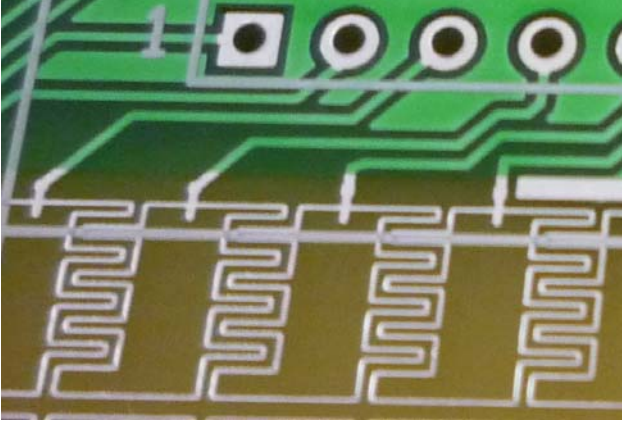


Fig. 2. An open EWOD board with the interdigitating electrodes. A similar setup was used in the present study.

Due to the small size scale, the design of microfluidic systems requires the consideration of such capillary forces which otherwise are overlooked when dealing with macroscopic systems. For a precise treatment this problem should be handled on a mesoscopic scale since the size of the bunch of molecules responsible for the appearance of surface tension forces is in this range, but it is straightforward and plausible to expect that a more phenomenological and simpler description be available so that fundamental assumptions can be made about an EWOD system.

Let a liquid droplet be placed on a flat, horizontal, solid surface surrounded by a gaseous substance. The flat surface is a dielectric usually covered by a hydrophobic layer to facilitate smooth droplet movement. The basics are given in [1] and [2]. As it is deduced in details in [1], the balance among the horizontal projection of surface tension forces on the triple line, along which the liquid, gas and solid meet, is given by the Young law,

$$\gamma_{SL} - \gamma_{SG} + \gamma_{LG} \cos \theta = 0, \quad (1)$$

where θ is the contact angle between the liquid and the solid and γ_{SL} , γ_{SG} and γ_{LG} are the surface tension forces between the solid and the liquid, the solid and the gas, and the solid and the gas, respectively. Digital microfluidics utilizes the fact that the contact angle is dependent upon electric forces that change the ionic concentration near the solid-liquid interface. Based on the pioneering work of Gabriel Lippmann, this electrocapillarity force can be combined with the Young law into the Lippmann-Young law that governs the variation of contact angle with the voltage applied across the solid-liquid interface:

$$\cos \theta = \cos \theta_0 + \frac{C}{2\gamma_{LG}} V^2. \quad (2)$$

Here, θ_0 is the contact angle without electric field, C is the capacitance of the dielectric layer that separates the liquid

from the electrode under the droplet, and V is the voltage. The capacitance is easily characterized by the relative permittivity of the dielectric, ϵ_D , as follows

$$C = \frac{\epsilon_0 \epsilon_D}{d}.$$

In the formula, ϵ_0 is the vacuum permittivity and d is the thickness of the dielectric layer. Using this equality we gain an equation that we also used in our simulations:

$$\cos \theta = \cos \theta_0 + \frac{\epsilon_0 \epsilon_D}{2d\gamma_{LG}} V^2. \quad (3)$$

Based on this equation and the typical dimensions of an EWOD chip, we expect a minimal voltage of several hundred volts to be needed in order to see a significant change in the contact angle.

III. THEORETICAL WORK

For preliminary studies and also for preparing for the experimental work we wanted to find a software that is multiphysical in nature and is able to take into account all physical aspects of the problem of (electro) wetting and yet is relatively easy to use. The COMSOL Multiphysics software package [3] provides a modular simulation environment that is widely used by many laboratories, engineering companies and workshops and proved to meet the requirements mentioned above. We used the 3.5a version, and has just become somewhat acquainted with 4.0, the newest release.

A. Simulation of Dean flow

As a test and to get familiar with COMSOL we tried to reconstruct the result of *Di Carlo* [4]. The review reports on an effect, the Dean flow, which is used in microfluidics for fluid mixing. Because of dimensional reasons, the Reynolds number accounting for the turbulence of a flow is usually rather low which corresponds to a laminar flow. This implies that streamlines do not cross each other which makes mixing a cumbersome process in microfluidics. Dean flow is a secondary phenomenon arising in a flow through a curved channel because of a mismatch of velocity in the downstream direction between fluid in the center and near-wall regions of the channel. This mismatch of velocity is caused by the inertial movement of the fluid as it passes through the bend. The effect was successfully simulated with the MEMS module of COMSOL as shown in Fig. 3. Only the lower half of a horseshoe-shaped curved channel was used as a computation domain since the physics is symmetric to the central horizontal plane. The red tube represents the path of a particle along the flow as it bends upwards (see side view in the inlet) due to the rotating secondary flow depicted by small red arrows in the cross section on the right.

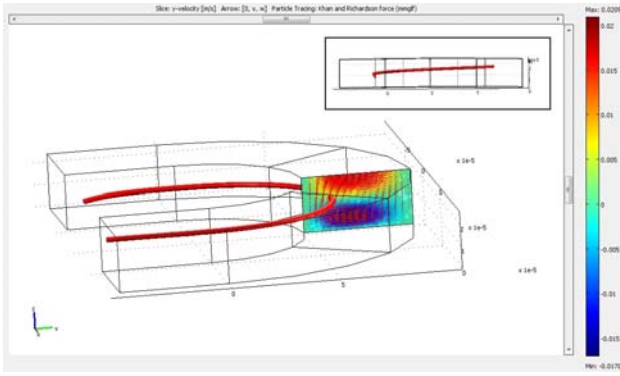


Fig. 3. The reproduction of the results of Di Carlo [2]. View of a water flow in a horseshoe-shaped curved channel. Only the lower half of the channel was involved in the computation due to symmetry in the central horizontal plane. The red tube is the path of a particle floating downstream and bent upwards by the secondary flow (Dean flow) arising from the inertia of the moving fluid in the curve. The color bar shows the horizontal component of the velocity field in the cross section just in the middle of the curve and the rotational movement of the secondary flow is clearly depicted by the small red arrows representing the velocity field. The inset shows a side view of the flow and emphasizes the upward bending of the track.

B. Axial 2D simulation of a sessile droplet

As a second step, a sessile droplet surrounded by air and placed on a dielectric was simulated. In the middle of the droplet a metal catena at high electric potential was immersed. *Hong et al.* [5] report on a COMSOL

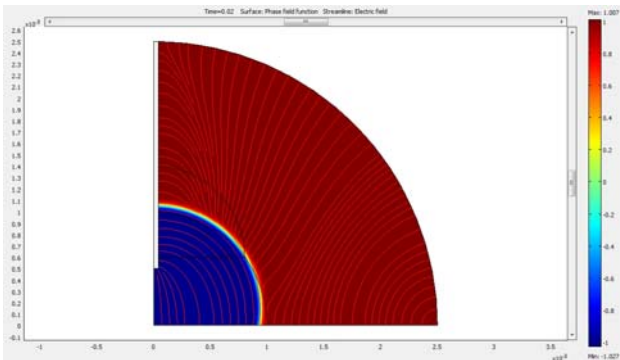


Fig. 4. A sessile droplet of water (blue) in air (red) placed on a horizontal electrode and penetrated by a catena (white) under a DC voltage of 500 V. The voltage dependency of the contact angle is described by equation (3). The color bar shows the volume fraction and the streamlines are electric field lines.

simulation of the shape change of this droplet under an alternating voltage of several (dozens of) kHz and 143 Volt RMS amplitude. Taking into account all physical parameters resulted in a fairly big data set that the software had to handle during calculation therefore the simulation was slow and consumed too much memory and – since our real goal was not to reconstruct this result – we had to settle for testing the fact that COMSOL can really handle a similar problem. We created a setup in which a water droplet of 1 mm radius was placed on a horizontal electrode in ambient air. The catena was held at a constant potential of 500 V and we used the electric and conduction parameters and boundary conditions found in [5] and equation (3) for the voltage dependence

of the contact angle. Fig. 4 presents the shape of droplet after 20 ms the potential difference was applied. The computation made use of the axial symmetry of the system thus the figure is symmetric to the left axis. In the MEMS module the phase field method was used to track the change of the droplet surface and the color bar shows the volume fraction correspondingly (-1 is pure water and +1 is pure air). The streamlines represent the electric field lines.

C. 2D simulation of a closed EWOD system

The third step would have been to carry out the previous simulation in 3D which could have given much more reliable results. Unfortunately, it turned out that both memory and computation time requirements were so large that we could not meet them. Three dimensions is a big step from two, it increases the number of mesh points in the computation domain and the number of degrees of freedom to an extent that the whole problem proves to be untreatable on even an improved PC platform (4 GB memory, 4 processor cores).

Therefore, as a compromise, a 2D closed EWOD system was simulated. The difference between them and open systems is that here droplets are confined in a closed rectangular tube and electrodes are placed under and above the droplet. In some respect, these systems are more efficient due to the symmetric arrangement of electrodes, e. g. droplet movement can be faster and their evaporation can be totally avoided. The medium around the droplet is usually silicon oil which we used in the simulation, too.

We managed to make a complete time dependent simulation that clearly shows how the contact angle and the interface between the two phases change with voltage and how the droplet moves between the electrode plates. Fig. 5a, b, c and d present the shape change and movement of a water droplet (blue) in silicon oil (red). There are four electrodes (thick black lines): one under and one above the initial site of the droplet that are on ground potential, and one in the lower and one in the upper plane having a potential of 500 V. In Fig. 5a, at the initial moment of the time

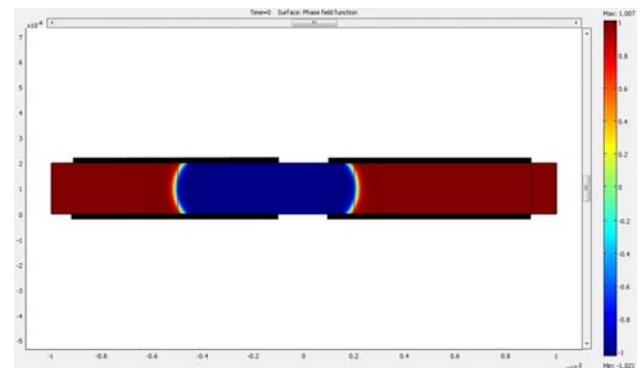


Fig. 5a. A 2D closed EWOD system, onset of simulation.

dependent simulation, the droplet still exhibits a hydrophobic contact angle in zero electric field between ground electrodes. Then, in Fig. 5b it is halfway between

the two electrode pairs with a more hydrophilic angle on its leading edge due to the stronger electric field.

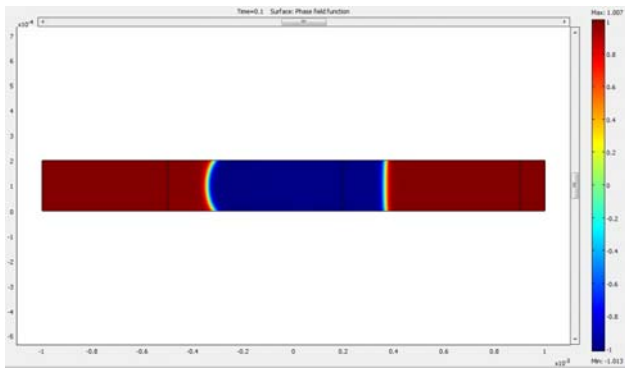


Fig. 5b. 100 ms after the onset of simulation. The right end of the droplet got out of the domain between the ground electrodes and changed its contact angle due to the stronger electric field.

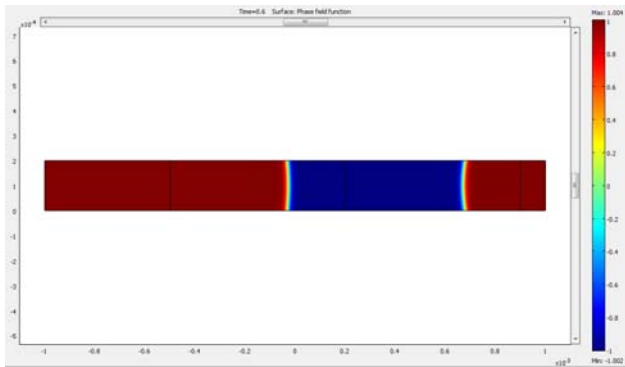


Fig. 5c. 600 ms after the onset of simulation. The droplet approaches the right pair of (activated) electrodes.

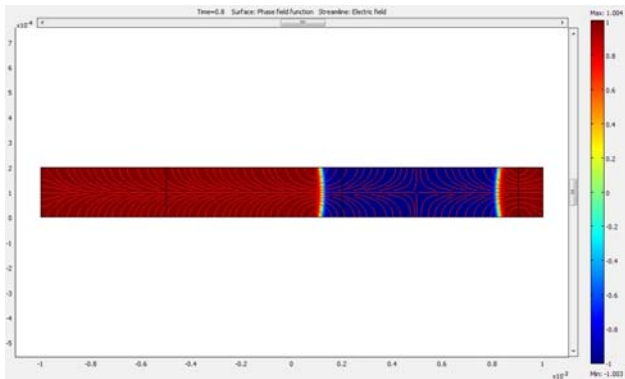


Fig. 5d. 800 ms after the onset of simulation. The droplet reached the activated pair of electrodes and stopped after its left edge got aligned with that of the electrodes. Red lines represent the structure of the electric field.

In Fig. 5c and d it reaches the domain of activated electrodes and comes to resting position as its trailing edge gets aligned with the left edges of the electrode pair. The structure of the electric field between electrodes is also shown.

These three simulations convinced us that COMSOL is a good choice for microfluidic modeling, though a 3D simulation with a finer mesh, which is indispensable for a precise prototype model, makes a bigger memory module absolutely necessary.

IV. EXPERIMENTAL WORK

My former student mate, András Laki, who was also working on this project several years ago, created many printed circuit boards we could use to test droplet moving. Some of these boards can be seen in Fig. 6. These boards mean just one example of several possible electrode arrangements, as the reader can find e.g. in [6].

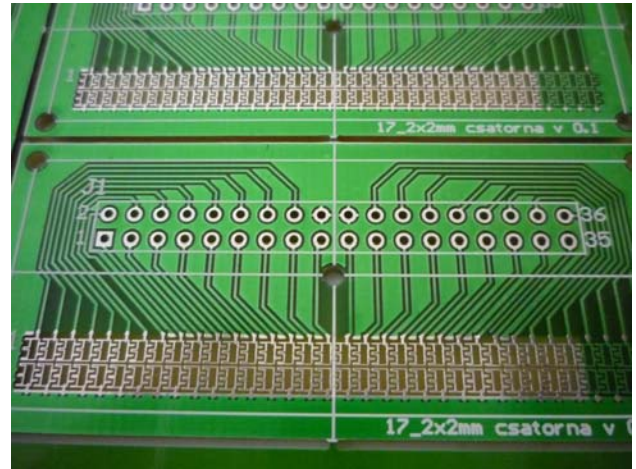


Fig. 6. Two of the printed circuit boards we used for droplet moving. There are two rows of electrode contacts with a dense layout of wiring, and two rows of interdigitating electrodes the lower of which is connected to a common ground.

We have several similar boards with somewhat different sizes of electrodes. In order to be able to move droplets by electricity, we had to cover the electrodes with a dielectric material. We used PDMS for this purpose which was spin-coated on the electrodes with the help of a centrifuge at about 2000 rpm. By our guess, we could achieve a thickness of several tens of μm in this way. Then, as a first step, a DC voltage of about 300 V was switched between two adjacent upper electrodes on the board while their lower counterparts were held at ground potential. We placed a bigger sized droplet of water on

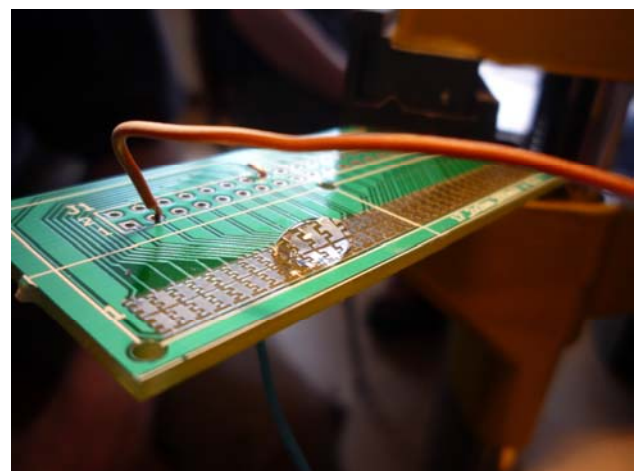


Fig. 7. One of the printed circuit boards with a water droplet on it. The droplet is elongated by the voltage difference between the two upper electrodes under it.

the electrodes and found that the droplet first elongated according to the voltage difference between the electrodes then started to flow toward the electrode at 300 V. This

produced a little asymmetric droplet motion but demonstrated that this simple device is capable of changing the wetting ability of the droplet and setting it into motion. On the other hand, we also experienced that probably due to discontinuities in the PDMS layer the electric insulation was not perfect and water came in direct contact with metal electrodes during the experiment which resulted electrolysis, caused bubble formation at the interface (especially at the corners) and ended in electrode erosion. By applying a thin Teflon layer on the PDMS we could eliminate this phenomenon but a smaller extent of evaporation was still present which we think was caused by that the insulating layer was still too thin (or the electric field is too intense) to prevent dielectric breakdown and a small current could flow through the droplet. Another reason could be that power supply was held constant. Using AC voltage instead of DC could help solve the problem of bubble formation since electrolysis can take place only if a steady current flow is present but this would be blocked by continually changing its direction. A frequency of several 10 kHz can be enough for this to happen as we found a decrease in evaporation at about 30 kHz of sinusoidal signal.

We have also started to assemble a programmable electrode controlling device using the PICDEM HPC Explorer Board from Microchip, as seen in Fig. 8. It connects to the PC through a PIC18F87J50 FS USB Demo Board so that it can be programmed through a USB connection in C language. It contains a microcontroller that switches high and low voltages at pins from where the electrodes on the printed board can be controlled to move droplets.

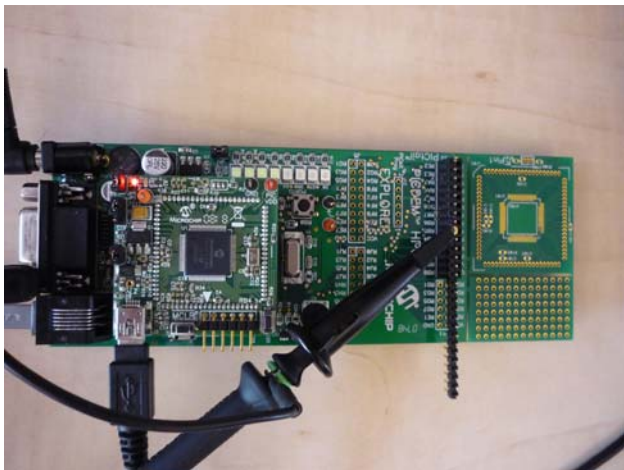


Fig. 8. The upper view of the PICDEM HPC Explorer Demo Board we will use for electrode controlling to move droplets.

IV. FURTHER PLANS

As far as the theoretical study of droplet manipulation is concerned, a real 3D simulation is needed to help the construction of an operating droplet manipulating open EWOD device. This requires more memory to be installed and perhaps the newest release of COMSOL with its newest special CFD module. In practice we want to further develop our electrode controller and create a

board on which an easy manipulation of the droplets is possible by activating proper electrodes under the droplet.

V. ACKNOWLEDGEMENT

We are grateful to Dr. Attila Tihanyi for his technical support in designing the circuitry required for electrode activation and his help in programming the PICDEM Board. We also would like to thank András Laki for his work with the printed circuit boards. The support of OTKA grant PD73653 is greatly acknowledged.

REFERENCES

- [1] H. Bruus, *Theoretical Microfluidics*, Oxford University Press, 2008.
- [2] J. Berthier, *Microdrops and Digital Microfluidics*, William Andrew, 2008.
- [3] <http://www.comsol.com/>
- [4] D. Di Carlo, "Inertial Microfluidics", *Lab Chip*, **9**, 3038-3046, 2009.
- [5] J. S. Hong, S. H. Ko, K. H. Kang and I. S. Kang, "Electromechanical analysis of AC Electrowetting of a Droplet", Excerpt from the Proceedings of the COMSOL Users Conference, 2007, Seoul.
- [6] M. Abdelgawad, P. Park and A. R. Wheeler, "Optimization of Device Geometry in Single-plate Digital Microfluidics", *J. App. Phys.*, **105**, 094506, 2009.

Shape detection of structural changes in long time-span image samples by new saliency methods

Andrea Kovács

(Supervisors: Dr. Tamás Szirányi and Dr. Zoltán Vidnyánszky)

kovan1@digitus.itk.ppke.hu

Abstract—The paper introduces a novel methodology to find changes in remote sensing image series. Some remotely sensed areas are scanned frequently to spot relevant changes, and several repositories contain multi-temporal image samples for the same area. The proposed method finds changes in images scanned by a long time-interval difference in very different lighting and surface conditions. The presented method is basically an exploitation of Harris saliency function and its derivatives for finding featuring points among image samples. To fit together the definition of keypoints and their active contour around them, we have introduced the Harris corner detection as an outline detector instead of the simple edge functions. We also demonstrate a new local descriptor by generating local active contours. Saliency points support the boundary hull definition of objects, constructing by graph based connectivity detection and neighborhood description. This graph based shape descriptor works on the saliency points of the difference and in-layer features. We prove the method in finding structural changes on remote sensing images.

Index Terms—remote sensing, Harris function, change detection.

I. INTRODUCTION

Automatic evaluation of aerial photograph repositories is an important field of research since manual administration is time consuming and cumbersome. Long time-span surveillance or reconnaissance about the same area can be crucial for quick and up-to-date content retrieval. The extraction of changes may facilitate applications like urban development analysis, disaster protection, agricultural monitoring, and detection of illegal garbage heaps or wood cuttings. The obtained change map should provide useful information about size, shape or quantity of the changed areas, which could be applied directly by higher level object analyzer modules [1]. While numerous state of the art approaches in remote sensing deal with multispectral [2], [3] or synthetic aperture radar (SAR) [4], [5] imagery, the significance of handling optical photographs is also increasing [6]. This paper focuses on finding contours of newly appearing/fading out objects in optical aerial images which were taken with several years time differences partially in different seasons and in different lighting conditions. In this case, simple techniques like thresholding the difference image [7] or background modeling [8] cannot be adopted efficiently since details are not comparable. These optical image sensors provide limited information and we can only assume to have image repositories which contain geometrically corrected and registered [9] grayscale orthophotographs. In the literature one main group of approaches is the postclassification comparison,

which segments the input images with different land-cover classes, like arboreous lands, barren lands, and artificial structures [10], [11], obtaining the changes indirectly as regions with different classes in the two image layers [6]. We follow another methodology, like direct methods [2], [3], [4], where a similarity-feature map from the input photographs is derived, then the feature map is separated into changed and unchanged areas. Our direct method does not use any land-cover class models, and attempts to detect changes which can be discriminated by low-level features. However, our approach is not a pixel-neighborhood MAP system as in [12], but a connection system of nearby saliency points. These saliency points define a connection system by using local graphs for outlining the local hull of the objects. Considering this curve as a starting spline, we search for objects' boundaries by active contour iterations. The above features are local saliency points and saliency functions. The main saliency detector is calculated as a discriminative function among the functions of the different layers. We show that Harris detector is the appropriate function for finding the dissimilarities among different layers, when comparison is not possible because of the different lighting, color and contrast conditions. In the following, we introduce a new change detection procedure by using Harris function and its derivatives for finding saliency points among image samples; then a new local descriptor will be demonstrated by generating local active contours; a graph based shape descriptor will be shown based on the saliency points of the difference and in-layer features; finally, we prove the methods capabilities for finding structural changes on remote sensing images.

II. CHANGE DETECTION WITH HARRIS KEYPOINTS

A. Harris corner detector

The detector was introduced by Chris Harris and Mike Stephens in 1988 [13]. The method first computes the Harris matrix (M) for each pixel in the image. Then, instead of computing the eigenvalues of M , an R corner response is defined:

$$R = \text{Det}(M) - k * \text{Tr}^2(M) \quad (1)$$

This R characteristic function is used to detect corners. R is large and positive in corner regions, and negative in edge regions. By searching for local maximas of a normalized R , the Harris keypoints can be found. Normalizing makes R smoother and only major corner points are detected. R

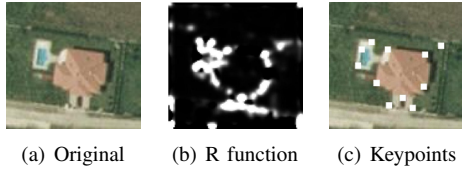


Fig. 1. Operation of Harris detector: Corner points are chosen as the local maximas of the R characteristic function

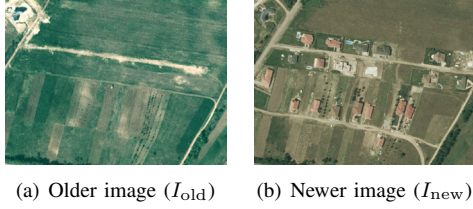


Fig. 2. Original image pairs (given by FÖMI around Szada village)

could also be used for edge detection: $|R|$ function is large and positive in corner and also positive but smaller in edge regions, and nearly zero in flat regions. We used this function in our later work. Figure 1 shows the result of Harris keypoint detection. On Figure 1(b) light regions shows the larger R values, so keypoints will be detected in these areas (Figure 1(c)).

B. Change detection

The advantage of Harris detector is its strong invariance to rotation, scale, illumination variation and image noise. Therefore it could be used efficiently for change detection in airborne images. In these kind of images, changes can mean the appearance of new man-made objects (like buildings or streets), or natural, environmental variations. As image pairs may be taken with large intervals of time, the area may change a lot. In our case the pieces of the image pairs was taken in 2000 and 2005 (Figure 2). It must be mentioned, that these image pairs are registered and represents exactly the same area. In our work we mainly focus on newly-built objects (buildings, pools, etc.). There are many difficulties when detecting such objects in airborne images. The illumination and weather circumstances may vary, resulting different colour, contrast and shadow conditions. The urban area might be imaged from different point of view. Buildings can be hidden by other structures, like trees, shadows, other buildings. These objects are quite various, which also makes the detection tough. To overcome a part of the aforementioned difficulties, our idea was to exploit the difference of the image pairs (I_{diff}). As we are searching for newly-built objects, we need to find buildups, that only exist on the newer image, therefore having large effect on the difference image. Our assumption was to find areas having high effect both in the difference image and in the newer image. These areas may define the keypoint candidates, indicating newly built objects.

To enhance the change caused by buildups, it is more efficient to apply only the red component of the image (when

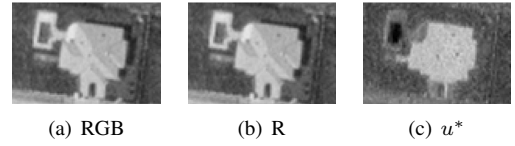


Fig. 3. Grayscale images generated on different components

talking about colour images), rather than all components (see Figure 3). So, from now on, original images are meant as $I_{old} = I_{old}^{red}$ and $I_{new} = I_{new}^{red}$. It is worthy to note, that further on the u^* component of the $L^*u^*v^*$ colorspace (Figure 3(c)) will also play an important role in our algorithm, as it is also effective in enhancing object contours. When searching for keypoint candidates, we call for Harris detector. As written before, the new objects have high effects both on the new and difference image, therefore we search for such keypoints that accomplish the next two criterias simultaneously:

- 1) $R(I_{diff}) > \epsilon_1$
- 2) $R(I_{new}) > \epsilon_2$

$R(\dots)$ indicates the Harris characteristic function (Eq. 1), ϵ_1 and ϵ_2 are thresholds. It is advised to take smaller ϵ_2 , than ϵ_1 . With this choice the difference map is preferred and has larger weight, therefore only important corners in the difference map will be marked. First, we examined the usability of intensity-based and edge-based difference map. These maps resulted keypoints candidates situated mostly on newly built objects, but the ratio of false positive and false negative points was exceeded the acceptable limit. Intensity and edginess are too sensitive to illumination change, so altering contrast and color conditions result the appearance of false edges and corner points and the vanishing of real ones in the difference map. As this preprocessing step is crucial, accuracy is expected to be as high as possible. Therefore we decided to use another metric instead of intensity and edginess and redefine the difference map according to the new metric. The chosen metric was the normalized Harris R characteristic function (\mathbf{R}). Therefore the difference map was calculated as:

$$I_{diff}^R = |\mathbf{R}_{old} - \mathbf{R}_{new}| \quad (2)$$

Modification of I_{new} looks as \mathbf{R}_{new} . The keypoint detection was the same as written before. Results are in Figure 4. Keypoints cover all buildings, and only a few points are false positive. After having some keypoint candidates indicating newly built objects, keypoints defining real changes should be selected somehow.

III. MATCHING WITH LOCAL CONTOURS

A. Detection of similar structures

According to [14], local contours around keypoints are efficient tool for matching and distinguishing, therefore this algorithm was now implemented for Harris corner points. The main steps for estimating local structure characteristics:

- 1) Generating Harris keypoints for difference map



Fig. 4. Result of detection based on the R-function

- 2) Generating the Local Contour around keypoints in the original image [15]
- 3) Calculating Modified Fourier Descriptor for the estimated closed curve [16]
- 4) Describe the contour by a limited set of Fourier coefficients [17]

Our assumption was that after having the FDs for the keypoints, differences between keypoint surroundings can be searched through this descriptor set. We extended the MFD method to get symmetric distance computation as it is written in [14]. By comparing a keypoint (p_i) on the first frame and on the second frame, D_i represents the similarity value. If the following criteria exists:

$$D_i > \epsilon_3 \quad (3)$$

where $\epsilon_3 = 3$ is a tolerance value, than the keypoint is supposed to be a changed area. Else, the keypoint is eliminated.

B. New edge map

After testing the algorithm, we realized that original active contours with intensity-based edge map, are sensible to changes. Even for similar contours, the method often generated false positive result. This meant that changeless places were declared as newly built objects. The aim of this step is to reduce the number of keypoint candidates by eliminating the false hits, therefore a better edge map can be found. The f edge map and E_{ext} external force of the original GVF snake (with $\mathbf{v}(x, y) = (u(x, y), v(x, y))$ vector field that minimizes E_{ext}) [15]:

$$f(x, y) = |\nabla(G_\sigma(x, y) * I(x, y))| \quad (4)$$

$$E_{ext} = \int \int \mu(u_x^2 + u_y^2 + v_x^2 + v_y^2) + |\nabla f|^2 |v - \nabla f|^2 dx dy \quad (5)$$

We used the Harris normalized $|R|$ characteristic function (Section II-A) instead of f edge map:

$$f_{|R|}(x, y) = G_\sigma(x, y) * |R(x, y)| \quad (6)$$

Detected contours are smoother and more robust in case of the $|R|$ -function. We benefit from this smoothness, as contours can be distinguished easier. However, as there is no real contour in



Fig. 5. Enhanced number of Harris keypoints

the neighbourhood of the keypoints, AC-method is only used for exploiting the local information to get low-dimensional descriptor, therefore significance of accuracy is overshadowed by efficiency of comparison.

C. Enhancing the number of saliency points

After selecting the saliency points indicating change, we now have to enhance the number of keypoints. Therefore we are looking for saliency points that are not presented in the older image, but exists on the newer one. We apply the Harris corner detection method again with some modification. By calculating saliency points for older and newer image as well, an arbitrary $q_i = (x_i, y_i)$ point is selected if it satisfies all of the following conditions:

- (1.) $q_i \in H_{new}$
- (2.) $q_i \notin H_{old}$
- (3.) $d(q_i, p_j) < \epsilon_4$

H_{new} and H_{old} are the sets of keypoints generated in the newer and older image, $d(q_i, p_j)$ is the Euclidean distance of q_i and p_j , where p_j denotes the point with smallest Euclidean distance to q_i , selected from H_{old} . New points are searched iteratively, with $\epsilon_4 = 10$ condition. Here, ϵ_4 depends on the resolution of the image and the size of buildings. If resolution is smaller, than ϵ_4 has to be chosen as a smaller value. Figure 5 shows the enhanced number of keypoints.

D. Affiliating edge detection and corner detection

Now an enhanced set of saliency points is given, which can be the basis of building detection. We redefine the problem in terms of graph theory. [18] A graph G is represented as $G = (V, E)$, where V is the vertex set, E is the edge network. In our case, V is already defined by the enhanced set of Harris points. Therefore, E needs to be formed. Information about how to link the vertices can be gained from edge maps. These maps can help us to match only vertices belonging to the same building. If objects have sharp edges, we need such image modulations, which emphasize these edges as far as it is possible. By referring back to Figure 3(b) and 3(c), we have already seen that R component of RGB and u^* component of $L^*u^*v^*$ colorspace can intensify building contours. Both of them operates suitably, therefore we apply both. By generating the R and u^* components (further on denoted as $I_{new,r}$ and $I_{new,u}$) of the original, newer image, Canny edge detection [19] with large threshold ($Thr = 0.4$) is executed on them.

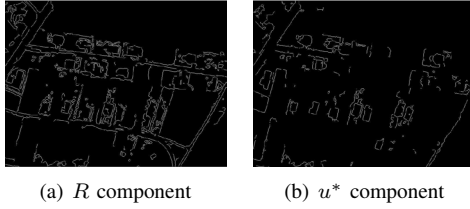


Fig. 6. Result of Canny edge detection on different colour components



Fig. 7. Subgraphs given after matching procedure

C_r and C_u marks the result of Canny detection. (Figure 6(a) and 6(b)) The process of matching is as follows. Given two vertices: $v_i = (x_i, y_i)$ and $v_j = (x_j, y_j)$. We match them if they satisfy the following conditions:

- (1.) $d(v_i, v_j) = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2} < \epsilon_5$,
- (2.) $C_{\dots}(x_i, y_i) = 1$,
- (3.) $C_{\dots}(x_j, y_j) = 1$,
- (4.) \exists a finite path between v_i and v_j on C_{\dots} layer.

$C_{\text{new}, \dots}$ indicates either $C_{\text{new}, r}$ or $C_{\text{new}, u}$. ϵ_5 is a tolerance value, which depends on the resolution and average size of the objects. We apply $\epsilon_5 = 30$. These conditions guarantee that only vertices connected in the edgemap are matched. We obtain a graph composed of many separate subgraphs, which can be seen in Figure 7. Each of these connected subgraph is supposed to represent a building. However, there might be some unmatched keypoints, indicating noise. To discard them, we select subgraphs having at least two vertices. To determine the contour of the subgraph-represented buildings, we used the aforementioned GVF snake method. The convex hull of the vertices in the subgraphs is applied as the initial contour. The object contours were calculated on u^* image component.

IV. EXPERIMENTS AND CONCLUSION

Some results of the contour detection can be seen in Figure 8. The main advantage of our method is that it does not need any building template and can detect objects of any size and shape. The method has difficulties in finding objects with similar colour to the background and sometimes one object is described with more than one subgraphs. These problems need to be solved in a forthcoming semantic or object evaluation step.

ACKNOWLEDGMENT

I would like to thank Dr. Tamas Sziranyi and Dr. Zoltan Vidnyanszky for all the help and support.



Fig. 8. Results of the contour detection.

REFERENCES

- [1] T. Peng, I. H. Jermyn, V. Prinet, and J. Zerubia, "Incorporating generic and specific prior knowledge in a multi-scale phase field model for road extraction from vhr images," *IEEE Trans. Geoscience and Remote Sensing*, vol. 1, no. 2, pp. 139–146, June 2008.
- [2] S. Ghosh, L. Bruzzone, S. Patra, F. Bovolo, and A. Ghosh, "A context-sensitive technique for unsupervised change detection based on hopfield-type neural networks," vol. 45, no. 3, pp. 778–789, March 2007.
- [3] R. Wiemker, "An iterative spectral-spatial bayesian labeling approach for unsupervised robust change detection on remotely sensed multispectral imagery," 1997, pp. 263–270.
- [4] Y. Bazi, L. Bruzzone, and F. Melgani, "An unsupervised approach based on the generalized gaussian model to automatic change detection in multitemporal sar images," vol. 43, no. 4, pp. 874–887, April 2005.
- [5] P. Gamba, F. Dell'Acqua, and G. Lisini, "Change detection of multitemporal sar data in urban areas combining feature-based and pixel-based techniques," vol. 44, no. 10, pp. 2820–2827, October 2006.
- [6] P. Zhong and R. Wang, "A multiple conditional random fields ensemble model for urban area detection in remote sensing optical images," vol. 45, no. 12, pp. 3978–3988, December 2007.
- [7] C. Benedek, T. Szirányi, Z. Kato, and J. Zerubia, "Detection of object motion regions in aerial image pairs with a multilayer markovian model," *Trans. Img. Proc.*, vol. 18, no. 10, pp. 2303–2315, 2009.
- [8] Cs. Benedek and T. Sziranyi, "Bayesian foreground and shadow detection in uncertain frame rate surveillance videos," vol. 17, no. 4, pp. 608–621, April 2008.
- [9] C. Shah, Y. Sheng, and L. Smith, "Automated image registration based on pseudoinvariant metrics of dynamic land-surface features," vol. 46, no. 11, pp. 3908–3916, November 2008.
- [10] L. Castellana, A. d'Addabbo, and G. Pasquariello, "A composed supervised/unsupervised approach to improve change detection from remote sensing," vol. 28, no. 4, pp. 405–413, March 2007.
- [11] C. Benedek, X. Descombes, and J. Zerubia, "Building extraction and change detection in multitemporal remotely sensed images with multiple birth and death dynamics," in *IEEE Workshop on Applications of Computer Vision (WACV)*, Snowbird, USA, 2009, pp. 100–105.
- [12] Cs. Benedek and T. Szirányi, "Change detection in optical aerial images by a multi-layer conditional mixed markov model," *IEEE Trans. Geoscience and Remote Sensing*, vol. 47, no. 10, pp. 3416–3430, October 2009.
- [13] C. Harris and M. Stephens, "A combined corner and edge detector," in *Proceedings of the 4th Alvey Vision Conference*, 1988, pp. 147–151.
- [14] A. Kovacs and T. Sziranyi, "Local contour descriptors around scale-invariant keypoints," in *International Conference on Image Processing*, Cairo, Egypt, Nov 2009, pp. 1105–1108.
- [15] C. Xu and J. L. Prince, "Gradient vector flow: A new external force for snakes," in *IEEE CVPR*, 1997, pp. 66–71.
- [16] Y. Rui, A. She, and T. Huang, "A modified fourier descriptor for shape matching in MARS," in *Image Databases and Multimedia Search*, 1998, p. 165180.
- [17] A. Licsar and T. Sziranyi, "User-adaptive hand gesture recognition system with interactive training," *Image and Vision Computing*, vol. 23, no. 12, pp. 1102–1114, 2005.
- [18] B. Sirmacek and C. Unsalan, "Urban-area and building detection using sift keypoints and graph theory."
- [19] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698, 1986.

Dynamic Feature and Signature Selection for Robust Tracking of Multiple Objects

Vilmos Szabo
(Supervisor: Dr. Csaba Rekeczky)
szavi@digitus.itk.ppke.hu

Abstract—The goal of this paper is to summarize a new tracking framework, which exploits dynamic feature and signature selection techniques for data association models. It performs robust multiple object tracking in a noisy, cluttered environment with closely spaced targets. This method extends the back-end processing capabilities of tracking systems by creating a two-level hierarchy between the parallelly extracted features. These features are dynamically selected based on spatio-temporal consistency weight function, which maximizes the robustness of data association, and reduces the overall complexity of the algorithm.

I. INTRODUCTION

Multiple object or target tracking is an important task in computer vision applications. However, it can become a challenging problem, especially if the object is in a dynamically changing environment. A number of computer vision applications could be characterized by two complex stages of processing. The first stage is the topographic image acquisition, which may include pre-processing, image segmentation, and post-processing. The second stage is a non-topographic sensing which includes feature-signature extraction, data assignment, and state-prediction. High resolution spatio-temporal detection can be accomplished using topographic or cellular processing hardware, such as the CNN (Cellular Neural Network) [1]. The multiple object tracking back-end is usually accomplished using serial DSPs (Digital Signal Processor). Therefore, the numerical complexity of the tracking algorithm is crucial in order to meet the systems real-time demand. This paper focuses on object tracking using dynamic data association and its spatio-temporal signature analysis. Application areas may include traffic monitoring, vehicle navigation, automated surveillance, and biological applications.

II. DYNAMIC MULTIPLE TARGET TRACKING FRAMEWORK

Multiple target tracking can be defined, as estimating the trajectory of objects in the image plane as they move around in the scene [2]. Generally, an object segmentation algorithm runs on each frame of the video flow in order to detect objects. This can be done on a CNN-like massively parallel topographic hardware to achieve high spatio-temporal resolution video flow processing. The detected objects are then assigned to consistent labels, called *tracks* [3]. The temporal analysis of tracks can be used to identify and select features that best represent each object. The final goal of target tracking is to determine the position of an object or a bounding box on

each frame of the video sequence. Our algorithm follows a bottom-up approach:

- 1) Pre-processing of input video flow
- 2) Parallel image segmentation algorithm
- 3) Post-processing of segmented video frame
- 4) Image labeling
- 5) Object shape and appearance representation
- 6) Parallel image feature extraction
- 7) Image feature normalization and selection
- 8) Assignment of object to tracks based on dynamic feature selection
- 9) Feature signature analysis

Steps 1–3 can be implemented on CNN-type hardware. Pre-processing of each video frame is an important step to eliminate unwanted noise, and to condition the signal for further analysis. Throughout the evaluation, Gaussian filtering that can be approximated on the CNNs resistive grid was employed. The time or scale parameter depends on the amount of noise in the scene. The range of pixel intensity values was converted to $I \in \{-1; 1\}^{NM}$ where N and M is the width and height of the image. In case of color processing, each chromatic channel is processed separately (see subsection C-Parallel Feature Extraction).

For post-processing, basic mathematical binary morphological [4], [5] operators were used. The aim was to connect fragmented objects with the closing operation, and to clear individual pixels created by the “non-perfect” segmentation algorithm. The k-step closing operation consists of a k-step dilatation followed by k-steps erosion. The k-step opening operation consists of a k-step erosion followed by k-steps dilatation.

Steps 4–9 are typical serial DSP-like processing. Each of the connected components is labeled on the binary image. A number of features are extracted from the connected components for the dynamic tracking. The results of feature signature analysis (8) have a feedback to the dynamic feature analysis (6) in order to calculate the track consistency metric (see subsections B-D for details).

A. Motivation

The motivation for employing dynamic feature selection for multiple object tracking emanates from the need to reduce the complexity of data association steps of the the overall algorithm. Let \mathbf{x} and \mathbf{y} be d-dimensional vectors where each

component corresponds to a feature value. The two most widely used distance metrics are the \mathcal{L}_1 city block (eq. 1) and \mathcal{L}_2 Euclidean (eq. 2) metrics.

$$\mathcal{L}_1 : d_1(\mathbf{x}, \mathbf{y}) = \|\mathbf{x}, \mathbf{y}\| = \sum_{n=1}^d |x(n) - y(n)| \quad (1)$$

$$\mathcal{L}_2 : d_2(\mathbf{x}, \mathbf{y}) = \|\mathbf{x}, \mathbf{y}\| = \sqrt{\sum_{n=1}^d (x(n) - y(n))^2} \quad (2)$$

The best feature will provide the maximum interclass distance between objects. Increasing the feature space dimensionality will increase the discriminative power. Therefore, the algorithm should try to select as few salient features as possible for data association. This decreases the number of features that need to be extracted. There are existing methods for dimensional reduction, such as *Principal Components Analysis* (PCA) [6]. These methods usually require a training set, or block processing for dimension reduction. The feature selection method explained in this paper is a recursive one; it has a relatively low computational complexity and is able to successfully select a set of salient features in a changing environment.

B. Simulation Videos

The algorithm was evaluated on three computer generated video flows. The first video *Scene 1 (Shapes)* contains five dynamically changing objects. See Figure 1 for a demonstration on three objects. Each object is able to change its location, visibility, orientation, color, shape, noise, and inner structure according to the following list:

- Location: [0–1]
- Visibility: [0–1]
- Orientation: [0–360°]
- Color: [red, green, cyan, blue]
- Shape: [circle, triangle, square, pentagon]
- Noise: [on off]
- Inner structure: [dots, lines, concentric circles]

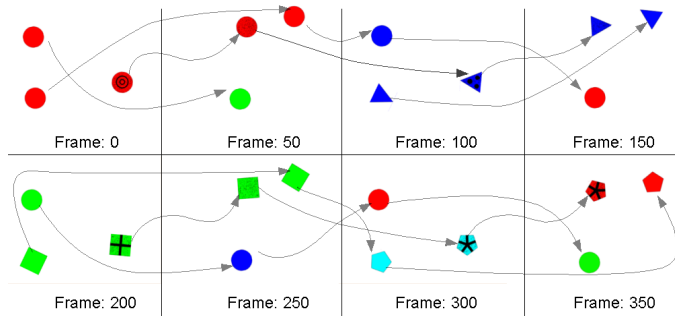


Fig. 1. Demonstration of the computer generated simulation video flow containing dynamic feature transformations of the objects. The dynamic transformations include location, color, shape, noise and inner structure changes.

The second video flow is called *Scene 2 (Bipeds)*. This scene contains walking humans with crossing and overlapping paths;

they are in partial and full occlusion, entering and exiting the scene. The third video flow is called *Scene 3 (Cars)*. The first two scenes contain non-rigid objects, while the third scene contains only rigid objects. Figure 2 shows actual frames from all three video flows, and Figure 3 shows the noisy versions. The parameter of Gaussian noise was $\mathcal{N}(0, 0.015)$ and resulted in an $SNR_{dB} = 15$.

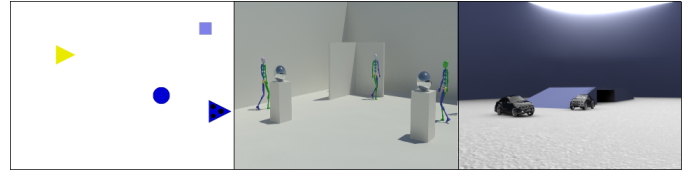


Fig. 2. Computer generated video flows that were used in the algorithmic evaluation. From left to right: Scene 1 (5 dynamically changing shapes), Scene 2 (6 Bipeds), Scene 3 (4 Cars)

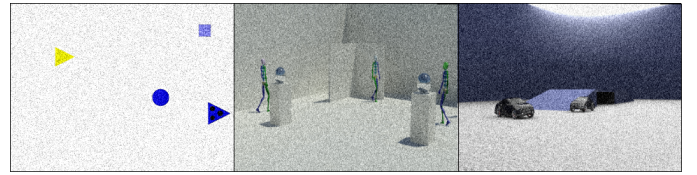


Fig. 3. Computer generated video flows with added Gaussian noise. From left to right: Scene 1 (Shapes), Scene 2 (Bipeds), Scene 3 (Cars)

C. Hierarchical Feature Extraction

The input image is highly redundant. The transformation, to reduce the dimensionality of input data while keeping relevant information content, is called *feature extraction*. For each object, a number of features are calculated from the image. A set of six statistically independent features that can identify and describe each object in a given frame were extracted. These features are summarized below:

TABLE I
SUMMARY OF THE TWO LEVEL HIERARCHICAL FEATURE EXTRACTION USED FOR THE DYNAMIC TRACKING FRAMEWORK.

Feature Group	Subgroup
1. Position	Location
	Speed
	Acceleration
3. Scale	Area
	Major Axis Length
	Minor Axis Length
	Bounding Box
4. Shape	Eccentricity
	Solidity
	Extent (Opacity)
5. Structure	Euler Number
6. Texture	Variance
	Average Y Luminance Component
7. Color	Average Cb Color Component
	Average Cr Color Component

The result of the segmentation algorithm is a binary mask, where black pixels correspond to the background pixels and white pixels correspond to the foreground pixels. The x and y coordinates are the centroid of each connected foreground pixel on the mask image. The shape of the objects is described by the eccentricity metric, which measures how much the object deviates from a circle. In order to extract the color information, the original color input image is converted to YCbCr color space. For each chromatic channel (luminance, blue and red) the average intensity value is extracted. Finally, each object is represented by this six dimensional vector (see Figure 4). This vector is normalized between 0 and 1 values in order to make comparable measurements between each frame of the video sequence.

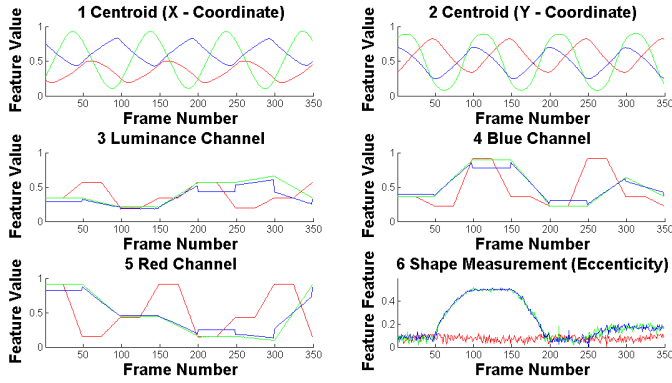


Fig. 4. Feature extraction results for three objects in Scene 1 simulation video flow. All feature values are normalized to [0–1] interval. The features shown in the figure in order from left to right and top to bottom direction: x, y coordinates, eccentricity, luminance channel, blue, and red chromatic channels

The change of the inner structure is considered as noise throughout the experiment. The choice of the feature set should be based on the requirements of a specific application area.

D. Dynamic Spatio-Temporal Feature Selection

In a real-time application, the number of features should be minimal to increase the speed of the system, but all the relevant information must be kept. This can be done by creating a hierarchy in the features based on their confidence or robustness. The noisy feature channels should be filtered out. The *tracking system* consists of feature selection, data assignment, state space estimation, prediction and error correction.

1) *Feature Selection*: The feature selection is done by analyzing the spatial and temporal property of each feature channel. The “good” features are selected based on a spatio-temporal consistency metric. Let \mathbf{x}_k^i and be the feature state space vectors at frame k for the i-th object. Let $\mathbf{Q}_k^{ij}(n)$ quality matrix (eq. 3) be the minimum of pair wise l_1 (eq. 1) distance of the current state space vector n-th component between the i and j-th objects.

$$\mathbf{Q}_k(n) = \min\{d_1(\mathbf{x}_k^i(n), \mathbf{x}_k^j(n))\} \mid (i > j) \quad (3)$$

The second term of the consistency metric is the inverse of the residual gradient magnitude of the previous state space estimation. Features that are well separated from each other and do not change much in time are preferred. The final consistency metric (eq. 4) is defined by a linear μ parameter homotopy of the first part and the second parts. (The variable m is the actual number of features.)

$$\mathbf{C}_k = (1 - \mu)\mathbf{Q}_k + \mu \frac{1}{\frac{1}{m} \sum_{i=1}^m |\mathbf{x}_{k-1}^i - \mathbf{x}_k^i|} \quad (4)$$

\mathbf{C}_k vector contains the quality measurement for each feature at a given time. Different feature selection strategies can be considered. A fix number of best features can be selected, or features can be selected above a given threshold level, resulting in a varying number of features for each frame. Figure 5 demonstrates the case where only the best feature is selected. Please see section III. - Performance Evaluation for comparison of the different feature selection strategies.

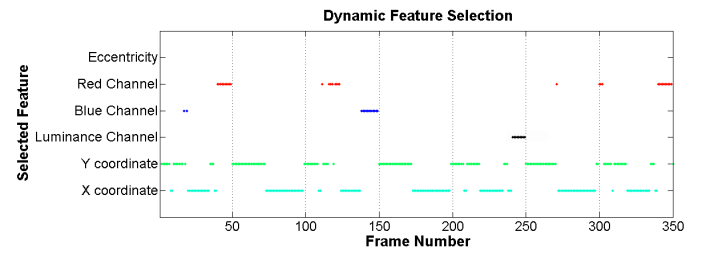


Fig. 5. Feature selection results based on spatio-temporal consistency metric of tracking states. The x-axis shows the frame number in the simulation video. The y-axis represents the selected feature.

2) *Data Assignment*: The assignment of measurements to consistent tracks is accomplished using a combinatorial optimization algorithm called the *Hungarian* [7] method. Only features that are selected contribute to the calculation of the distance matrix. The assignment algorithm is used to match the current and predicted states together with minimal cost. The time complexity of the assignment algorithm is low order polynomial. More complex data association models can be applied, such as described in [8].

3) *State Space Estimation: Interacting Multiple Model* (IMM) was used as state estimation framework. Recursive steady-state *Kalman* filters [9] are used for the state prediction and correction phases. The prediction filters are also know as alfa, alfa-beta, alfa-beta-gamma filters. Particle filters may be used to improve the accuracy of the results [10].

III. PERFORMANCE EVALUATION

The evaluation of the tracking algorithm was on computer simulated videos. For each frame, the mask video flows were also generated. Figure 6 displays example frames of the binary masks that were used as references for system performance evaluation.

The black and white mask images do not contain information about tracking individual objects. For this reason an object map is synthesized, where each object is assigned a

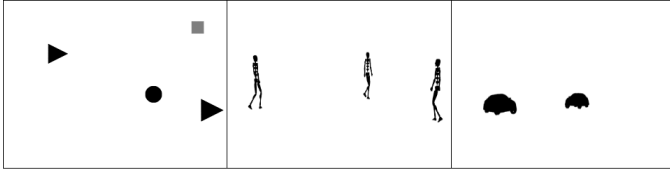


Fig. 6. Reference mask image flows for the simulation videos. From left to right: Scene 1 (Shapes), Scene 2 (Bipeds), Scene 3 (Cars). (The images in the figure have been inverted.)

unique color for track representation (see Figure 7). For each color value, the x and y coordinates are extracted, which gave the reference tracks for the evaluation (see Figure 8). Note that for Scene 1 (Shapes) the object ID mask is multiplied with the corresponding binary mask before evaluation.

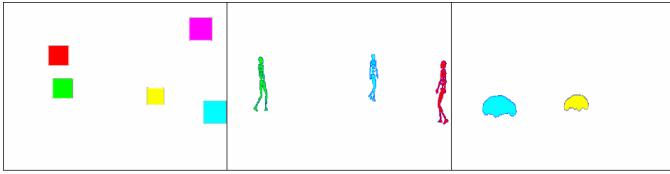
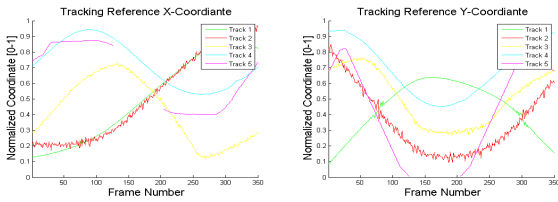
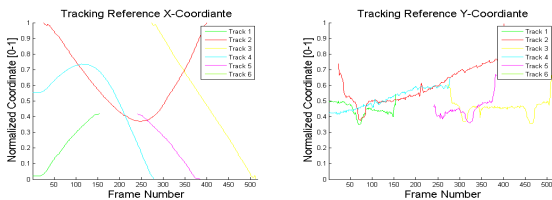


Fig. 7. Object ID map flows for the simulation videos. Each object is assigned a unique color. From left to right: Scene 1 (Shapes), Scene 2 (Bipeds), Scene 3 (Cars). The images have been modified for printing.

Scene 1 (Shapes) :



Scene 2 (Bipeds) :



Scene 3 (Cars) :

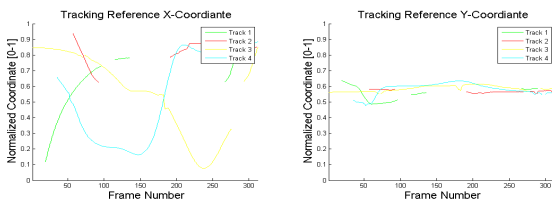


Fig. 8. Reference tracks extracted from Object ID Map. From top to bottom: Scene 1 (Shapes), Scene 2 (Bipeds), Scene 3 (Cars). The left column corresponds to the x coordinate of the tracks, and the right column correspond to the y coordinates of the tracks

The *mean square error* (MSE) is calculated between the reference track and measured tracks. The MSE can be calculated from the following equation (eq. 5):

$$MSE = \frac{1}{n} \sum_{i=1}^n d_1(Ref_x, Meas_x) + d_1(Ref_y, Meas_y) \quad (5)$$

A total of three feature selection strategies were evaluated. The first is the *Best-Feature* selection strategy, where only one feature is selected. The second is the, *K-Dynamic* selection, where a feature is selected above a given threshold level. This threshold was set such that the achieved MSE is around the third selection strategy. The third strategy is when all the features (*All-Feature*) are used by the algorithm.

Figures 9, 10 and 11 show the mean square error on each frame for Scene 1 (Shapes), Scene 2 (Bipeds) and Scenes 3 (Cars), respectively.

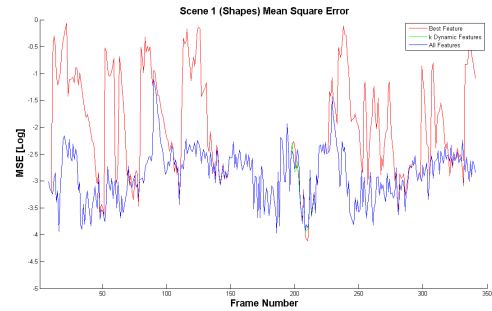


Fig. 9. Scene 1 (Shapes) mean square error comparison of the three feature selection strategies. The x axis is the frame number, and the y axis is the MSE in logarithmic scale.

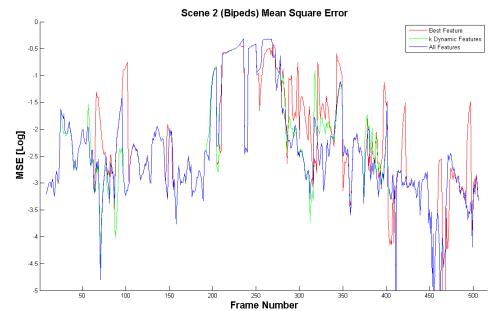


Fig. 10. Scene 2 (Bipeds) mean square error comparison of the three feature selection strategies. The x axis is the frame number, and the y axis is the MSE in logarithmic scale.

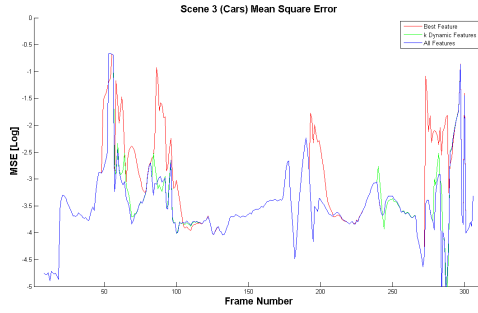


Fig. 11. Scene 3 (Cars) mean square error comparison of the three feature selection strategies. The x axis is the frame number, and the y axis is the MSE in logarithmic scale.

Table 1 summarizes the average MSE measurement for all the scenes.

TABLE II

SUMMARY OF MSE MEASUREMENT BETWEEN THE REFERENCE AND MEASURED TRAJECTORIES FOR THE DIFFERENT FEATURE SELECTION STRATEGIES.

Scene	Feature Selection Method		
	Best-Feature	K-Dynamic	All-Features
Scene 1 (Shapes)	0.0643	0.0025	0.0025
Scene 2 (Bipeds)	0.0462	0.0454	0.0443
Scene 3 (Cars)	0.0060	0.0040	0.0039

The *K-Dynamic* selection preforms equally to the *All-Feature* selection strategy.

Table 2 shows the average number of features that were extracted. The *K-Dynamic* selection used 3-4 features on average rather than all the features.

TABLE III

SUMMARY OF THE AVERAGE NUMBER OF FEATURE SELECTION FOR THE DIFFERENT SELECTION STRATEGIES.

Scene	Feature Selection Method		
	Best-Feature	K-Dynamic	All-Features
Scene 1 (Shapes)	1	4.04~ 4	7
Scene 2 (Bipeds)	1	2.97~ 3	7
Scene 3 (Cars)	1	2.59~ 3	7

Figure 8 shows the final object detection and tracking result. The images also include the bounding box and trajectory of each tracked objects.

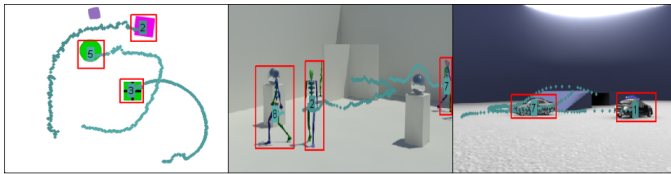


Fig. 12. Object detection and tracking results on the three simulation videos. Scene 1 (Shapes) frame: 215, Scene 2 (Bipeds) frame: 250, Scene 3 (Cars) frame: 200

IV. CONCLUSION

A new tracking framework that uses a dynamic feature and signature selection method was introduced. This algorithm can be used to track objects in a changing environment after a topographic CNN-like segmentation and feature extraction. The algorithm arranges the parallelly extracted features into a hierarchy, based on their consistency measurement. The overall complexity is reduced by keeping only the relevant features for tracking the objects in the scene. This saves considerable processing time. Based on tests on synthesized videos, it has been confirmed that selecting 3-4 features dynamically could result in as good tracking as using all the features. Future work will include the development of more complex selection strategies and testing the algorithms on real video sequences.

REFERENCES

- [1] L. O. Chua, T. Roska, T. Kozek, and A. Zarandy, *Cellular Neural Networks*. New York: Wiley, 1995, ch. The CNN paradigm a short tutorial, p. 114.
- [2] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM Comput. Surv.*, vol. 38, no. 4, p. 13, 2006.
- [3] K. Smith, D. Gatica-Perez, J. Odobez, and S. Ba, "Evaluating multi-object tracking," in *In Workshop on Empirical Evaluation Methods in Computer Vision*, 2005.
- [4] R. M. Haralick, S. R. Sternberg, and X. Zhuang, "Image analysis using mathematical morphology," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 9, no. 4, p. 532550, 1987.
- [5] K.-R. Park and C.-N. Lee, "Scale-space using mathematical morphology," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, p. 11211126, 1996.
- [6] I. T. Jolliffe, *Principal Component Analysis*, 2nd ed. Springer, October 2002.
- [7] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2, p. 8397, 1955.
- [8] R. Karlsson and F. Gustafsson, "Monte carlo data association for multiple target tracking," *IEEE Target Tracking: Algorithms and Applications*, 2001.
- [9] G. Welch and G. Bishop, *An introduction to the kalman filter*, ser. Tech. Rep. NC, USA: Chapel Hill, 1995.
- [10] X. Wang, S. Wang, and J. Ma, "An improved particle filter for target tracking in sensor system," *Sensors 2007*, vol. 7, no. 144156, 2007.

GPGPU Accelerated Scene Segmentation Using Nonparametric Clustering

Balázs Varga

(Supervisors: Dr. Kristóf Karacs and Dr. Gábor Szederkényi)

varga.balazs@itk.ppke.hu

Abstract—Our aim in this summary is to take a step closer to semantic image understanding. We utilize the bottom-up approach consisting of image segmentation followed by a cluster merging procedure. Our main tool is a nonparametric image clusterization method: the mean shift algorithm using joint spatial-range feature space. We consider spatial information so that the mean shift can distinguish topographically differing objects in the scene. However, this feature costs additional computational demand through increased number of kernel functions. For this reason the proposed algorithm runs the mode-defining kernel iterations parallelly by utilizing the many-core processor architecture present in the general-purpose graphics processing unit (GPGPU). We use our own voting procedure for pixel-cluster assignment. We applied numerical evaluation to show that our solution efficiently speeds up the image clusterization procedure. The algorithm is also tested to work on images containing typical scenes of the Bionic Eyeglass Project.

Index Terms—Image understanding, Image segmentation, Parallel processing

I. INTRODUCTION

Scene segmentation is one of the most common, yet most versatile tasks of image processing, for which we can enumerate two main approach methodologies:

- **The top-down methodology** constructs image segments by finding objects in the scene using descriptors (shape, color, texture etc.) and/or semantic information.
- **The bottom-up methodology** constructs image segments by assembling the previously oversegmented scene's segments, using descriptors (shape, color, texture etc.) and/or semantic information.

In our scene segmentation framework, we use the mean shift algorithm, which follows the principles of the bottom-up approach. Among today's most advanced segmentation methods, such as graph cuts [1], normalized cuts [2], or various types of k-means [3][4], the mean shift is one of the most studied and applied nonlinear techniques. Basically having no *a-priori* knowledge demand, it can dynamically set the number of segmented clusters through a nonparametric framework. Our solution gives a notable speed-up to the mean shift method, by parallelizing its internal structure. Instead of running it on a CPU, we use a many-core *general-purpose graphics programming unit* (GPGPU) [5]. The motivation here is, to construct a real-time scene segmentation algorithm, which can aid numerous tasks formulated in the Bionic Eyeglass Project (BEP) [6], such as banknote -, traffic light- or crosswalk detection [7].

II. ALGORITHMIC BACKGROUND

After the mean shift procedure was introduced by Fukunaga and Hostetler [8] in 1975, it was Cheng [9] who pointed out 20 years later that the mode seeking process of the algorithm is a parallel hill climbing method, applying the clusterization algorithm in the Hough space. Following another half a dozen of years of smoldering, Meer and Comaniciu gave an extensive overview [10] of the segmentation framework, using it for image segmentation and discontinuity preserving smoothing. Their approach concerning color images is briefly summarized in subsection II-A.

A. Mean shift in the joint feature space

The mean shift procedure considers its feature space as an empirical probability density function. A local maximum of this function (namely, a region over which it is highly populated) is called a mode. Mode calculation is formulated as an iterative scheme of mean calculation, which takes a certain number of feature points and calculates their mean value by using a weight kernel function. Meer and Comaniciu used a composite feature space consisting of both topographical (*spatial*) and color (*range*) information of the image. As a result, each feature point in this space is represented by an $\chi = (x_r; x_s)$ 5D vector which consists of the corresponding pixel's $x_s = (x, y)$ 2D position in the spatial lattice, and its x_r 3D color value in the applied color space (e.g., the $x_r = (Y, Cb, Cr)$ coordinates). The iterative scheme for the calculation of a mode is as follows: let χ_i and z_i be the 5D input and output points in the joint feature space for all $i = [1, n]$, n being the number of pixels in color image I . Then for each i

- 1) Initialize $k = 1$
- 2) Compute the iterative formula

$$\chi_i^{k+1} = \frac{\sum_{j=1}^n \chi_j g \left(\left\| \frac{x_{r,j} - x_{r,i}^k}{h_r} \right\|^2 \right) g \left(\left\| \frac{x_{s,j} - x_{s,i}^k}{h_s} \right\|^2 \right)}{\sum_{j=1}^n g \left(\left\| \frac{x_{r,j} - x_{r,i}^k}{h_r} \right\|^2 \right) g \left(\left\| \frac{x_{s,j} - x_{s,i}^k}{h_s} \right\|^2 \right)} \quad (1)$$

until the mean shift vector $\|\chi_i^{k+1} - \chi_i^k\|$ falls under a given threshold, where g denotes the Gaussian kernel function, with h_s and h_r being the spatial-, and range bandwidth parameters respectively.

- 3) Allocate $z_i = \chi_i^{k+1}$; that is, output value z_i is given by feature point χ_i after the final $(k + 1)^{th}$ step.

Those z_i points, which are adequately close to each other, are concatenated resulting discrete, non-overlapping clusterization of the input image. All pixels in the cluster inherit the color of its mode. The main advantage of applying the joint feature space is that the algorithm became capable of discriminating scene objects based on their color *and* position; making mean shift a multi-purpose, nonlinear, nonparametric tool for image segmentation. On the other hand the disadvantage of the algorithm, as it was specified earlier by Cheng is its high computational complexity of $\mathcal{O}(n^2)$.

B. Acceleration strategies

Comaniciu and Meer introduced an efficient technique called the coarse grid [11] in order to highly reduce complexity. Briefly, they perform random tessellation of the feature space with $m \ll n$ kernels, execute mean shift segmentation (resulting $z_i, i = [1, m]$ modes), merge close modes (as in [10]), then assign each χ_i feature point to the closest z_j cluster-defining mode. They showed that this approach is capable of producing technically equal segmentation quality with a computation demand of $\mathcal{O}(m * n) \ll \mathcal{O}(n^2)$. Ever since, several alternative techniques have been proposed to achieve speed-ups, e.g. through space discretization and downsampling [12], local subsets [13], expectation-maximization [12][14], hierarchical solutions [15][16], and the Newton iteration method [17][18]. Although our current system does not use these alternative techniques, later on most of them can easily be added to our framework enhancing its speed and reducing its time demand.

C. Oversegmentation: the advantage's tradeoff

The clear advantage brought by the usage of the spatial domain is the topographical discriminative potential: objects with similar or even the same color can be distinguished, if they are topographically distinct. On the other hand spatial discrimination of two discrete objects requires the usage of two kernels. Therefore in the case of very detailed images, spatial filtering involves a computational tradeoff: proper coverage of the feature space necessitates the usage of numerous kernels.

III. OUR APPROACH

Both the original and the improved algorithms follow the bottom-up strategy, when the output is a result of oversegmentation, which is followed (off-line), or accompanied (on-line) by a cluster merging procedure. We considered the off-line mean shift algorithm, which is divided into two main subtasks: the mode calculation and the cluster merging procedure. Mode calculation algorithm is a highly data parallel [19] task: the same iterative procedure is performed on the elements of the feature space with each kernel having a different seed point. We implemented the mode seeking task on a GPGPU, and compared its performance with the CPU implementation of the procedure.

A. Architectural motivation

The design of the CPU and the GPGPU is approached completely different. A state of the art CPU has six cores and can utilize two threads per core, which enables it to work fast on linear data streams. Our day's GPGPUs initiate over twenty thousand threads on hundreds of cores, which are arranged in a topographic mesh. For this reason the GPGPU can outperform the CPU in the case when working on single-instruction-multiple-data (SIMD) related tasks, such as the calculation of the mode points. However, computation on the GPGPU has an important drawback: CPU to GPGPU and GPGPU to CPU memory transfers are highly time consuming, so that CPU-GPGPU hybrid processing should be avoided as much as possible.

B. GPGPU implemented mode calculation

Our algorithm applies Comaniciu and Meer's coarse grid technology [11] on the joint feature space. But instead of running the iteration specified by eq. 1 on a single kernel until saturation, we extend this computational framework by running the iterative algorithm simultaneously on several mean shift kernels, which we call *multiple simultaneous mode seeking* (MSMS).

The MSMS begins by selecting m initial mean points randomly in the joint feature space. The mean shift iteration is then started from these seed points by using Gaussian kernel functions having a common (h_s, h_r) spatial-range bandwidth parametrization for each kernel. The procedure is terminated, when the length of each kernel's mean shift vector becomes smaller than a pre-defined threshold value.

In order to properly assign all feature points to the corresponding mode, we constructed a voting system. In every iteration of mode seeking, for each kernel we compute pixel-wisely the following *cumulative confidence value* (CCV):

$$C_{i,j}^{k+1} = C_{i,j}^k + g \left(\left\| \frac{x_{r,j} - x_{r,i}^k}{h_r} \right\|^2 \right) g \left(\left\| \frac{x_{s,j} - x_{s,i}^k}{h_s} \right\|^2 \right) \quad (2)$$

with

$$C_{i,j}^0 = 0 \quad (3)$$

where $C_{i,j}^k$ denotes the confidence value of pixel i at the k^{th} iteration for kernel j . Note that the calculation of the CCV does not require additional computation, as it is a part of the mean shift iteration's numerator. Let $C_{i,j}$ denote the final CCV

computed in the last iteration and let CID_i stand for the i^{th} pixel's *cluster ID*. After the modes are retrieved and every $C_{i,j}$ has been obtained, each image pixel gets associated with a mode using the following rule:

$$CID_i = \arg \max_j (C_{i,j}) \quad (4)$$

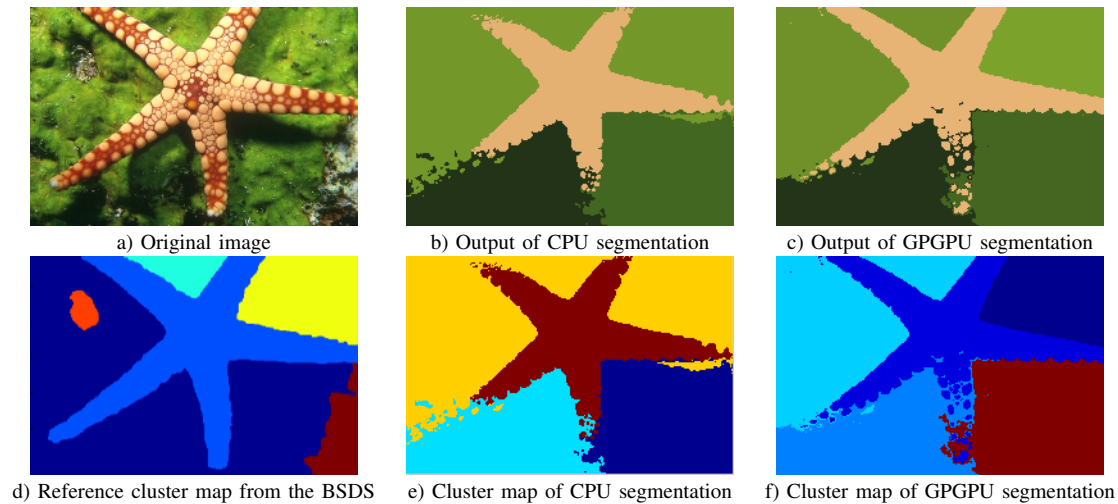


Fig. 1. An example of segmentation made on the different computation platforms. Segmented images are result of mode seeking followed by cluster merge. $(h_s, h_r) = (0.08, 0.01)$, $m = 54$ clusters were merged into 4 (CPU case) and 5 (GPGPU case)

C. CPU implemented cluster merging

The number and structure of final clusters are constructed with cluster merging, which currently runs on the CPU. Cluster i and j are joined, if they satisfy the criteria:

- C1. The two clusters have a common border in terms of the eight-neighbor connectivity.
- C2. $\|x_{r,i} - x_{r,j}\| < h_r$

In this case the position of the mode is recalculated. Let NP_i and NP_j denote the number of pixels in clusters i and j respectively, and let us say that both C1 and C2 criteria holds and the two are merged into cluster k . Then the color information carried by mode z_k of the newly formed cluster is

$$z_k = \frac{NP_i * z_i + NP_j * z_j}{NP_i + NP_j} \quad (5)$$

i.e. it is a weighted average of the conjoined duet. The procedure runs iteratively until no classes can be merged.

IV. EVALUATION FRAMEWORK

The proposed algorithm was evaluated from two different aspects:

- 1) Quality and performance comparison.
- 2) A real-life segmentation task.

Our aim in the first case was, to show that the GPGPU variant of the algorithm provides the exact same quality as the CPU implementation but with a much smaller runtime demand. For this reason, the coarse grid CPU-optimized mean shift procedure was implemented with as few differences from the GPGPU version as possible. The main difference is that while the GPGPU runs the MSMS version, the CPU does the mode calculation one by one (*single simultaneous mode seeking*, SSMS). In the second case the algorithm was enhanced with customized pre-, and post-processing in order to solve an actual clusterization assignment. The motivation here besides a use case, is that in general, image segmentation is an ill-posed problem [20]. In our second evaluation task, the details

- such as the characteristics of the input and the output - are somewhat well-posed, enabling better quality comparison possibilities. A short description of the two evaluations are given in the following subsections.

A. Quality and performance comparison

Several publicly available image corpora exist in order, to make different algorithms comparable. For the evaluation of the segmentation algorithms, we selected 50 color images from the Berkeley Segmentation Dataset (BSDS), [21] for which multiple human-made segmentation maps are provided as reference. Let the name *best parameter pair* (BPP) denote a pair of (h_s, h_r) kernel bandwidth parameters that result in a closest-to-the-reference segmentation for a given image of our *evaluation image set*. In order to determine such pairs, the CPU algorithm was utilized the following way: 64 alternative segmentations were made for each evaluation image using 8×8 different (h_s, h_r) bandwidth values for each of the 50 images. During the process, the algorithm computed and logged the number of clusters before and after the merging procedure, and the elapsed time of the mode seeking. Then, depending on the number of provided BSDS references 1 to 4 BPPs were selected for each image, resulting a total of 117 pairs. Next, the GPGPU algorithm was run on the evaluation set using the corresponding BPPs, and finally the elapsed time of the mode seeking was compared to the CPU algorithm. It is worth to note that the GPGPU runtime includes the time-demanding CPU to GPGPU and GPGPU to CPU data transfers. Furthermore, since the number of mean shift iterations depends on the (randomly selected) initial kernel position, we ran both algorithms multiple times with the same parametrization, and compared the average running times. Moreover cluster merging was done using the same CPU-based algorithm; therefore time consumption of the mode merging procedure was not part of the comparison. All measurements were run on a single PC equipped with 2GB of RAM and an Intel E6400 CPU running at 2.13GHz. The

GPGPU was an nVidia G92 GPU operating with 112 stream processors and 1024MB of video RAM.

B. Existing segmentation task

The segmentation framework was also evaluated in a real-life application. A subtask of the Bionic Eyeglass Project (see section I) is to detect, whether there is a crosswalk present on the scene [22]. This is done in three steps:

- 1) Isolate asphalt on the image.
- 2) Extract white stripes from asphalt, if present.
- 3) Decide, whether they form a crosswalk.

Our segmentation algorithm was used to substitute the previous version of the asphalt detection. After analyzing the characteristics of the subtask, the segmentation algorithm was enhanced with two, computationally cheap steps: color-based pre-filtering and morphological post-processing.

V. RESULTS

In the case of performance measurements, on average, the GPGPU was able to segment the scene 2.997 times faster than the CPU, under the same circumstances. Fig. 1/a) displays an example image from the BSDS among with a segmentation made by human observer on 1/d), and our segmentation results made on the different computational platforms on 1/b) and 1/c). In the case of the asphalt segmentation task, the enhanced framework has proven to work more accurate than the original asphalt segmenter algorithm lacking the mean shift [23]. Fig. 2 displays an example of an input image, and an output on which the non-asphalt pixels are masked out.

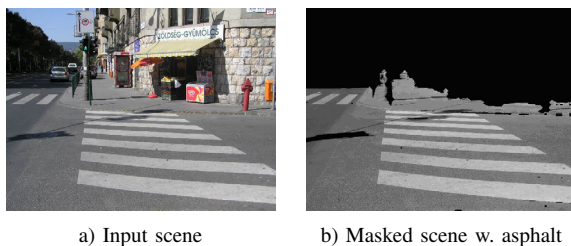


Fig. 2. An example of the asphalt isolation task. The mean shift algorithm successfully masked out the majority of pixels not belonging to the asphalt.

VI. CONCLUSION

We successfully implemented the nonparametric mean shift clustering algorithm using the joint spatial-range feature space onto the many-core GPGPU, for which we used our own voting procedure for mode selection. The GPGPU algorithm proved to run almost three times faster than its CPU variant. The algorithm proved to work well in a use case of asphalt detection. Later on we plan to enhance the system using the acceleration strategies described in subsection II-B.

REFERENCES

[1] V. Kwatra, "Graphcut textures: Image and video synthesis using graph cuts," *ACM TRANSACTIONS ON GRAPHICS*, vol. 22, pp. 277–286, 2003.

[2] J. Shi and J. Malik, "Normalized cuts and image segmentation," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Juan, Puerto Rico, 1997, pp. 731–737.

[3] P. Bradley and U. Fayyad, "Refining initial points for K-Means clustering," in *ICML '98: Proceedings of the Fifteenth International Conference on Machine Learning*. Morgan Kaufmann Publishers Inc., 1998, pp. 99, 91.

[4] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, "An efficient k-means clustering algorithm: Analysis and implementation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 881–892, 2002.

[5] D. Luebke, M. Harris, N. Govindaraju, A. Lefohn, M. Houston, J. Owens, M. Segal, M. Papakipos, and I. Buck, "GPGPU: general-purpose computation on graphics hardware," in *Proceedings of the 2006 ACM/IEEE conference on Supercomputing*. Tampa, Florida: ACM, 2006, p. 208.

[6] K. Karacs and M. Radvanyi, "A prototype for the bionic eyeglass," in *Proc. 12th Int Cellular Nanoscale Networks and Their Applications (CNNA) Workshop*, 2010, p. 1.

[7] K. Karacs, A. Lazar, R. Wagner, D. Balya, T. Roska, and M. Szuhaaj, "Bionic eyeglass: An audio guide for visually impaired," in *Proc. IEEE Biomedical Circuits and Systems Conf. BioCAS 2006*, 2006, pp. 190–193.

[8] K. Fukunaga and L. Hostetler, "The estimation of the gradient of a density function, with applications in pattern recognition," *Information Theory, IEEE Transactions on*, vol. 21, no. 1, pp. 40, 32, jan 2003.

[9] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 8, pp. 799, 790, 1995.

[10] D. Comaniciu and P. Meer, "Mean shift: a robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, 2002.

[11] —, "Distribution free decomposition of multivariate data," *Pattern Analysis and Applications*, vol. 2, no. 1, 1999.

[12] M. Carreira-Perpinan, "Acceleration strategies for gaussian Mean-Shift image segmentation," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 1 (CVPR'06)*, New York, NY, USA, 2006, pp. 1160–1167.

[13] K. Zhang, M. Tang, and J. Kwok, "Applying neighborhood consistency for fast clustering and kernel density estimation," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, San Diego, CA, USA, 2005, pp. 1001–1007.

[14] M. A. Carreira-Perpinan, "Gaussian Mean-Shift is an EM algorithm," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 5, pp. 767–776, 2007.

[15] S. Paris and F. Durand, "A topological approach to hierarchical segmentation using mean shift," in *2007 IEEE Conference on Computer Vision and Pattern Recognition*, Minneapolis, MN, USA, 2007, pp. 1–8.

[16] D. DeMenthon, "Spatio-temporal segmentation of video by hierarchical mean shift analysis," *Language*, vol. 2, p. 1, 2002.

[17] M. A. Carreira-Perpinan, "Mode-finding for mixtures of gaussian distributions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1318–1323, 2000.

[18] C. Yang, R. Duraiswami, D. DeMenthon, and L. Davis, "Mean-shift analysis using quasi-Newton methods," in *Proceedings of the International Conference on Image Processing*, vol. 3, 2003, pp. 447–450.

[19] W. D. Hillis and G. L. S. Jr, "Data parallel algorithms," *Communications of the ACM*, vol. 29, no. 12, p. 1183, 1986.

[20] T. Poggio and V. Torre, "Ill-posed problems and regularization analysis in early vision," *Artificial Intelligence Lab. Memo*, vol. 773, 1984.

[21] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, Vancouver, BC, Canada, 2001, pp. 416–423.

[22] M. Radvanyi, G. E. Paziienza, and K. Karacs, "Crosswalks recognition through cnns for the bionic camera: Manual vs. automatic design," in *Proc. European Conf. Circuit Theory and Design ECCTD 2009*, 2009, pp. 315–318.

[23] M. Radvanyi, B. Varga, and K. Karacs, "Advanced crosswalk detection for the bionic eyeglass," in *Proc. 12th Int Cellular Nanoscale Networks and Their Applications (CNNA) Workshop*, 2010, pp. 1–5.

Distortion analysis and possible corrections of the pictures projected by a projector

András Gelencsér
(Supervisor: Prof. Tamás Roska)
gelan@digitus.itk.ppke.hu

Abstract — We contact increasingly often with digital projectors nowadays, slowly they will be an integral parts of our daily lives. It is a rightful claim that we would like to see the projected picture in the same colours as the original picture on our monitor display. However, we may experience, that this expectation is not always fulfilled. The different devices which we use and our environment have a distortion effect on the projected picture, which may modify the gradation of the colors in lesser or greater measure. To eliminate this distortion and to result a high-quality image is not an easy task, for the simplest example because each of us sees the world in a little different way, our color experiences may be various a little bit. During my work I examined the factors which can modify the original picture. These should be taken in consideration during the projection. I also considered which kinds of image processing and image analysing techniques would be proper to achieve the projected picture to be as accurate as possible in correlation with the picture which can be seen on the computer.

Index Terms — projector-camera system, colors, radiometric model, image processing

I. INTRODUCTION

The different digital projectors gain more and more space in daily life, we use them in education, at work, on conferences. The cheaper and better quality projectors appear in the households slowly. There are two major factors, which are affecting our visual sensation. The first is the quality of the projected picture, which depends not only on the quality of the apparatus, but from many other factors, like the surface of the projection screen (e.g.: wall or canvas), or on the environmental illumination. The second is the image and colour processing mechanism of our eyes. Everybody sees the world a little different and the colours play a big role in our object recognition [1]. For example the character and background colour is very important in the texts.

Maybe the following problem is familiar for everybody: while we see the pictures perfectly on our monitor screen, the audience sees badly or hardly the projected result. The colours of the picture may be modified because of the disturbing factors mentioned above, and the quality is deteriorating apparently. The manual calibration of the projector, so as to correct the picture of the projection, is not always self-evident. The solution could be an automatic system, which would consider the different disturbances and compensate those errors.

My idea is the following: I take some shoots from the projected pictures with the help of a camera and I would calculate the distortion, in the possession of the original pictures on the computer, with the help of image processing procedures. With this data I would deform my original pictures, so the modification and the disturbance during the

projection will extinguish one another and the final result will be such a projected picture, which is the possible nearest one to the picture on the monitor.

II. GENERALLY ABOUT THE PROJECTORS

A digital projector is such a device, which can take the video signal and projects it onto a projection screen, using very bright light and a lens system. The quality of the projected image depends on the native display resolution, the light output (or brightness) and the contrast. The bigger these three parameters are, the more beautiful picture could we received. The next projector types are common [2]:

CRT projectors use red, green and blue cathode ray tubes. This is the oldest system still in regular use, because it can provide the largest screen.

LCD projectors use liquid crystal display chip. It is a sort of pixel array. Each individual pixel can allow the light to pass through or not and the combination of these dots will produce the image (Figure 1.). This is the most simple system and it is affordable for business or home use.

DLP projectors use Texas Instruments' DLP technology [3]. This technology works with 2 million hinge-mounted microscopic mirrors. This mirrors control the amount of projected light. There are two types of these in the way of color creating. The first type is which uses single DLP chip and color filter the other type uses three DLP chips, each of them for one of the basic colors (RGB).

LCoS projectors use liquid crystal on silicon technology. It is similar to the DLP technology; however, it uses highly reflective liquid crystals instead of individual mirrors.

LED projectors use one of the above mentioned technologies for image creation. The only difference is that their light source is an array of Light Emitting Diodes. It's a great advantage that it doesn't need lamp replacement.

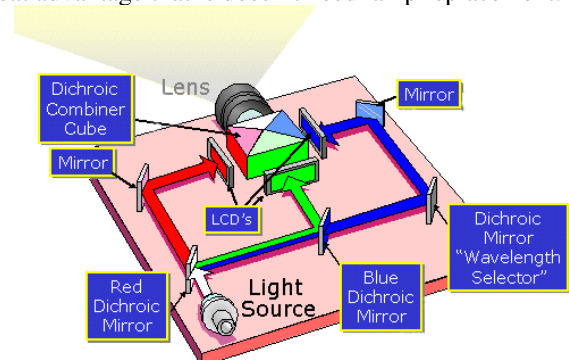


Figure 1. An illustration for the LCD projector. The final image is the combination of the RGB components.

III. PROJECTOR-CAMERA SYSTEM

I will use the following system (Figure 2.). This model is set up according to the usual projector usage. I project some pictures or slideshow from my laptop, which is connected to a digital projector. The projector radiates the image to a projection screen. Nowadays most of the laptops have an inbuilt camera or if not we can purchase a cheap USB camera. With this device I can acquire the result image and upload to the laptop so I can compare the original to the acquired picture, analyze the differences, determine the distortions and modify the original image to get a better projected image against the different distortions.

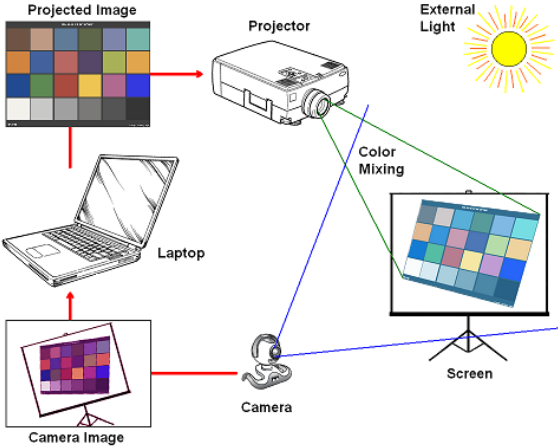


Figure 2. My projector-camera system. The figure illustrate that the original picture may go through changing because of the different distorting effects.

IV. RADIOMETRIC MODEL

With the help of the radiometric model, I can describe the color transformation from the projector through the projection screen to the camera. With the consideration of the possible disturbing factors the following formula should be written (1):

$$C_L = \int (f(\lambda) + P_K \omega_K(\lambda)) s(\lambda) q_L(\lambda) d\lambda \quad (1)$$

Let's examine this formula in its details. The result C_L is the brightness perceived by the camera on L color channel. P_K means the brightness of the projector's K color channel. Usually K and L are the basic RGB channels. $\omega_K(\lambda)$ is the spectral response for the K projector channel, where λ is the wavelength of the light. The brightness is modulated by the spectral response. Mark the irradiance of the environmental lightning with $f(\lambda)$ and add to the previous product. The $s(\lambda)$ parameter is the spectral reflectance of the projected screen surface and at last, but not least $q_L(\lambda)$ means the camera spectral response for L color camera channel. These last two parameters are also modulating the final result. The integration is done over the visible spectrum.

There are a couple of projects which use this model to define the color mixing caused by the applied apparatus and let prepare a compensation algorithm to enhance the projected image, for examples see [4], [5].

V. DISTORTION FACTORS

We examine the possible occurring anomalies in this chapter in details. It is an important viewpoint how the possible factors influence the quality of the projected picture. According to that we can decide whether it is necessary or not to deal with a specified factor.

We begin it with the projector. The main modification factor here is the spectral response. It should be mentioned, that spectral response is unfortunately not linear on the full region, so the projected image's brightness can differ from the needful signal, and these results the projected colors may be modified. In this case we must reduce or increase the necessary color channel(s) power according to the brightness level to get a better picture. The figure 3 shows an example. If the video signal means the 100% level then at low brightness level the red and blue channel are stronger, the green channel weaker like 100%.

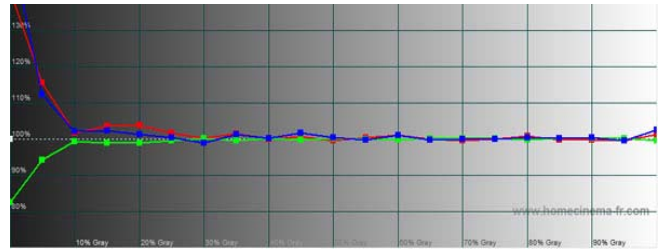


Figure 3. A general projector spectral response. At low or high brightness level the projected signal can differ from the received video signal.

The spectral response is constant for a device, so if we define it once we can count with it afterwards.

At older projectors several annoying phenomena can occur. The most typical is lamp weakening. The easiest way to solve this problem is the component replacement. But it can occur that only one of the channels get wrong and the picture get a bluish or reddish shade. Of course the spectral response calibration can solve this problem to, because it is a special case of this. Another usual error is, when the light power is not homogeneous in the whole projected screen. The middle of the screen is too bright, but the corners are darker. To fix this mistake, we need some sort of mask or map, which describes the unhomogeneous of the screen.

The projection screens have important parameters too which affect the result image. The spectral reflectance of the screen depends on the material and the texture. Canvas with an ideal reflectance and a homogeneous surface is not always available, sometimes we must use for example the wall, which have its own texture, and accordingly it can disturb the result. So the algorithm has to prepare to adapt the projection to the different surfaces. Of course we may regard the projection screen constant, during the projection, namely it doesn't change its surface or color.

The camera's spectral response modifies the measured picture as well, so we must only calculate with it during the correction algorithm, but I don't need to compensate this one, because it doesn't contribute to the final result. Usually the cameras have not only color alteration, but some mapping error too, for example radial distortion. These errors can be negated with camera calibration techniques.

I ran into another problem during the examination of the measurements. I noticed that the background colors are not constant on the consecutive pictures. The reason of this is that cameras have an auto-white-balance function. The camera try to acquire such a pictures which white balance is near to a predefined value. If we assume that the

illumination and the background do not change under this short time period of the calibration, then the white balance alternation on the measured pictures depends only on the projected image. Because this AWB function, the camera sees the pictures in different luminousness, although it isn't true. This function should be disabled or the white balance should be homogenized in the pictures, to get correct measurements.

Important to mention the external lights, for example the Sun or the artificial lights, the neon lamps. Probably these change the most dynamically between normal circumstances (the sky will be clouded, the audience need more light to make notes etc.), while the previous discussed parameters can be static under a given projection, disregarding the changes because of the cameras AWB function.

VI. MEASUREMENTS

As I mentioned earlier I use only such devices for the experimental measurements which are spread in the everyday usage. So I don't use any spectrometer and don't make any modification on the projectors or cameras .

After I reviewed what kind of disturbance factors may be taken into consideration in the course of the projection, I created a test series. The series is a short slide show, which contains necessary pictures according to my opinion to collect data about the distortions during the projection. The first few slides contain respectively only one pure color: red, green, blue, black or white.

As I have already mentioned, most projectors produce the whole picture from the three basic colors. With RGB slides I can analyze if the projector can generate the true color values and whether the channels work correctly or not.

In case of a fully black slide theoretically the projector don't project anything. Of course some little light can escape from the device but if the projector is not too old, then it is irrelevant. But there are cases when the projector can't block the light and the black image isn't good so it must be corrected. White is not so important, but I can check the maximum projection power of the projector.

Before I constructed the test slide show I looked after which methods are usually used to determine the color generation of a device. Macbeth Color Checker is a chart, which is an important tool for serious photographers and film developers [6]. It provides a set of known reference colors which can be used as a setup and adjustment standard in film and video production (figure 4).



Figure 4. Macbeth Color Checker is a collection of standardized color samples, used for color comparisons and measurements such as in checking the color reproduction of an imaging system or calibrates a device like a digital camera. On the left (A) is the CIE standard, on the right(B) my own Color Chart, based on the original one (A).

It's a great advantage is, that it can provide valuable information if comparing the projected Macbeth Chart with

the original one. There are different color standard so there are many Macbeth Chart too. I chose the CIE standard (figure4. (A)) and made a slightly modified one, because the original chart includes only pale colors, as in the nature and in the world these colors are frequent. But as we usually see pure colors on the computer screen, my chart contains the original colors in a little brighter variation.

I took a few nature photos at the end of the test slides, so I have data from projection between normal circumstances.

The measurement process is the following. I project the slides with the projector and meanwhile from each single picture I take a screenshot with the help of the connected camera on the laptop. I repeat the measurement in different circumstances, for example I turn off or on the light I pull the drapery in, I change the position of the camera. I made records with different projectors in different rooms and built a database with enough data to make some analysis. Here are some examples from the database (figure 5.), one from a good projector (A) and one from an older one (B). In the latter, we can see that the blue channel became more dominant. With spectral response calibration this problem can be solved.



Figure 5. Two snapshots from the measurement database. On the left (A) we see a picture from a new projector, on the right (B) from an older one. We may observe that the right projected image glitters in blue nuance while the left projector gives a beautiful clear picture.

VII. POSSIBLE CORRECTION TECHNIQUES

Correction of the image quality is based on the concept of feedback. The projected as well as the acquired picture can be taken and the original one is given, so difference can be measured. Using this difference the result image can be enchanted.

As mentioned before, the radiometric model is basis of many corrective algorithms. Using this model, the color mixing caused by the projector's spectral response and the reflectance of the projection surface can be eliminated. It seems to worth implementing an algorithm using this method, since its competence is already proved [4], [5].

Adaption to the changing light is another interesting problem. With other words we would like to attain, that a given picture look the same as possible, against the different light circumstances, according to the opportunities looks like the same. This problem is the so called color constancy. The human eye has the ability to determine the colors of a known object independently from the color of the light source. Namely we see a blue cube outdoor in a bright day and in the room by the light of a lamp also blue [1]. The Macbeth Color Checker appears a good choice after all, because there are color constancy algorithms, which are based on it [7]. If we detect any environment light change,

just pop up for a few second the Color Chart and run the algorithm to recalibrate the distortion parameters.

There are however some non-hardware problems which haven't been mentioned yet. As cleared in the introduction, the perception of people is limited. The definition of our eyes falls off in inverse proportion to distance. Furthermore the different background colors can emphasize or conceal an object, which phenomenon intensifies, if we see the projection screen far [1]. It seems a good idea, to invest the corrective algorithm with a function which is able to signal, if we want use too little font size in the presentation or the color combinations are not well chosen. The basis of this task can be leaded back to the retinex theory [8].

The biological background is the following. In the human eye there is an area on the retina, in which in case of getting light stimuli a particular ganglion cell changes its activity. This area is called the cells receptive field [11]. This structure can emphasize the significant points of the visual field, detect edges and provide constancy. Constancy means that the retina can ignore the change of the brilliance. There are cells with special receptive fields: single-opponent or double opponent cells. They play a big roll to detect color brightness and distinguish particular colors. If we are able to code some similar functionality to a program, then it could decide how much a given picture will be understandable for the human eye.

VIII. THE ALGORITHM UNDER CONSTRUCTION

We discussed what kind of tasks should perform the corrective algorithm. However it is necessary to distinguish the background and the projected picture on the records firstly, before any kind of processing. It may seem to be an easy problem with many solutions, but if we begin to resolve it, then it emerges to be not self-evident.

I tried different approaches to find the accurate projected field on the image. At first, I presumed that I can find the projected area according to its brightness. So I tried two different techniques, pixel value projection and histogram analysis, but none of them gave a fair result. Both algorithms can determine the corners of the projected area with some mistake, but when I used edge detection I got surprisingly better results, shown on figure 6.

I started the examination of the discussed algorithms, but unfortunately I couldn't bring them into a successful operable state yet, because of their complexity. In the next period, first of all I will implement and test these algorithms in my code.

IX. FURTHER GOALS

The perspective aim is that the finished algorithm should not run on a separate computer, but would be integrated into the projector.

There are projector-systems currently in some projects [9] and in the trade too [10] which work with camera. I presume that the camera and some operation unit will be an integral part of the projector sooner or later.

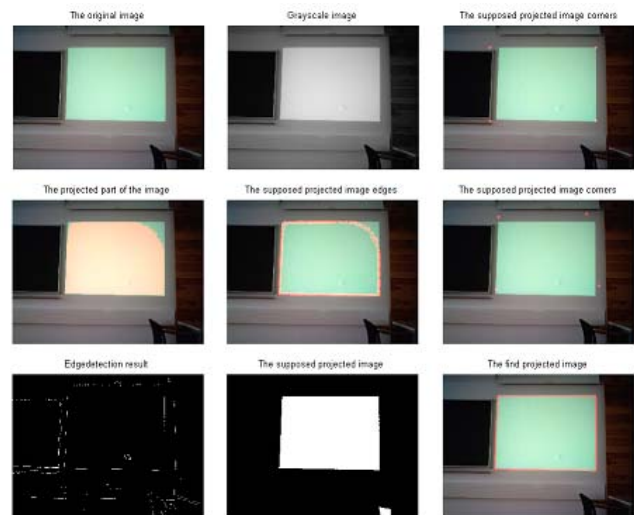


Figure 6. The results of the projected area search on an image. Every row shows a different technique. The first two methods, pixel value projection and histogram analysis, do not give a fair result. The last one, where I use edge detection, however finds the projected area very well.

Taking these into consideration, an adaptive algorithm running industrially in the projectors in the future is not a bold idea. A couple of pictures are acquired with the help of the inbuilt camera on the beginning of the projection; some calculations are done so these projectors project the obtained video signal with the suitable modifications. Thank to this we receive the possible most beautiful picture despite the different disturbing factors. It is conceivable, that the outlined problem will be simplified, because the inner parameters of the projector are available in this case.

REFERENCES

- [1] R. Sekuler, R. Blake: *Észlelés*, Osiris, 2004
- [2] Generally about the projectors
http://en.wikipedia.org/wiki/Video_projector
- [3] Texas Instruments' DLP technology
<http://www.dlp.com/technology/how-dlp-works>
- [4] M.-H. Lee, H. Park, J. Park "Fast Radiometric Compensation Accomplished by Eliminating Color Mixing Between Projector and Camera" *IEEE Transactions on Consumer Electronics*, Vol 54, No. 3, 2008
- [5] K. Fujii, M. D. Groosberg, S. K. Nayar "A projector-camera system with real-time photometric adaptation for dynamic environments" *IEEE Computer Society Conferene on Computer Vision and Pattern Recognition*, 2005
- [6] D. Pascale "RGB coordinates of the Macbeth ColorChecker" *The BabelColor Company*, 2006
- [7] P. V. Gehler, C. Rother "Bayesian Color Constancy Revisited" Microsoft Research, Cambridge
- [8] E. H. Land "The Retinex Theory of Color Vision" *Scientific American* December, 1977
- [9] R. Kjeldsen, C. Pinhanez, G. Pingali, J. Hartman "Interacting with Steerable Projected Displays" *5th International Conference on Automatic Face and Gesture Recognition*, Washington, 2002
- [10] Interactive projectors displays in commercial use
<http://www.touchmagix.com/index.html>
- [11] Semir Zeki, *A Vision of the Brain*, Blackwell scientific publications, Oxford

Sequence data mining

Kálmán Tornai

(Supervisor: Dr. János Levendovszky)

kami@digitus.itk.ppke.hu

Abstract—In several different case the task is to find patterns in vast data sequences, data sets. For example in biology the problem is to find patterns in a protein structure or a DNA sequence. The source of the input data could be log files from a web market, where the customers' action is being logged. Analyzing this logs more accurate marketing actions can be performed, new pieces of goods could be advised for the customer. From log file several exchanges systems' or informatics systems' function can be observed. After a failure or some event the prequel actions logged could be retrieved from logs and in the future the supervision of the event is predictable. In this study I will look into the first part of task: finding the sequential patterns efficiently, and speeding up the procedure exploiting the potential in parallel computing.

I. SEQUENCE DATA MINING

A. Task and definitions

To outline the task take the following example. There are four sequences, three of them followed by an event and the last one corresponds to normal function.

- A F G D S N G F D J Z A D G A – Event
- R T J E A U N T D J T A D G A – Event
- Q R L S N D J H I H G A – Event
- I O K H G A B Q N J D – Normal

The italic subsequences can be found in all of three sequences. Also you can notice that between *D J* and *G A* there are exactly the same length gap. Therefore not only the symbol sequence but the gap may be value information.

To understand the report some basic notion will be clarified: **Sequence** is a sorted list over τ type: $S = s_1, s_2, s_3, \dots, s_m$, where m is the length of sequence. The elements of sequence can be indexed.

$\alpha = \alpha_1, \alpha_2, \dots, \alpha_n$ is **subsequence** where $\beta = \beta_1, \beta_2, \dots, \beta_m$ is a sequence (or β is supersequence of α sequence), where $\exists j_1 \dots j_n \quad 1 \leq j_1 < j_2 < \dots < j_n$ and $\alpha_1 = \beta_{j_1}, \alpha_2 = \beta_{j_2}, \dots, \alpha_n = \beta_{j_n}$ holds. The notation is the following $\alpha \sqsubset \beta$.

Support is the measurement of fitness. It shows that how many supersequence exist in the database for one particular subsequence.

Patterns are subsequences, which support level is greater or equal than a predefined minimal support level. For example in DNA sequences the support level for one particular pattern is the number of sequences where the pattern occurs.

B. Basic algorithms

In this section two fundamental algorithm will be introduced.

1) *Apriori method – GSP algorithm*: This algorithm is based on a trivial fact: $S' \sqsubset S \implies \text{Support}(S) \leq \text{Support}(S')$. The support level of a subsequence is greater or equal than the support level of sequence.

Algorithm 1 GSP

```

 $\mathcal{F}\{1\} \leftarrow \{x \mid \text{support}(x) \geq \text{minsupport} \wedge x \in \Sigma\}$ 
 $n \leftarrow 1$ 
while  $\mathcal{F}\{n\} \neq \emptyset$  do
   $n \leftarrow n + 1$ 
   $\mathcal{F}\{n\} \leftarrow \{x \mid \text{support}(x) \geq \text{minsupport} \wedge$ 
     $x = x_1, \dots, x_n, x_{n+1} \wedge x_1, \dots, x_n \in \mathcal{F}\{n-1\}$ 
     $\wedge x_{n+1} \in \Sigma\}$ 
end while

```

The GSP algorithm is a bread-first searching algorithm. In the first step find all symbols which support level is big enough. Then extend all patterns all possible way. When in the searching three there is a branch which support level is not high enough, prune the branch. The iteration continues until is impossible to grow any pattern. The algorithm is faster when the pattern-enlarging part only uses frequent symbols to extend the patterns. One can easily see this is the slowest method to find all possible patterns in the input database.

The efficiency of this algorithm can be improved, it will be discussed later.

2) *Prefix method – PrefixSpan algorithm*: This algorithm also uses a pattern growing method to find patterns in the input database. The PrefixSpan method is based on the prefix property: Given $\alpha = e_1 e_2 \dots e_n$ and $\beta = e'_1 e'_2 \dots e'_m$ sequence, where the symbols in sequences are alphabetically ordered. β is prefix of α sequence, when $m < n, e_i = e'_i$ for all $i < m-1$, and $e'_m \subseteq e_m$ and $e_m - e'_m$ follows e_m in alphabetical order. (The notion is the intuitive prefix notion.)

Algorithm 2 PrefixSpan subroutine

```

Require:  $\alpha, l, D|_\alpha$ 
 $\mathcal{S} \leftarrow \{b \mid \text{support}(x) \geq \text{minsupport} \wedge D \in S|_\alpha\}$ 
   $b$  could be concatenated with  $\alpha$  pattern
 $\alpha'_i \leftarrow \alpha b \quad \forall b \in \mathcal{S}$ 
  Construct projected database for  $\alpha'_i$ , and call PrefixSpan.

```

Parameters are: α pattern, l is the length of pattern, and $D|_\alpha$ is the projected database for pattern α . (The projected database contains the sequences from the original database where the pattern exists.)

PrefixSpan is a depth-first searching algorithm. In the first step the algorithm finds the most frequent symbols. After the projected databases will be constructed for these symbols one by one. Each projection contains the subsequences (first iteration symbols) from the original database and the next symbol. (Therefore only the possible 2 length patterns will be constructed. Election of the patterns which are supported enough is the following step. This recursion continues until it is impossible to grow the pattern. To start PrefixSpan, call **PrefixSpan** ($\emptyset, 0, S$)

C. Improved versions of previous algorithms

In this section the improved version of GSP and PrefixSpan algorithms will be introduced. Many publications are related to this area, many different improvements (minor and major) are available to these algorithms, but in this report only the most common will be mentioned.

1) *SPADE – Improved version of GSP*: Firstly, trying to extend the patterns, SPADE will use only most frequent symbols instead of the method used in GSP. This will eliminate a bunch of branches in the searching tree, because many of obviously nonpattern sequence will not be generated.

Secondly – and this is the important part of SPADE – to extend a valid pattern SPADE uses the following rule: If exist two pattern which are the same but last symbol it is possible to mix new possible patterns. Such: if $p_i = sz_1$ and $p_j = sz_2$ then $p_{n+1} = sz_1z_2$ and $p_{n+2} = sz_2z_1$.

Algorithm 3 SPADE algorithm

```

 $\mathcal{F}\{1\} \leftarrow \{x \mid support(x) \geq minsupport \wedge x \in \Sigma\}$ 
 $n \leftarrow 1$ 
while  $\mathcal{F}\{n\} \neq \emptyset$  do
   $n \leftarrow n + 1$ 
   $\mathcal{F}\{n\} \leftarrow \{x \mid support(x) \geq minsupport \wedge$ 
     $(x = x_1, \dots, x_n, z_1, z_2 \vee x = x_1, \dots, x_n, z_1, z_2) \wedge$ 
     $\wedge x_1, \dots, x_n \in \mathcal{F}\{n-1\} \wedge$ 
     $z_i \in \{z_i \mid (p = s, z_1), (q = s, z_2); p, q \in \mathcal{F}\{n-1\}\}\}$ 
end while

```

2) *PrefixSpan – Bi-level projection*: The bi-level projection speeds up the projection the slowest part of the original PrefixSpan algorithm. Given α pattern and corresponding projected database ($S|_\alpha$). The algorithm determines the most frequent neighbouring symbol pairs. To do this, the projected database is being scanned and a matrix to count the occurrences of symbol pairs will be constructed. When the matrix is ready the longer patterns can be generated and the corresponding projections can be made. In case of inputs big enough the speed-up is remarkable due to eliminating a bunch of projection.

II. NUMERICAL RESULTS

All previous algorithm has been implemented in Matlab environment, using the potential of the Matlab language to construct as efficient code as possible. The result and the

Algorithm 4 PrefixSpan Bi-level subroutine

```

Require:  $\alpha, l, D|_\alpha$ 
 $S \leftarrow \{b \mid support(x) \geq minsupport \wedge b \in D|_\alpha\}$ 
Construct a P matrix sized  $n \times n$ ,  $n = \#(S)$  Elements are
the number of neighbouring symbols in current projected
database.
 $\mathcal{T} = \{(a, b) \mid ab \in D|_\alpha \wedge support(ab) \geq minsupport\}$ 
if  $\mathcal{T} = \emptyset$  then
   $\mathcal{T} = \{(a, \emptyset) \mid a \in D|_\alpha \wedge support(a) \geq minsupport\}$ 
end if
 $\alpha'_i \leftarrow \alpha, a, b \quad \forall (a, b) \in \mathcal{T}$ 
Construct projected database for  $\alpha'_i$ , and call PrefixSpan.

```

performance of the algorithms corresponds to the references. During the examinations the following aspects were taken.

- Runtime how depends on minimal support parameter.
- Runtime how depends on the size of the input.
- Memory usage how depends on the size of the input and the minimal support parameter.

All result is correlated with number of symbols, number of different symbols and the distribution function of symbols. The order of tested algorithm is the following (my results are the same as in the referenced literature.) PrefixSpan with bi-level projection; PrefixSpan; SPADE; GSP.

Following figure shows the comparison of tested algorithms using the same input and different minimal support values. The time axis is logarithmic.

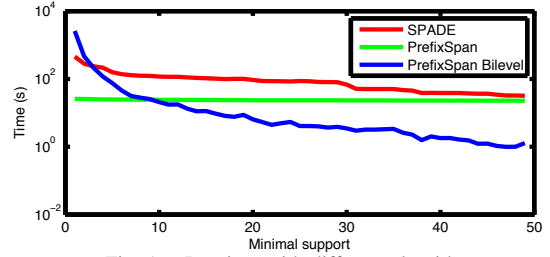


Fig. 1. Runtime with different algorithms

If the input size doubles the PrefixSpan algorithm solves the problem approximately double time. This gave the basic idea of parallelization. Following figure shows the runtime of PrefixSpan bi-level algorithm in function of input size.

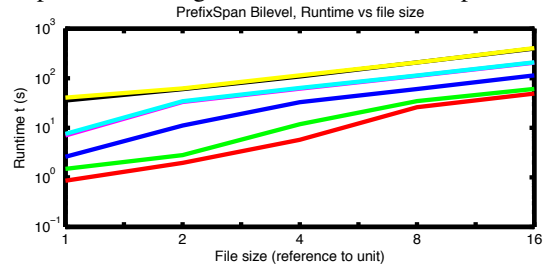


Fig. 2. Runtime for different input size

III. IMPROVEMENTS FOR PREFIXSPAN BI-LEVEL

A. Conceptions

Several considerations could be taken into account, for example:

- Minimizing processor time
- Minimizing memory usage, therefore bigger input could be processed
- Minimizing disk IO operations. If the input is located in a database and could not be loaded to main memory this might be the first issue.

These goals are obviously controversial. In general using less memory implies more processor time. Minimizing disk IO could lead to loading the whole input into the main memory implying a huge memory utilization.

Saving processor time could be done with bi-level projection or pseudo projection. (It is not discussed in this report.) More restriction from the task side (maximal and minimal pattern length, continuous patterns) might also save processor time.

Another possibility is to run the searching procedure parallel. In this case the branches in the searching tree are not necessarily equalized, therefore some parallel task may run longer than the others. So the algorithm cannot be scaled well.

To reduce memory usage and number of IO operations the following trade-off solution can be done. The input is preprocessed and the symbols are substituted with numbers. This dictionary will help to not store long symbols (character string for example) several times but only one. And the operations are faster because the numerical comparators are faster than comparators based on characters.

For the input file a backbone database is generated. This is a symbol database, where the occurrences of symbols are stored. This will help to construct the projected database faster, and provide the support information much faster than counting again and again. Also a table is constructed to represent the input data. It stores the symbol's sequences, and the projection is based on this table. The projection database is virtual, only a mask exists for the original database eliminating redundant information to reduce memory usage.

This design step implies that both the symbol database and backbone database must be constructed before starting the real pattern searching. Naturally construction takes time, but almost every case the gain is more than the construction time.

B. Parallelism with input splitting

The time cost of the PrefixSpan algorithm doubles when the input data size doubles. This observation led to an idea: split the input and run parallel the PrefixSpan algorithm. The split should be done very simply to get approximately equally sized inputs for sub-tasks. The minimal support parameter should be also reduced, the simplest way is to divide it with the number of the parallel threads. It is being assumed that the patterns and symbols in input database are equally distributed. In some cases it causes a problem: if the symbols are not equally distributed then it is possible to have one input slice supporting much more a pattern than the others. This may cause that in only one thread may find the pattern, and during a merging procedure this pattern will be omitted. To solve the problem a more complicated algorithm has to be used.

This parallel algorithm has been implemented on a common multi-core system (Intel Core2). The figure following shows

the speed-up running the task parallel on four cores. The theoretical limit cannot be reached because the synchronization between the threads and the thread management slows down the process.

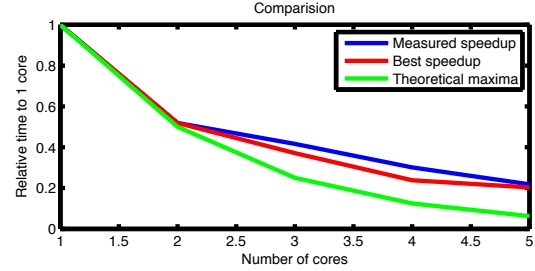


Fig. 3. Speed reduction

The table also shows the runtime values.

Threads	1	2	4	8	16
Time	122,615	63,579	45,499	29,216	24,873
Ratio	1	0,519	0,371	0,238	0,203

Fig. 4. Speed reduction – values

The main drawback is that after the separate thread results the patterns must be combined in order to get accurate result and it is impossible to reach the theoretical result and speed-up.

C. Real data validation

On real DNA data the parallelized PrefixSpan bi-level algorithm performed as it can be seen on the next figure. The theoretical limit cannot be reached, and after a certain amount of used cores the consumed time getting more and more.

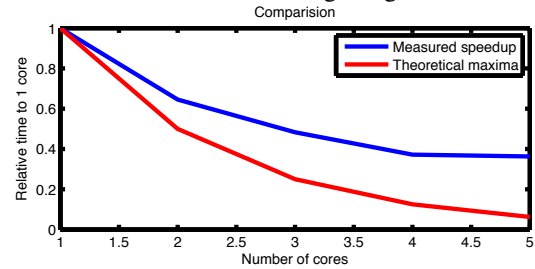


Fig. 5. Speed reduction on real-world data

IV. NOVEL ALGORITHM ON PARALLEL ARCHITECTURE

In this last section the conception of a new algorithm will be introduced. The design considers the parallelism as the most important property and tries to highly utilize the available many-core hardware.

A. Concept

The symbol sequences can be treated as sequence of numbers as written previously. These sequences can be propagated through a pipeline of cells, which cells count symbol occurrences in the sequence. One cell depending on implementation could take care of one symbol only or multiple comparison could be performed in one cell. The sketch of these pipelines is on the following figure.

In the first round the most frequent symbols can be determined. After the content of cells must be redefined. All

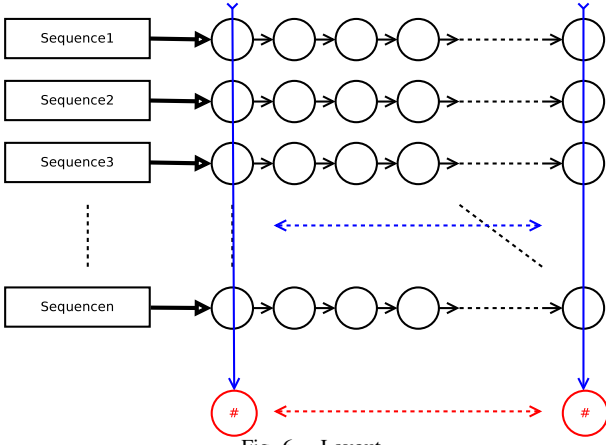


Fig. 6. Layout

possible two-length pattern should be generated and the cells in the second run counts the support level of these two-length patterns. This is repeated till the patterns cannot be extended.

After each run the non-frequent symbols can be omitted from patten length extension, therefore less cell is being used.

The algorithm based on the SPADE algorithm's idea, the a priori property: $\forall s = s_1, s_2, s_2, \dots, s_n \Rightarrow support(s_i) \geq support(s) \forall i$

B. Theoretical running time

For each run the time is determined by the longest sequences and the number of sequential cells.

$$T^{(k)} = \max_i length(sequence_i) + c^{(k)} - 1$$

The election and redefine operations also takes time and it is important that the number of cells varies between the runs.

Algorithm 5 Novel Sequential Data Mining

```

 $\mathcal{P}_0 \leftarrow \emptyset$ 
 $\mathcal{S} \leftarrow$  symbols in input data set
 $n \leftarrow 1$ 
while  $\mathcal{P}_n \neq \emptyset$  do
   $\mathcal{C}_n \leftarrow \{C_{n-1}^i \times \mathcal{S} \ \forall i\}$ 
  Propagate sequences over cells, to count occurrences
  ( $c^i = support(C_n^i) \ \forall i$  and  $s^i = support(\mathcal{S}^i) \ \forall i$ )
   $\mathcal{P}_n \leftarrow \mathcal{C}_n \leftarrow \{C_n^i \mid C_n^i \in \mathcal{C}_n \wedge c^i \geq minsupport \ \forall i\}$ 
   $\mathcal{S} \leftarrow \{S^i \in \mathcal{S} \wedge s^i \geq minsupport \ \forall i\}$ 
   $n \leftarrow n + 1$ 
end while

```

C. Properties of the algorithm

The new algorithm is being compared to the existing solutions in this section

Advantages of the algorithm:

- Due to big-scale parallelism the support level calculation happens on several sequences, and all pattern candidate are being checked, so the number of iteration is much more less.
- The general idea could be implemented on more architecture (FPGA, GPGPU, Cell), to utilize the many-core supported advantages

- The local connections between cell could be utilized with proper layout. For example if a pattern's last couple of symbols is prefix of another pattern the cells could be connected to each other to minimize the number of comparisons. (Also this is a more complex problem.)

Disadvantages of the algorithm:

- The non trivial re-configuring task cannot be done parallel well, and takes noticeable amount of time.
- The hardware has physical constraints therefore the number of cells is limited. If the input is big enough multiple sweeps must be performed in one run. This multiplicity enlarges the complexity of reconfiguring task and the whole algorithm.
- If there is not enough cell one cell can make more complex and slower computation or it is possible to do some sequential data processing in one cell, but this may cut back the advantage gained on parallel processing.

V. SUMMARY AND FUTURE WORK

In this report the sequential data mining problem and some common basic solution has been introduced. The second part of the report focuses on the possible improvements on the existing algorithms partly practical improvements. The last section introduces a new aspect and algorithm do solve the initial problem on many-core hardware with highly parallelism.

In the near future I am going to implement this algorithm on GPGPU. A brief study is ready about the architecture. Also I would like to outline other implementation method on FPGA, and lay the general implementation rules down.

The FPGA implementation is very simple and might be utilize the hardware best due to the simplest processing units can be in a cell: a comparator and an accumulator register. The major drawback is the reconfiguration phase. It is very hard to do it efficiently on-line.

The GPGPU, for example nVidia Fermi architecture forces some important decisions, how to design the kernel functions and iteration steps.

REFERENCES

- [1] Guozhu Dong, Jian Pei Sequence Data Mining, Springer, 2007
- [2] Toshihide Sutou, Keiichi Tamura, Yasuma Mori, and Hajime Kitakami Design and Implementation of Parallel Modified PrefixSpan Method, A. Veidenbaum et al. (Eds.): ISHPC 2003, LNCS 2858, pp. 412-3422, 2003; Springer-Verlag Berlin Heidelberg; 2003
- [3] Makoto Takaki Keiichi Tamura, Hajime Kitakami Dynamic Load Balancing Technique for Modified PrefixSpan on a Grid Environment with Distributed Worker Model, Hiroshima City University;
- [4] Rakesh Agrawal, John C. Shafer Mining Sequential Patterns: Generalization and Performance Improvements, Research Report RJ 9994, IBM, Almaden Research Center, San Jose, California, December 1995;
- [5] Jian Pei, Jiawei Han, Behzad Mortazavi-Asl, Helen Pinto, Qiming Chen, Umeshwar Dayal, Mei-Chun Hsu PrefixSpan: Mining Sequential Patterns Efficiently by Prefix-Projected Pattern Growth, Simon Fraser University Burnaby;
- [6] Dhany Saputra, Dayang R. A. Rambli, Oi Mean Foong Mining Sequential Patterns Using I-PrefixSpan, International Journal of Computer Science and Engineering; 2008
- [7] J. Pei, et al Mining Sequential Patterns by Pattern Growth: The PrefixSpan Approach, IEEE Transaction on Knowledge and Data Engineering, vol. 16, no. 11, pp.14240-1440, Nov. 2004.

Towards moving platform video processing Object detection on many core architectures

Tamás Fülöp
(Supervisor: Dr. Ákos Zarándy)
fulta@digitus.itk.ppke.hu

Abstract — This paper is a step to object detection on many core architectures. These novel systems are very important nowadays, and give us new opportunities to invent new programming approaches and solve the most complex problems. The report starts with a few thoughts about many core architectures. After it provides a short summary about the problem of object detection then it suggests possible algorithmic ways to solve it. Finally, it gives ideas about pedestrian detection.

Index Terms — Many core architectures, image segmentation, object recognition, pedestrian detection

I. INTRODUCTION

The state-of-the-art processors are highly parallel systems. The big chip producers and research institutes develop new and more complex parallel architectures. The classical algorithms cannot run efficiently on these kilo processor chips without optimization, or without new algorithmic thinking idea. The Virtual Machine concept [1] is a possible way to use kilo processor chips more efficiently. This concept predicts platform independent algorithms. Using this concept we need to reorganize the algorithmic tasks and thinking in real parallel systems. The “parallel thinking” seems easy but sometimes it needs to leave the known programming paradigm. The local connections – Precedence of Locality – are the most important idea in this paradigm (like in Cellular Nonlinear Network). My challenge is thinking about it as image segmentation.

First, I would like to show few important things about segmentation problems. Afterwards, I would show a specialization as problem of object detection such as pedestrian detection. At last, the looming object detection's new results will be introduced.

II. OBJECT DETECTION

Object segmentation is the key initial step of practically all security-surveillance and other monitoring algorithms. Many foreground-background separation algorithms exist; there are two classes of these algorithms. The first assumes stationary camera platform, while the second considers moving camera platform.

A. Problem of object detection on stationary camera platform

The stationary camera platform has extensive literature. These algorithms usually use a reference image, when no moving objects are in the scene. It needs stable camera,

constant illumination conditions and a good background model. The algorithms are usually simple subtract the captured image from the background image. Where the result is not zero, there is an object. The problems of these algorithms are that the methods depend on the illumination, and they have many adaptation problems.

The reason of this failure is that the background is slowly changing on the pixel level due to the changing of illuminations (sun shines from different angle, shadows are moving, clouds are coming and going, etc), which accumulates significant changes over the time.

The gradient based algorithms are illumination independent, because these algorithms are based on image structure, like variance, color ratio between two pixels, or brightness difference of two pixels. I have already implemented an algorithm on many core architectures which is based on local pixel relation [2].

All of these algorithms need to update the reference image continuously to introduce the latest “slow” changes. This induces the stability and plasticity dilemma, because quick update leads to losing slow moving objects, while slow update cannot follow the illumination or other drastic background changes.

Other problems, usually most of the update algorithms use difficult statistical methods to predict the background; hence these use huge memory to store statistical information for calculations.

B. Problem of object detection on moving platform

We have seen that the object detection is a very difficult problem even in the stationary camera platform. What can we do with a moving platform? How can we approach this problem? Have we got any information about the environment, about scene?

The approach is platform dependent, because we should use different method in different environment like camera on a car, camera on a plane, camera on a pole, etc. My study belongs to the first one.

The first question in moving platform detection is what is an object? We should make a decision about it, so it is a recognition problem now.

We can make a background model, like the stationary camera model, and we can calculate with camera matrix. If we have any additional information about the camera (position, angle, etc.) we can calculate easier. The movement can be looming, shrinking, lateral or rotation motion. Calculating lateral or rotation motion is easy, because we can use image corresponding techniques, and the objects are invariant in rotation and offset but not in ratio.

When a camera goes ahead, it makes parallax, which cause optical flow too: objects appear or disappear, it seems that the relative position changes. It changes the scene, and the background. If we know the position of the camera, we can calculate the relative position of objects but first we need to detect the objects. If the camera is moving in very difficult environment (like a city), the algorithms need to be more precise. This way it is not working as perfect as we want, or we need more complex algorithm and hard calculations.[7] Figure 1. shows, how the optical flow works, when a car goes ahead. Colors encode length of flow vectors, show the consequence of motion parallax. The warmer the color encode, the longer the motion.



Figure 1. Visualization of considered optical flow fields. Warmer colors encode longer flow vectors.[7]

Another way, when we try to find objects on the picture with their properties (like color, brightness (shadow or not?), texture deviation, etc.). Thus we make correspondences between two pictures. How can we achieve good segmentation? What is “an object” if the segmentation is found as different?

The second problem is the correspond because it is difficult, if we have many possible objects.

C. Problem of object detection with stereo camera system

Using two or more cameras can be more efficient because two pictures can give more information about environment and parallax. This method is using complex calculation between two cameras. We need to find special properties that correspond between the pictures. Many problems are similar to those discussed in section A and B.

III. TYPICAL OBJECT DETECTION AND RECOGNITION APPROACHES

We have seen that simple object detection is not an easy problem. We need to specialize the paradigm: finding well defined objects.

We have an image and the algorithms only search well specified objects. This seems easier as we have an object model, and we make a simple search on the picture. However, it has got two problems yet:

- Large computational complexity occurs when we search objects. An object can be occurring in different form (small or large, hiding some of the

parts, etc.)

- Algorithms need well defined object model. How can we make these models? How can we write their properties? What properties are important?

Now, I would like to show how these algorithms work on pedestrian detection. Many pedestrian detection projects have been existing all over the world. Many of these projects have been supported by the European Union, or the Automotive Industry (PROTECTOR, SAVE-U, WATCH-OVER, ACTIVE-SFR). I have found many articles about the pedestrian detection problem [3][4][7]-[12] but all of these suggest to me that the pedestrian detection is an open question. The algorithms are not enough reliable yet, or they are constrained in some sense (many of them non real-time, need special accessories, etc.).

Detection and recognition algorithms have two steps:

- Region of Interest selection (ROI) as an hypothesis generation
- Classification as a model matching

A. ROI selection

The ROI selection is the first step of detection algorithms. The algorithm gets an image (or its transformation) and localizes those regions which contain possible interesting objects.

Interesting region can be found when the algorithm searches special parameterized properties (shapes in pictures (circle, square, and triangle)) or non parameterized properties (pedestrian outlines). Non parameterized outline needs much observation, how a pedestrian is built up, what is the possible position of a pedestrian.

The simplest technique to obtain initial object location is the hypotheses of the sliding window technique. The detector windows are shifted over the images. The computational costs are high for real time processing. [3] It has some constraints: some objects can be found in special environment (or never). For example a camera will never look for pedestrian in the sky. Objects can be pedestrians if we found them on the road (gray or dark background as asphalt).

Other techniques are based on extra information: stereo vision or difference between two pictures.

B. Classification

We should classify the ROI selected part. The generative classification models are based on shape or texture recognition. The discriminative models used statistical approaches to approximate the decision by learning parameters of discriminate functions. [3][9]

1) Generative models

The appearance of the pedestrian class in terms of its class conditional density function. Shape models not the most attractive models because these models contain reduced variations of pedestrian appearance. These models give a probability for the pedestrian class.

Shape model is a 2D or 3D human model with plausible positions. Such models require large corpus of example

shapes with their transformations. Moreover, efficient matching techniques able to use it, which based on distance transforms. Some shape models have been combined with texture information for iterative error minimization.

2) Discriminative models

Discriminative models are approximated the Bayesian maximum-a-posteriori decision by learning the parameters of a discriminate function between the pedestrian and non-pedestrian classes from training examples. The training features usually the pixel intensities, intensity changes or other extracted properties.

These two models can be combined as it is demonstrated [8] below, and shown in Figure 2.[8]

These two models working with manually designed feature extraction (Feature dictionary that represents local intensity differences at various locations) or automatic feature extractions. The automatic extraction should be working as the human visual cortex or it should use adaptive techniques (AdaBoost). The feature vector based methods are easily implementable. It uses a codebook, which contains a body of a person as set of vectors.

In other case, Support Vector Machines, Neural Networks with local receptive field can be used for classification [3]. An interesting study classify pedestrians with their walking rhythm [10].

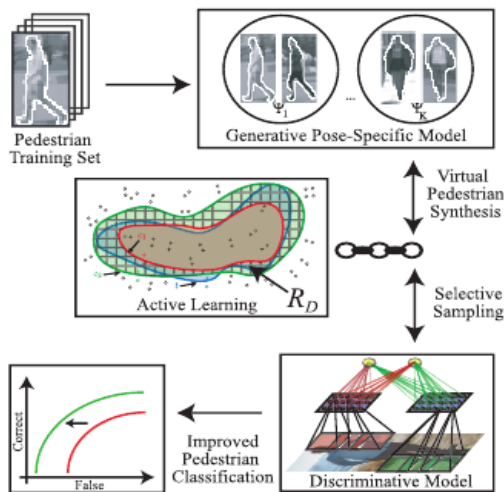


Figure 2. The top of the figure shows the generative model, the bottom of the picture shows the discriminative model. Some article combine these as learning.[8]

IV. APPROACHES OF PEDESTRIAN DETECTION

The presented algorithms are working but most of them are not enough reliable (60-80%), and does not work fast enough to make decisions (7-23 fps).[4] The biggest problem of these algorithms are the large computation complexity, which needs optimization.

Some kilo processor chips are fast enough and many of them use only a few Watts (such as Eye-RIS 1,5W>). We need to search new algorithms to use these chips efficiently, so we need to reorganize the problems for this new architecture.

I was looking for new ways on solving this problem effectively, even in kilo processor chips. I did not want to

change the basis of the recognition algorithm (first: ROI selection, afterwards the classification).

I tried three ways for ROI selections:

- Edge detection algorithms then using any transformations
- Edge detection followed by shape metrics for corresponding
- Segmentation with fundamental image properties (color, texture) then using shape metrics for corresponding

1) Edge detection algorithms then using any transformations

It is a trivial attempt. Many edge detection algorithms exist. These algorithms are available on many core architectures, and based on local connections.

I tried different transformation methods, such as the Hough transformation (Figure 3.), which is used for pattern recognition. In simple environment it works well. This method in more complex environment (strongly textured images) could not be used. The edge detection algorithm finds many edges. Probably a better configurable algorithm finds less edge that makes the algorithm more usable.

Another problem, it requires a good model for comparison. The Fourier Transformation and the Laplace Transformation are unsuccessful.

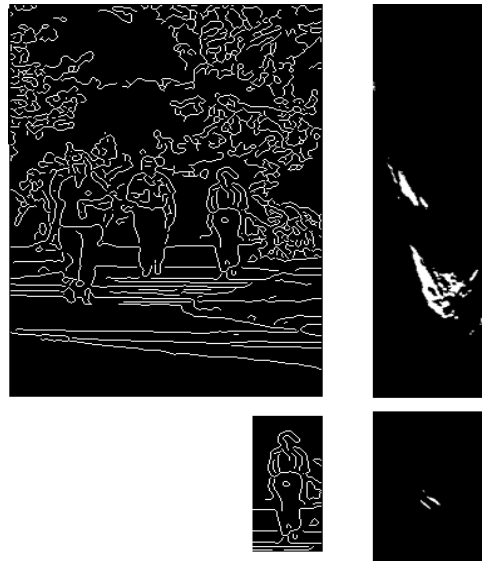


Figure 3. The Hough transformation. Top left is a picture after edge detection, top right is its Hough transformation. Bottom pictures have a person and its Hough transformation.

2) Edge detection followed by shape metrics for corresponding

In this case, the algorithm searches some special shapes, like an edge of a head. The problems with this algorithm are the highly textured images (city, leaves on a tree) and the size depending on the distance of the pedestrian. This method can be combined with other methods to achieve better result, because it is quite simple. Other parts of the human body is not good enough to search body, because the leg can be hidden behind a postbox or other objects, and other parts of the body are not reliable enough for recognition if we use their edges only. Some methods search for the shape of the whole body. These shapes can be

arranged in a hierarchy as shown by Figure 4. This hierarchy is good for better and faster classification.

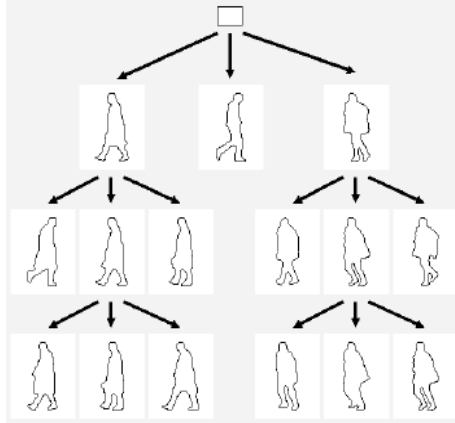


Figure 4. Hierarchy of pedestrian shapes [12]

3) Segmentation with fundamental image properties then using shape metrics for corresponding

Firstly, I would like to show that, how we can extract information. The most trivial information is the color of pixel. Color based shape detection is not a trivial problem because the surface of objects are usually not homogeneous. The color transformation is a trivial possibility to extract information. The HSV or HSL color space contains very useful information about objects. Hue channel determines the color of surface, so we can detect the surface with the same colors. The Saturation and Value (or Light) channels provide useful information. In case of a colorful surface we might get the same reflection or light intensity from a place although we might receive high variance result on the Hue channels.

Other important property is the image texture. What is the deviation, and how it can be defined. Applying the Markovian Random Field model (MRF) is a possibility for color images too. [5] The grayscale MRF does exist for the CNN [6], so it has some literature and algorithm for it.

The Active Contour detection is a promising possibility, but presently it is not good enough for pedestrian segmentation because it has got many constraints for objects.

V. CONCLUSION AND FUTURE WORK

This work is a short survey and experimental results. I have examined and implemented various algorithms to ROI selection, because it's very important to understand, how these methods work. What are the bottlenecks? Very interesting question is the application of these algorithms on many core architectures? If it is not possible then how can we do a good object recognition algorithm in many core architectures?

I have looked that, some of them are usable on these systems, like MRF model, but these need to expand their opportunities. Some algorithms need to be reorganized or rethought. I have started my experiments with stationary camera model but I have not calculated background model. Perhaps it is enough for ROI selection. The classification will be used for the motion information.

I have not got any opportunity to work on classification yet. The reason is that the classification is based on a good

ROI selection.

I am determined to create a good ROI selection algorithm that would be implementable on many core architectures using these experimental results. Afterwards I will need to work on classification methods.

ACKNOWLEDGMENT

The author expresses his thanks to Furukawa Electric Institute of Technology for all its support. The author is also grateful to Ákos Zarándy for the discussions and his suggestions.

REFERENCES

- [1] T. Roska, "Virtual and physical cellular machines", NSF/MIND Workshop on Architectures for Post-CMOS Switches, Notre Dame 2009
- [2] T. Fülöp, Á. Zarándy, *Real-time moving object segmentation algorithm implemented on the Eye-RIS focal plane sensor-processor system*, NOLTA 2008 Conference, Budapest
- [3] M. Enzweiler, D. M. Gavril, *Monocular Pedestrian Detection: Survey and Experiments*, IEEE Trans. On Pattern Analysis and Machine Intelligence, Volume 31, No 12, Dec 2009
- [4] D. M. Gavril, *Multi-cue Pedestrian Detection and Tracking from a Moving Vehicle*, International Journal of Computer Vision, Vol 73, 2007
- [5] Z. Kató, T. Pong, *A Markov random field image segmentation model for color textured images*, Image and Vision Computing Vol 24, 2006
- [6] T. Szirányi, J. Zerubia, *Markov Random Field Image Segmentation Using Cellular Neural Network*, Circuits and Systems I: Fundamental Theory and Applications, Vol 44. Jan. 1997
- [7] M. Enzweiler, P. Kanter, D. M. Gavril, *Monocular Pedestrian Recognition Using Motion Parallax*, Intelligent Vehicles Symposium Eindhoven 2008
- [8] M. Enzweiler, D. M. Gavril, *A Mixed Generative-Discriminative Framework for Pedestrian Classification*
- [9] S. Munder, D. M. Gavril, *An Experimental Study on Pedestrian Classification*, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol 28. No. 11, Nov. 2006
- [10] C. Pai, H. Tyan, Y. Liang, H. Mark Liao, S. Chen, *Pedestrian detection and tracking at crossroads*, The Journal of The Pattern Recognition, Vol. 37, 2004
- [11] C. Curio, J. Edelbrunner, T. Kalinke, C. Tzomakas, Werner von Seelen, *Walking Pedestrian Recognition*, IEEE Transactions on Intelligent Transportation Systems, Vol. 01, Sep. 2000
- [12] D. M. Gavril, *Pedestrian Detection from a Moving Vehicle*, Proceedings of the European Conference on Computer Vision, Dublin, 2000.

Dynamic Object Detection in Urban Environment

Mihály Radványi
(Supervisor: Dr. Kristóf Karacs)
radmige@itk.ppke.hu

Abstract— In this paper I present an algorithm that help blind and visually impaired people to navigate through urban environments by detecting and recognizing pedestrian crosswalks. Not only the presence of crosswalk is important, but the orientation and position related to the subject, to be able to help them approaching and passing through intersections. A few modifications of the original method [1] and new ideas will be discussed in this paper, regarding to color filtering, position and orientation estimation. The key-framing method will be introduced which results speedup and gives the possibility to run different recognition tasks parallel on the same video flow.

Index Terms— bionic, CNN, crosswalk, eyeglass, mean shift, orientation, position

I. INTRODUCTION

The Bionic Eyeglass [2],[3] is a portable device recently proposed to aid blind and visually impaired people in everyday navigation, orientation, and recognition tasks that require visual input. Although accessible pedestrian signals are more and more frequent at crosswalks in busy intersections, visually impaired people will not be able to get about independently until the infrastructure reaches full coverage on the routes they are using. Deciding if a crosswalk is present in the neighborhood of the user is one of the few important tasks we have identified based on the feedback from potential users.

In our previous works we have already introduced the concept and the prototype of the Bionic Eyeglass [4] which, to provide enough computing power was built using the Bi-i visual computer [5] as its main computational platform, that is based on the Cellular Neural/Nonlinear Network – Universal Machine (CNN-UM) [6],[7] and the underlying Cellular Wave Computing principle.

My experiments showed great results in detecting and recognizing pedestrian crosswalks. One of the most important factors causing some of the false results in my previous experiments [1] was the improper identification of the road surface area. In this paper I present a more efficient method for estimating road surface areas through advanced color segmentation steps, using the mean shift method.

Beside the improvements of the recognition and detection part, new methods were needed, to be able to estimate the orientation, position and direction of the crosswalk, thus getting a fully functional system that navigates and helps the user to reach the crosswalk and to get through intersections. A few results will be discussed in this paper.

However, by processing an image flow frame by frame robust decisions can be achieved, in cases when there is strong correlation between frames – e.g. on video flow the frames

following each other are similar – running the whole algorithm for each frame is a waste of time and processing power. A few assumptions on speedup can be done by selecting some candidate key-frames that are fully processed, and than for the following frames only a few estimations are done. This could easily led and help us to run different recognition tasks parallel on the same image flow. Some basic ideas of that will be discussed in this paper.

The paper is structured as follows: in Section 2 the improved crosswalk detection is discussed; Section 3 describes the position and orientation estimating task and than introduces the candidate key-framing method; Section 4 shows the results; Section 5 concludes the paper.

II. CROSSWALK DETECTION

A. Detecting the surface of the road

As already shown in previous papers [1], my main goal was to get the clearest crosswalk marks out of the input images, this way I could design a reliable decision function, which properly describes the properties of the detected objects and gives me a straight output. I start out with a color and a grayscale representation of the input. The frames are processed through a CNN algorithm that has two parallel processing threads, both based on different properties of the certain situation.

One uses advanced feature classification, with the goal of masking out those pixels in the scene, which does not belong to the road surface (referred to as *background*), i.e., constructing a binary mask of the asphalt and the signs printed on it, including the stripes of the crosswalk (referred to as *foreground*). This step is carried out through the mean shift clustering method [8], a procedure based on kernel density estimation [9]. The main aspect concerning this segmentation was, to exploit such similarity properties of a given scene, which cluster the asphalt and its overlay signs into the same class.

The advantage of the procedure is that it is feature space independent, and practically has one parameter: the kernel bandwidth of the used Gaussian weighting function. A well known weakness of the mean shift algorithm is that the selection of this parameter is not straightforward [10], furthermore it has to be selected carefully to avoid possible over-, or under segmentation.

Finally the remaining mask is labeled, and the largest region is selected as the foreground.

B. Extracting zebra candidates

We use this mask to obtain the grayscale values of the foreground area. Its brighter regions correspond to the painted crosswalk stripes, which are extracted by binarizing the image with an adaptive threshold based on the grayscale histogram.

The other main branch begins with the calculation of an adaptive threshold value from the bottom half of the grayscale input frame, assuming that the horizon is in the upper half of it, and the crosswalk lies below. After binarizing the image with this value, on both branches I carry out a series of template instructions [11]. When the two flows are joined together, their results are summed up by using a logical AND function giving me the so called candidate stripes. At this point I have those objects that could fit both the color and contrast criterions.

C. Space variant filtering

In many cases at this point there are still a few false objects detected usually related to brighter areas of the road surface due to lighting conditions. To avoid the problems that could be caused by these objects, size filtering is carried out with a newly introduced space-variant thresholding. Considering the perspective distortion of the crosswalks it can be stated that the further the stripe from the image plane, the smaller it appears, however the false objects above them are usually bigger. First, diffusion is used in order to get the area of the alternating black and white zebra stripes blurred together resulting on a middle gray patch. Meanwhile a much darker areas appear at the location of the patches not belonging to the crosswalk area due to their bigger vertical extension. The closest stripes can still behave like these objects, resulting on dark patches after diffusion. I introduced a space-variant thresholding scheme. The upper the object in the image, the lower the threshold for potential removal.

D. Decision

After all these processing and filtering tasks are done a VCCD [11] template is applied on the edge map of the output image. At the end a decision function is used by fitting different sized rectangles on the VCCD image and choosing the one with the biggest area. By using the area (A) and the height (h) of it, and evaluating a two dimensional sigmoid type function μ , a proper confidence value, the so called “zebra value” can be calculated:

$$\mu(h, A) = 2 - \frac{2}{1 + e^{-((A')^2 + (h')^2)}} \quad (1)$$

$$h' = \max\left(2 - \frac{h}{h^*}, 0\right) \quad (2)$$

$$A' = \max\left(2 - \frac{A}{A^*}, 0\right) \quad (3)$$

where h^* and A^* being the threshold values for the two variables. By increasing this threshold, we can decrease the number of false positive results; thereby increase the robustness of the algorithm. The obtained confidence value is

used to make the decision on whether a crosswalk is present in the input frame. See Figure 6. for examples.

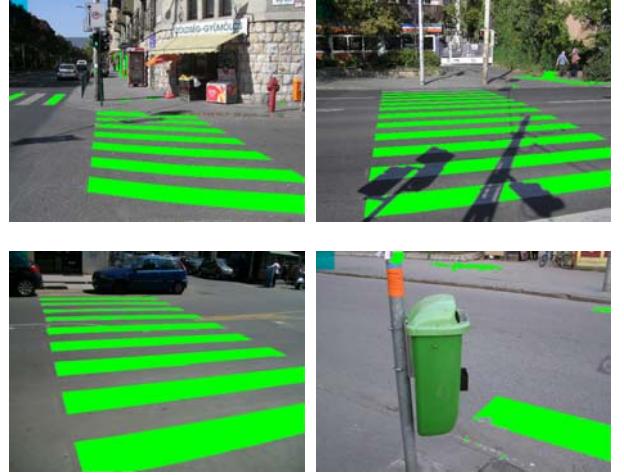


Figure 1. Examples of present and found zebra objects drawn onto the input images.

III. TOWARDS NAVIGATION

Recognizing crosswalks and establishing the presence of them is important, but in order to navigate blind and visually impaired people through intersections, we have to go further. The previously described method gives a confidence value whether there lies a crosswalk in front of the user, but does not contain any additional information of the position and orientation of it. When talking about position I mean the position on the frame where the crosswalk appears (left, right, middle) and the distance from the user (up – further, down - closer). Orientation is closest to the meaning of being in direction. These two things are very important to know, otherwise it cannot be guaranteed that the user will stay on the crosswalk by going simply towards.

A. Estimating position

Based on the detected crosswalk signs there is an obvious solution for estimating the position by simply calculating the extremums of the coordinates, and drawing a bounding rectangular on the crosswalk stripes. See Figure 5. for some examples. Due to the effect of noise, this method can easily overestimate the position and the area of the crosswalk, so a more accurate solution will be introduced.

The main idea is to calculate the center of mass of the crosswalks. For that, a short CNN algorithm is developed. As one can realize, on the output of the detection and recognition part, the crosswalk does not appear as one, connected object, but as independent stripes. The first thing we have to do is to combine them into one huge object. This was done by putting three CNN templates in a series. At first vertical shadow [11] was applied upwards and downwards separately, as it can be seen on Figure 2. Than the two shadowed images are logically ANDed which perfectly gives back the area of the crosswalk. The center of mass than can easily be calculated, and checked in which area of the image does it fall into.

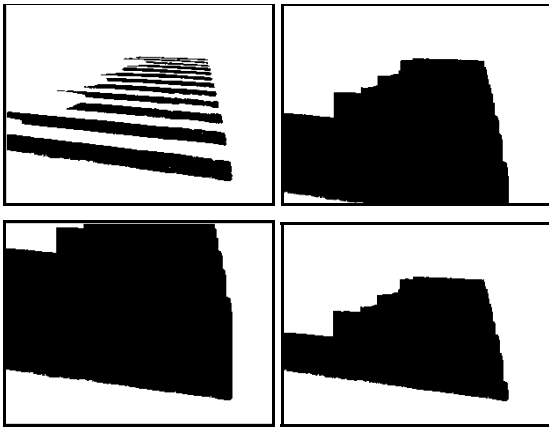


Figure 2. Upper line: Input image and vertical shadow downwards, bottom line: vertical shadow upwards, LOGAND of the two shadowed images

B. Estimating orientation

As one can see on Figure 3. my idea was to fit an arrow on each crosswalk sign, that shows the proper direction for passing by. The arrow lies on the median – line that is located on half angle (blue line) between the two side lines (green lines). To calculate the correct position, we have to find the four corner points of the crosswalk (shown in red) and than to calculate the side lines, and the median.

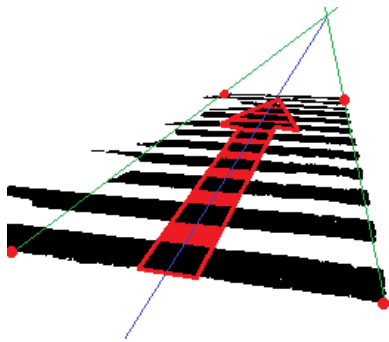


Figure 3. The basic idea of establishing the direction of the crosswalk

In order to find the corner points – thus the orientation – of the present crosswalk, I fit two lines on each frame, one onto the bottommost and one onto the uppermost stripe. At the intersection of the crosswalk stripes and lines there lies the corner points.

In practice the best way is to analyze the closest (base) line first. In many cases, especially when the current frame was taken from a close point of view – so the user is actually passing through the crosswalk – the closest stripes are crossing the image boundary, and the rest appearing on the frame is too short. Based on this observation these stripes are removed from further processing. At first the images are down sampled, and the coordinates of the bottommost black pixels are collected. With minimal number of outliers a line is fit on these coordinates by calculating the slope from the modus of the coordinate differences.

After having the bottommost line, one would easily draw a parallel line, shifted up as an estimation for the uppermost slope. Due to the perspective distortion of the crosswalks, it would be wrong, as it can be seen on Figure 4. with red. A better solution is to find one point of the uppermost stripe, than RECALL the whole stripe and fit a line onto it (green one). Figure 4. shows that these estimations are accurate enough. The corner points can easily be calculated as the intersections of the fitted lines and stripes, thus the orientation of the crosswalk can be found.

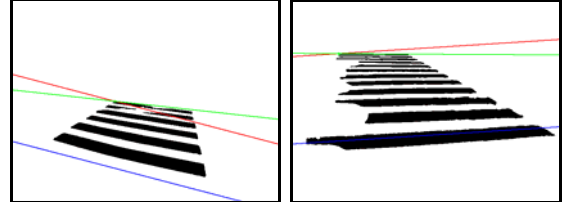


Figure 4. Examples of orientation estimation. Blue: base line; red: shifted, parallel to base line, green: line calculated by RECALLing uppermost stripe

C. Key framing

However, processing each frame with the full crosswalk detection algorithm gives robust results, running the whole algorithm each time is a waste of time and processing power, due to the high correlation and similarity between the frames following each other. A few useful assumptions can be done, like selecting some key-frames that are to be fully processed, and than for the following ones, just quick calculations and confirmations are to be done.

As in Section III/A I have already mentioned, if a crosswalk is found, a bounding rectangular can easily be drawn. My candidate key-framing method is based on the following. If a crosswalk is found, and the zebra value of it is high enough, I draw the bounding rectangular.

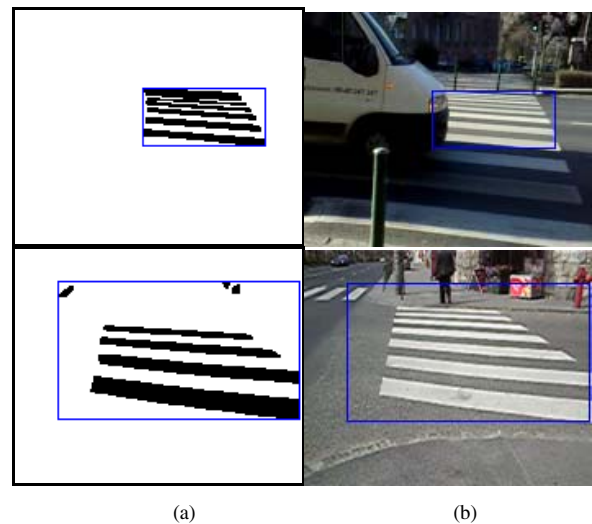


Figure 5. (a): found crosswalks with high zebra values and bounding rectangular; (b): following frames showing the candidate processing area.

On the following frame only a quick processes are done in that candidate area – like a THRESHOLD followed by an EDGE template[11] – to get confirmatory zebra values. It is repeated unless the value get low, or I reach the end of a five frame long cycle. This can be done in a fraction of the running time of the original algorithm. This way I get the possibility to run different algorithms parallel in one frame, like calculating the confirmatory zebra value, and on the same frame an estimation for position and orientation can still be done real time. Figure 5. Shows some examples.

IV. RESULTS AND DISCUSSION

The improved algorithm has been tested on a much broader range of possible inputs, with an increased number of test pictures. The performance of crosswalk detection on five different, prerecorded videoflows, summing up on about 1700 frames are shown in Table I.

TABLE I. DETECTION RESULTS ON ALL THE VIDEOS

	Crosswalk detected	Crosswalk non detected
Crosswalk	613	447
No Crosswalk	15	666

In 73.4% of the cases it performed well, however false positive (missclassified non-crosswalk) result show up only in 1% of the cases. Considering that two of the videos were recorded in bad lighting conditions, I made a different summary consisting only the good quality ones. The results are shown in Table II.

TABLE II. DETECTION RESULTS ON GOOD QUALITY VIDEOS

	Crosswalk detected	Crosswalk non detected
Crosswalk	525	200
No Crosswalk	15	491

In that case the algorithm performed well on 82.5% of the cases. From a practical point of view, false positives are much more dangerous than false negatives, because they would induce the person to cross at a point where no crosswalk is present, so this is the value that has to be minimized at all costs. False negativ (missclassified crosswalk) results appeared in 25% – all the five videos – and 16% – on good quality videos – of the cases. That high percentage of missclassification is usually due to the slow adaption of the builtin auto-gain function of the mobile camera. That causes slow fade in, and fade out effect, which lasts for about a dozen of frames.

The methods, discussed in Section III. performed well on early test, however they are still in a progressive development state. All the results are promising, and the different algorithms are onto be fused together.

V. CONCLUSIONS

I presented a CNN algorithm to detect pedestrian crosswalks by using the mean shift method as the initial color segmentation method. The main advantage of mean shift approach is that the constructed algorithm forms a robust, nonparametric system. The disadvantage is that in certain cases it can be affected by large shadows or very bright areas.

Considering the perspective distortion of the crosswalks, the coefficients of the morphological post-processing following the CNN algorithm can vary in space, which increases the performance of the algorithm.

When dealing with video flows, we can greatly improve recognition by making use of the hypothesis about the location of the crosswalk based on previous frames. By using candidate key-frames that are fully processed, on the following frames we have the possibility to make estimations on the position and orientation of the crosswalks

REFERENCES

- [1] M. Radvanyi, G. Pazienza, and K. Karacs, "Crosswalk Recognition through CNNs for the Bionic Camera: Manual vs. Automatic Design" in Proc. 2009. European Conference on Circuit Theory and Design (ECCTD'09) Antalya, Turkey, 2009.
- [2] T. Roska, D. Bálya, A. Lázár, K. Karacs, R. Wagner, and M. Szuhaj, "System aspects of a bionic eyeglass," in Proc. of the 2006 IEEE International Symposium on Circuits and Systems (ISCAS 2006), Island of Kos, Greece, May 21–24, 2006, pp. 161–164.
- [3] K. Karacs, A. Lázár, R. Wagner, D. Bálya, T. Roska, and M. Szuhaj, "Bionic Eyeglass: an Audio Guide for Visually Impaired," in Proc. of the First IEEE Biomedical Circuits and Systems Conference (BIOCAS 2006), London, UK, Dec. 2006, pp. 190–193.
- [4] K. Karacs, A. Lázár, R. Wagner, B. Bálint, T. Roska, and M. Szuhaj, "Bionic Eyeglass: The First Prototype, A Personal Navigation Device for Visually Impaired," in Proc. of First International Symposium on Applied Sciences in Biomedical and Communication Technologies (ISABEL 2008), Aalborg, Denmark, 2008.
- [5] A. Zarandy and C. Rekeczky, "Bi-i: a standalone ultra high speed cellular vision system," IEEE Circuits Syst. Mag., vol. 5, no. 2, p. 36–45, 2005.
- [6] T. Roska and L. O. Chua, "The CNN universal machine: an analogic array computer," IEEE Trans. Circuits Syst. II, vol. 40, pp. 163–173, Mar. 1993.
- [7] L. O. Chua and T. Roska, Cellular Neural Networks and visual computing. Cambridge, UK: Cambridge University Press, 2002.
- [8] M. Á. Carreira-Perpiñán, "Fast nonparametric clustering with Gaussian blurring mean-shift," in Proceedings of the 23rd international Conference on Machine Learning (Pittsburgh, Pennsylvania, June 25 - 29, 2006). ICML '06, vol. 148. ACM, New York, NY, 2006. pp. 153–160.
- [9] K. Fukunaga and L. Hostetler, "The estimation of the gradient of a density function, with applications in pattern recognition," in IEEE Trans. Information Theory, Vol. 21, Issue: 1, 1975. pp. 32–40.
- [10] D. Comaniciu and P. Meer, "Mean shift analysis and applications," in The Proceedings of the Seventh IEEE International Conference on Computer Vision Vol. 2, 1999. pp. 1197–1203.
- [11] L. Kék, K. Karacs, and T. Roska. (2007) Cellular wave computing library, version 2.1 (templates, algorithms and programs). [Online]. Available: http://cnn-technology.itk.ppke.hu/Template_library_v3.1.pdf visited on 19-06-2010.

Towards recognition-driven semantic shape classification

Attila Stubendek

(Supervisors: Dr. Kristóf Karacs and Dr. Tamás Roska)

stuat@digitus.itk.ppke.hu

Abstract—An attempt for general shape feature classifier is introduced towards human-like semantic learning. The goal is to fill in the semantic gap between the low-level descriptors and the high-level models avoiding the classic case-dependent specialized learning paradigm. The used shape descriptors were chosen based on adequacy for scale- and rotation-invariance, recognition, description and parallel computation. The classification capabilities of the three descriptors (Zernike Moments Descriptor, General Fourier Descriptor, Central Distance Descriptor) are compared according to higher level geometric properties.

Index Terms—global features, image semantics, shape description, recognition

I. INTRODUCTION

In object-recognition the 2D picture taken by a camera is usually characterized by its global and local features. For extracting local properties the image is divided into segments using statistical information (color, texture, energy)[1] or to parts with the same size not taking into consideration the information[2].

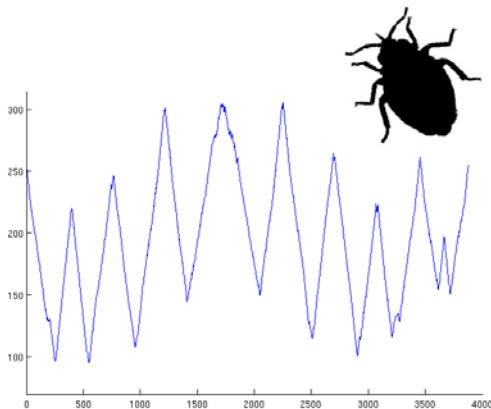


Figure 1: The centroid distance function of the bug shape

After the segmentation features are extracted from the segments of interest. The main features are color, texture, shape[3] and in some cases special points or regions[4]. The present work discusses shape description methods and their semantic discrimination capabilities.

The analysis of the segments can be utilized not only directly to identify given characteristics but the segmentation can also be corrected towards the real semantic-segmentation.

II. SHAPE DESCRIPTORS

The shape can be characterized in various ways, the published descriptors differs in the mathematical

formalization of the shape and in also in the feature extraction. I investigated shape descriptors that:

1. can be easily created by mathematical analysis,
2. is feasible for recognition,
3. are optimal for many-core architectures,
4. are rotation and scale-invariant.

A. Boundary-based methods:

In boundary-based methods only those pixels are taking into consideration that are inlier but are connected to at least one outlier pixel.

Central distance descriptor – A distance vector is created from the distances of the border-points and the central point of the shape. This vector can be considered as a function and can be analyzed in several ways.

The order of the points is essential when measuring the distances. Jadev et al.[5][6] used breadth-first search. In this case the function curve became serrated because the

processed pixels in order are not necessary neighbors and this causes the jumps in the function.

I modified the algorithm by processing the points in depth-first way, using stack data structure. In this case and in normal way a consistent shape border line results continuous central distance function. To avoid other problems caused by one-pixel thick tails or only diagonally continuous parts four times larger border-picture was used without interpolation and finally morphological skeletonization was applied.

Once the function is created a normalization to a given parameter and a transformation – typically and actually Fourier – is made resulting the feature vector. An example of a CD function is on Figure 1.

The disadvantage of the CDD is the lack of ability to reconstruct the original shape from the feature vector because of the loss of the angle data. If an angle vector is saved, more information can be kept by the model. The other disadvantage of the method is that the holes in the shapes are not put into consideration.

Curvature Scale Space – The curve-method is mentioned only to give an other example for boundary-based group, but it does not fulfill the b) and c) requirements.

B. Region-based methods:

Generic Fourier Descriptor – Using two dimensional Fourier-type transformations to characterize images is a popular technique. For shape-recognition the most used and studied method is the Generic Fourier Descriptor (GFD). The image has to be squared, and it is considered as a 2D function.

In order to fulfill the invariance requirement, the image is transformed into polar-space, sampled to a given resolution and transformed by 2D-Fourier transform. The result is an \mathbf{r} by $\mathbf{\theta}$ sized complex matrix as the feature vector [7] (Figure 2).

The disadvantage of the GFD is that because of the polar-transformation the central region of the object is dealt with higher attention while border regions can easily lose their characteristics because of the high-scaled resize on the periphery.



Figure 3: The GFD process: a) The original image, b) the image transformed to polar space, c) the Fourier spectra of the polar image

Zernike Moments Descriptor – The image canvas is considered as a statistical space and the inner pixels as statistical set of 2D points and the shape is characterized by the statistical moments of the points. The higher number the order of the moments is, the better is the description and special is for the given shape losing general features.

The standard moments are often used to express eccentricity but the moments are not rotation invariant. A modification can handle this problem. The Zernike Moments Descriptor (ZMD) projects the shape onto a unit disc and the $(n,m)^{th}$ moment is computed by the following formulas:

$$A_{nm} = \frac{m+1}{\pi} \sum_n \sum_m P_{xy} [V_{nm}(x,y)]^*, \text{ where}$$

$$V_{n,m}(r,\theta) = \left[\sum_{s=0}^{\lfloor \frac{m-|n|}{2} \rfloor} (-1)^s F(m,n,s,r) \right] \cdot e^{jn\theta}, \text{ and}$$

$$F(m,n,s,r) = \frac{(m-s)!}{s! \left(\frac{m+|n|-s}{2} \right)! \left(\frac{m-|n|-s}{2} \right)!}$$

The feature-vector is than an $n+1 * n+1$ complex matrix of the moments, where n is the maximal order. The rotation-invariance is not direct in moments but in their magnitude. Same but rotated shapes has different moments but equal absolute value of moments.[8] [9]

The disadvantage of the ZMD is that computing \mathbf{F} constants is highly complex and different constants have to be computed for different image size and order number. The computation of a constant vector of order of 20 takes 35 seconds on an usual PC (2.2 GHz Dual Core, 2GB RAM), but probably the time can be reduced using higher parallelization.

The big advantage of the ZMD is that the shape is easily reconstructed from the transformed one using only few



Figure 2: Reconstruction from Zernike moments. On the left side is the original shape, on the right the reconstructed. 15 orders were moments. (Figure 3).

Due to robustness and efficiency of Zernike moments the method is being used recently in medical imaging and describing 3D objects too.[10]

Wavelet-descriptor – The descriptor is just mentioned as an other type of integral-transformation descriptors, but the Wavelet transformation is used rather as a similarity-index between pictures[11] or used together with another method.[5][6]

III. SEMANTIC LEARNING

1) The traditional learning

The classic learning paradigm uses mainly low-level features for target detection. A classification is made by computing distances from the class means and the minimal distance is chosen, or a learning model is trained, mainly neural networks, detection trees or regression.

The disadvantage of this method is the inflexibility and the lack of capability for generalization and abstraction.

2) The general semantic learning

The semantic learning concept introduces an intermediate semantic layer on the top of feature extraction that enables the semantic characterization of the target, resulting extreme abstraction 3D detection capability.

Given a special feature that is not necessary directly tagged in the samples of the training set but it is described by previously known features, the general semantic query answers two questions:

- a) “Which features are present in a given shape?”
- b) “Which shapes contain a given feature?”

The first approach towards semantic characterization of the sample is describing every little segment and in some way building up a hierarchical model. In shape analysis this direction is called structural decomposition and only few attempts were made.

An other approach is not finding every local feature but extracting global features from the image that can be useful for example for visually impaired people. Oliva et al.[12] showed that global characteristics, like distance, openness, etc. can be derived using transformations. In my work I focused on this direction but in the area of shapes.

From this consideration I defined global shape-features and tried to build a model trained by a machine-learning algorithm. These special features are the following:

- (1) Edges – Is the border serrated or smooth?

- (2) Corners – Are the corners acute or rounded?
- (3) Detailedness – Similar to the serration but in higher resolution. Is the shape compact or detailed?
- (4) Axial symmetry – The direction of the axis is not important, and not precise symmetry is needed.
- (5) Central symmetry – Similar to axial symmetry but central.
- (6) Flatness – It the shape longish or roundish?
- (7) Characteristic line – What is the characteristic line of the shape? The use of the edge and skeleton lines appeared in the work of Zaboli et al.[13] and Torres et al.[14], when they used the data from gradual skeletonization.
- (8) Symbol – Is the investigated shape a character, symbol, or not? An other morphologic approach was given by Karacs and Roska.[15]

Examples are demonstrated on Figure 4. Beyond these properties other features can be defined too, the developed framework can be easily extended.

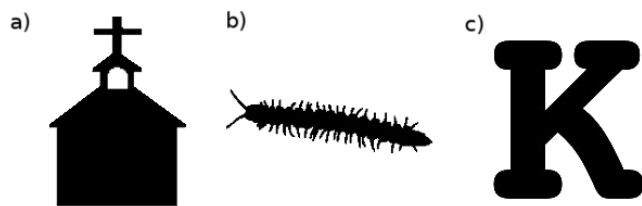


Figure 4: Sample pictures from the database. The features: a) smooth borders, acute corners, axial but not central symmetry, not longish, not a character, detailed and border-typed; b) longish, serrated border, acute aces, symmetry of both types, not a character, detailed and skeleton-typed; c) smooth borders and corners, only axial symmetry, character and skeleton-typed not longish and detailed

IV. REALIZATION

A. The database

The shape database was collected from various websites using Google picture finder. The images were binarized and resized to the size of 200 x 200 pixels. The images were tagged based on the semantic classes. However not every shape has every feature class because of ambiguity.

B. Used algorithms

For the GFD and CDD calculations and for pre- and post-processing I used my own programs created in Matlab. The ZMD code was downloaded from a public site.

Because every input for the training algorithm has to be a row vector with real numbers, if the result of a descriptor was a matrix (ZMD) I reshaped it to a row vector and if the descriptors were complex (ZMD, GFD) the real and imaginary parts were considered as two feature vector elements.

In the tests I used 50-long distance vector for CDD, 20 radial and 20 tangential steps in the GFD and maximal 20th order of the ZDF.

C. Machine learning

I used the Matlab's LVQ networks for classifying the shapes. The Learning Vector Quantization algorithm has a multi-layer artificial neural network which applies a winner-

take-all Hebbian learning-based approach. The schematic structure of an LVQ-net is on Figure 5.

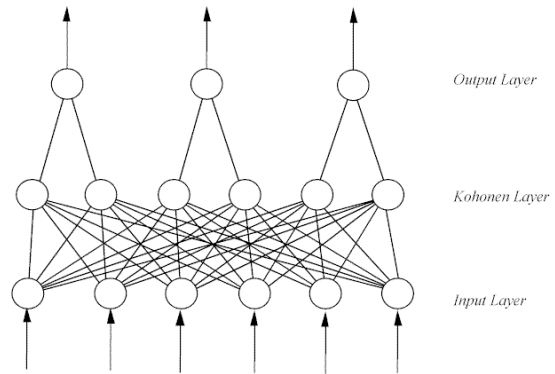


Figure 5: The structure of the LVQ net. (Image source: <http://www.afrinc.com/products/fire-detection/>)

IV. RESULTS

A. Feature classification

The introduced algorithms were tested to classify the shapes based on the general shape features. The results of the classification are on Table 1 and on Figure 1:

Correct answer %	ZMD (20)	GFD (20 20)	CDD (50)
Edges	90,5	69,2	83,3
Borders	82,5	70,7	79,5
Axial symmetry	87,5	75,7	80
Central symmetry	92,5	79,5	90,9
Flatness	100	88	92,9
Character	92,9	77,8	77,8
Characteristic line	92,9	80	84,4
Proportions	81,8	77,3	84,4

Table 1: Efficiency of the algorithms

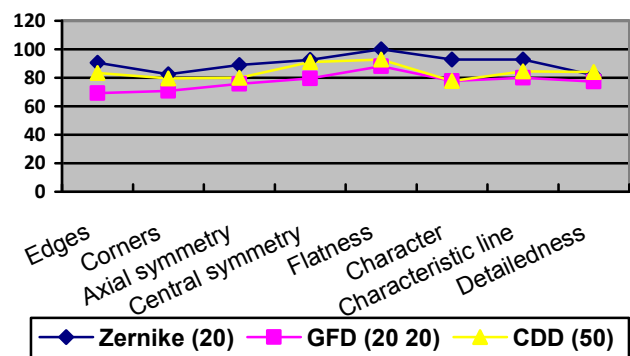


Figure 6. Results on a graph

B. Semantic query

Using the trained models that classify the shapes by the given properties, the system is able to answer the general semantic queries.

The result for the semantic query “Which features are present in the shapes?” is on Figure 6 are on Table 2.

VI. FUTURE PLANS

The future work on shape descriptors can be diversified. On one hand the algorithms can reach higher efficiency by optimizing their parameters, on the other hand new descriptors can be given and new features can be defined.



Figure 7: Test pictures for semantic query 1.

	Picture a)				Picture b)				Picture c)			
	c	G	Z	C	c	G	Z	C	c	G	Z	C
Edge.	2	1	2	2	1	1	2	1	1	1	1	1
Corner	1	2	1	2	1	2	2	2	2	2	2	2
AxSy.	2	2	2	2	2	2	2	2	1	1	1	1
CenSy.	1	1	1	1	1	1	1	1	2	2	1	1
Flat.	1	1	1	1	1	1	2	1	1	1	1	1
Symb.	2	2	2	1	1	1	1	1	1	1	1	1
CharL.	2	2	2	2	1	1	1	1	1	1	1	1
Det.	1	1	1	1	2	2	2	2	2	2	2	2

Table 2: Results of the semantic query 1. The characters in the second row are: c - correct, G - GFD, Z - ZMD; C - CDD; The property numbers are: Edge: 1-smooth, 2-serrated; Corner: 1-rounded, 2-acute; Flatness: 1-roundish, 2-longish; Characteristic line: 1-border, 2-skeleton; Axial, Central symmetry, Symbol, Detailedness: 1-no, 2-yes; The bad classifications are shown in red color.

The interesting result of test was the “wrong” classification of borders of the second image. Actually it is really hard to decide the property and probably the tagging before the training was wrong and was not consistent with the other hand-made classifications.

The second semantic query “Which shapes contain the given feature?” was executed too. An example answer is in Table 3.








Yes:				
No:				

Table 3: An example answer for a query: "Which shapes are axially symmetric from the given samples?" The used descriptor is the ZMD.

V. CONCLUSION

In our experiments the ZMD gave the best results for semantic classification, in one case the CCD was better. However the difference between the methods seems to depend on the property to be classified. That might explain the fact that previously published results arrived to different conclusions regarding the classification capabilities of these methods. [5][6][7][9]



Figure 8: A shape where the decomposition is complicated but the human eye can recognize the shapes easily.

An other direction leads to the already mentioned structural decomposition. If we have shapes stuck together in the way that they cannot be divided correctly even by morphological opening, the shape analysis needs a new kind of view (Figure 7).

REFERENCES

- [1] M.I Krinidis and I. Pitas, "Color Texture Segmentation Based on the ModalEnergy of Deformable Surfaces",*IEEE T. On Image Processing*, 2009
- [2] G. Carneiro, A. B. Chan, P. J. Moreno, N. Vasconcelos, "Supervised Learning of Semantic Classes forImage Annotation and Retrieval",*IEEE T. on Pattern analysis and Machine Intelligence*, 2007
- [3] J. Wang, and Y. Yagi, "Integrating Color and Shape-Texture Featuresfor Adaptive Real-Time Object Tracking",*IEEE T. On Image Processing*, 2008
- [4] A. W.M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-Based Image Retrievalat the End of the Early Years",*IEEE T. On Pattern Analysis and Machine Intelligence*, 2000
- [5] R. B. Yadava, N. K. Nishchalb, A. K. Guptaa, V. K. Rastogi, "Retrieval and classification of objects using generic Fourier,Legendre moment, and wavelet Zernike moment descriptors andrecognition using joint transform correlator",*Optics & Laser Technology*, 2008
- [6] R. B. Yadava, N. K. Nishchalb, A. K. Guptaa, V. K. Rastogi, "Retrieval and classification of shape-based objects using Fourier,generic Fourier, and wavelet-Fourier descriptors technique: A comparative study",*Optics and Lasers in Engineering*, 2007
- [7] D. Zhang and G. Lu, "Generic Fourier Descriptor for Shape-based Image Retrieval", *IEEE International Conference on Multimedia and Expo, 2002 - Citeseer*, 2002
- [8] A.Khotanzad and Y.H. Hong, "Invariant image recog-nition by zernike moments",*IEEE Trans. on PatternAnal. and Machine Intell.*, 1990
- [9] H. Shin Kim and Heung-Kyu Lee, "Invariant Image Watermark Using Zernike Moments",*IEEE T. On Circuits and System for Video Technology*, 2003
- [10] V. Venkatraman, P. R. Chakravarthy³ and D. Kihara, "Application of 3D Zernike descriptors to shape-based ligandsimilarity searching",*Journal of Cheminformatics*, 2009
- [11] M. P. Sampat, Z. Wang, S. Gupta, A. C. Bovik and M. K. Markey, "Complex Wavelet Structural Similarity:A New Image Similarity Index",*IEEE T. on Image Processing*, 2009
- [12] M. R. Greene and A. Oliva, "Natural Scene Categorization from Conjunctions of Ecological Global Properties",*Proceedings of the 28th Annual Conference of the Cognitive Science Society*, 2006
- [13] H. Zaboli, M. Rahmati and A. Mirzaei, "Shape Recognition by Clustering and Matchingof Skeletons",*Journal Of Computers*, 2008
- [14] R. da S. Torresa , A.X. Falcoa, L. da F. Costa, "A graph-based approach for multiscale shape analysis",*The Journal Of The Pattern Recognition Society*, 2003
- [15] K. Karacs and T. Roska, "Locating and Reading Color Displayswith the Bionic Eyeglass",*RET Szentágotthai Tudásközpont*, 2005

Wave Computational Abilities of Large Infrared Proximity Arrays

Miklós Koller
(Supervisor: György Cserey)
kolmi@itk.ppke.hu

Abstract—A new hardware-software system is introduced in this paper. It is made of a peripheral device, which contains an array of sensors, and a high-level processing algorithm, running on a PC. The cells apply some special modification on their local connection attributes, as compared to the Cellular Neural Networks' local connections. The cells of this device have reflective type infrared distance sensors, each cell containing a LED and a phototransistor.

This special hardware can play dual role: not only the input of the computational system is made up during the measurement, but on the other hand, it forms a non-uniform weighting factor on the connections of adjacent cells. This locally adapted weighting is accomplished by the reflectance feature of the measured object (or the environment). This modification of the local connections has an important meaning: the physical environment takes a part (and probably the control) of the processing.

In this way, we can achieve a more complex processing, instead of utilizing the raw distance image as an input only.

Index Terms—sensor-array, infrared

I. INTRODUCTION

This device was originally developed to deal with the phenomenon of the hyperacuity [1] in space [2], [3]. A key element of the necessary conditions of this theoretical hypothesis was to applying sensors in group, where the information quantity projected to an elementary sensor contains more values, than applying it alone. That is why we designed and manufactured a sensor block, where the elemental sensor was an infrared distance-sensor (a light emitting diode and a phototransistor). The elemental sensors formed an array, most likely an 8×8 matrix. So all of the cells may work as light-source and sensor. The applied measuring methodology had the following basis: from a given 3D-surface we were able to take more than one shot, altering the spatial pattern of the activated light-sources, while all of the sensors was measuring the reflected light. In this way we were able to improve the resolution of the finally created picture [4].

The architectural construction of our device makes possible to handle it as the input interface of a simulated, parallel computational system, formed from independent, equal ranked computational units. The topographical order of the elementary sensors suggests to implement the computational system of the two-dimensional cellular automata (CA), furthermore, to realize a special CNN simulator, where the environment modifies the weights between the cells. For this simulator the input data is serviced by our panel.

In the case of the CA implementation, the local connectivity between the adjacent units can be accomplished through the

reflected light from the environment. The purpose of this CA implementation is only "proof of concept", not commensurable with performance-oriented implementations [5].

In the case of the CNN simulator, we implemented some kind of "image sequence" processing: the input image is not a normal, two-dimensional data set (as a pure picture containing average-values of the LEDs' reflected light), but every sensor measures its input pixel value under different light-activating pattern.

This paper is organized as follows: in Section II. some information about the sensor array and the computational models are presented. In Section III. we describe the details of the implementation; in Section IV. we show the experimental results; and in Section V. the conclusions are presented.

II. HARDWARE-SOFTWARE SYSTEM

A. Sensor array

The sensing part of this device contains 64 infrared distance sensor (TCRT1000), in a 8×8 matrix order. Each cell contains an infrared LED and an infrared phototransistor. The size of the whole array about 9×9 cm. The circuit realization needed two printed circuit board: one of them contains the distance sensors, the other contains the controlling and processing circuits. Fig. 1. shows the manufactured device. This modular build permits of changing the sensor array without the rebuild of the servicing circuit.

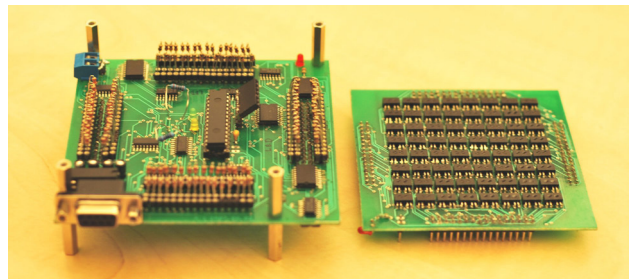


Fig. 1: The manufactured Large Infrared Proximity Array.

The main control function is managed by a microcontroller (MCU - PIC18f2321), which uses additional ICs to handle the sensor array. The LED-switching subsystem uses serial in - parallel out shift-registers (74HCT164) to designate the LEDs (which must be activated); and Darlington-arrays (ULN2803) to drive for sufficient power. The read-out subsystem uses

analog multiplexers (4051D) to merge the outputs of the phototransistors. To measure the output current of a phototransistor, we use a potential-dropping resistor, and the on board AD converter module of the MCU. The microcontroller has its communication interface through RS-232 to contact with the PC (high-level / additional data-processing).

It is important to note: our hardware does not contain real parallel computational and processing units, only the sensorial data-collecting works on a parallel way. The spatial distribution of the physical sensors generated the idea: a simulator-like implementation of the individual computational units would be "interface-able" on their input side with the measuring data of the elementary sensor blocks.

B. Software distribution

The microcontroller on the sensor-board collects the raw data of the measurement. This data are sent to an application to the PC, which realizes the model of the computational cells and their time-evolution. The microcontroller would be almost able to service all of the necessary computations, but the iterative development is easier when the computations are implemented on a PC.

III. DETAILS OF IMPLEMENTATION

A. Cellular Automaton

The purpose of implementing the cellular automaton was only to show the role of the local connections, through the reflected light.

The implementation realizes Conway's Game of Life [6]. The computational abilities of the individual automata are simulated in software, the sensor array realizes only the locally inter-cell communication. That means: we have 64 "individual" CAs. In every iterative loop, the living cells activate their light source, making possible to all of the cells to decide: in the forthcoming generation they going to be dead or alive. Every individual cell measures its neighboring cells'

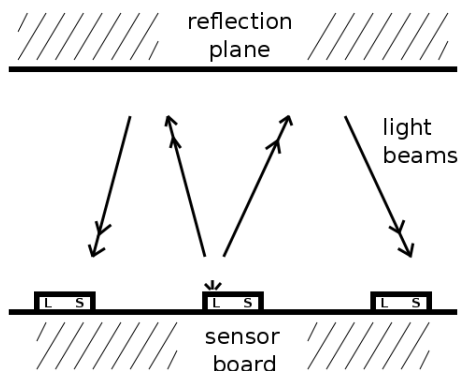


Fig. 2: The measurement setup while simulating a Cellular Automaton. We use constant, flat plane as reflective medium, and every sensor in every cell tries to measure the adjacent cells light.

light. We use threshold values to know from the measured light-quantity the number of the living adjacent cells. After each cell has measured the adjacent cells' reflected light value, the simulator computes for each cell the appropriate state in the next iteration. We use a flat plane for reflective media (for the schematic of the measurement setup see Fig. 2.), which is placed parallel with the surface of the sensor array. That is why the state-changes of the individual cells are not supervisable through looking the lightning LEDs. We use the result of the simulator to validate the "correctly ordered" generation arising.

B. Cellular Neural Network

In the case of our CNN simulator, we follow the above mentioned system-distribution: the sensor array realizes only the input channel of every cell. The computational/processing abilities of every cell are almost realized in the simulator, purely on virtual way. There is no real individual processing item behind every cell.

The most interesting part of our CNN simulator, is the acquisition of the input picture. To show the correct details, we are going to analyze a simple cell, for example in the i -th row and the j -th column. The general state equation of our elementary cell, $C_{i,j}$, is as follows:

$$\dot{x}_{i,j} = -x_{i,j} + \sum AY + \sum BU + z$$

The input matrix (U) in a normal system [7] (and with $r := 1$) is filled with the neighbouring and the own input pixel-value. In our system, we do not know exactly, what input value is measured by the neighbouring cells' sensor. Instead of telling it in a "galvanic" or "software" way, we try to measure the reflected light of the neighbouring cells' LED. It means: we use the environment in the local connections, somehow as a weighting factor, which could probably raise the processing power of the array [8]. Furthermore - if we work only with uncoupled templates - we could implement the inter-cell connections in a pure, light-coupled way. For the neighbour cells' input value acquisition of an individual cell, see Fig. 3.

The simulator uses the Euler-approximation of the CNN state-equation. By every iteration, every cell uses its light source to get input information from the corresponding part of the measured object. But when one cell measures his own LED's reflected light, the neighbours are also measuring its reflected light, from the appropriate direction. For cell $C_{i,j}$, in the state-equation, the neighbours input is not exactly that value, what the neighbours measure above themselves: the environment play some role in the acquisition process of the input values.

IV. EXPERIMENTS

In this section we show experimental results for both of the above mentioned simulators.

Fig. 4. shows a Game of Life simulation, with the initial pattern of a double Bipole. It is an oscillator, with period

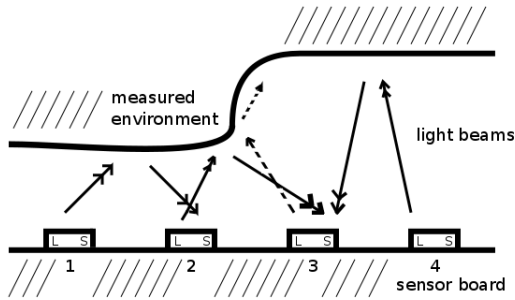


Fig. 3: This picture shows, how play significant role the environment, when a cell want to know it's neighbours' input. When the 2nd and the 3rd sensor is measuring the corresponding adjacent cell's input: the 3rd sensor (S3) could measure almost correct the neighbour cells' input (from L2 and L4), but the 2nd sensor (S2) have other measuring value from the neighbours' reflected light (from L1 and L3), than the neighbours have in real.

number 2.

Fig. 5. shows the CNN simulator outputs. In this measurement we use the well known grayscale edge-detection template (EDGEGRAY) [9], which is as follows:

$$A = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{pmatrix}, B = \begin{pmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{pmatrix}, z = -0.5$$

It is important to mention, that the input weights have modified in certain sense, which effect comes from the untraditional input "image" capturing. There is a stair on the input picture, the right side of the picture is the far step. The measurement setup was just like the schematic drawing on Fig. 3.

V. CONCLUSION

In this paper [10] we presented a large infrared proximity array (LIPA), which was originally made to analyze the phenomenon of the spatial hyperacuity. Due to its architecture it is capable to process image-flow type information like CNN, with a significant difference: the local interconnections between the cells are realized through the reflected light of the other cells. In this way, the environment can cause locally different connection-weights, which could be advantage by the processing of the arising image-flow. We introduced this achievement through simple examples, but applying our method we expect more promising applications using complex CNN algorithms.

ACKNOWLEDGEMENT

The Hungarian Scientific Research Found (OTKA) which supports the multidisciplinary doctoral school at the Faculty of Information Technology of the Pázmány Péter Catholic University is gratefully acknowledged. And special thanks goes to Ákos Tar.

REFERENCES

- [1] K. Lotz, L. Boloni, T. Roska, and J. Hamori, "Hyperacuity in time: a CNN model of a time-coding pathway of sound localization," *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on* [see also *Circuits and Systems I: Regular Papers, IEEE Transactions on*], vol. 46, pp. 994–1002, Aug. 1999.
- [2] A. Brückner, J. Duparré, A. Bräuer, and A. Tünnermann, "Artificial compound eye applying hyperacuity," *Opt. Express*, vol. 14, no. 25, pp. 12076–12084, 2006.
- [3] D. T. Riley, W. M. Harmann, S. F. Barrett, and C. H. G. Wright, "Musca domestica inspired machine vision sensor with hyperacuity," *Bioinspiration & Biomimetics*, vol. 3, no. 2, p. 026003 (13pp), 2008.
- [4] A. Tar, M. Koller, and Gy. Cserey, "3D geometry reconstruction using large infrared proximity array for robotic applications," *ICM 2009. 5th IEEE International Conference on Mechatronics*, 2009.
- [5] J. Tran, D. Jordan, and D. Luebke, "New challenges for cellular automata simulation on the GPU," 2004.
- [6] M. Gardner, "The fantastic combinations of John Conway's new solitaire game "life";," *Scientific American*, vol. 223, pp. 120–123, October 1970.
- [7] L. O. Chua and T. Roska, "The CNN paradigm," *IEEE Trans. Circuits Syst. I*, vol. 40, pp. 147–156, Mar. 1993.
- [8] M. Balsi, "Generalized CNN: Potentials of a CNN with non-uniform weights," in *Cellular Neural Networks and their Applications, 1992. CNNA-92 Proceedings., Second International Workshop on*, pp. 129–134, Oct 1992.
- [9] L. O. Chua and T. Roska, *Cellular Neural Networks and visual computing*. Cambridge, UK: Cambridge University Press, 2002.
- [10] M. Koller and Gy. Cserey, "CNN computational abilities of large infrared proximity arrays," *CNNA 2010. 12th International Workshop on Cellular Nanoscale Networks and Their Applications*, 2010.

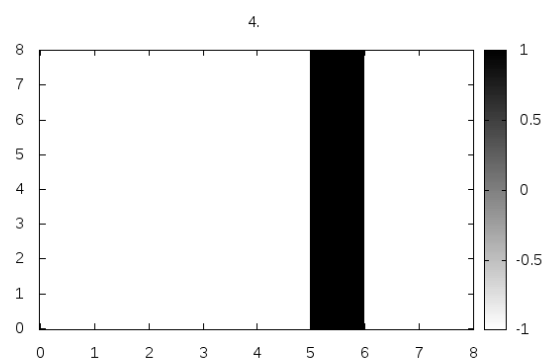
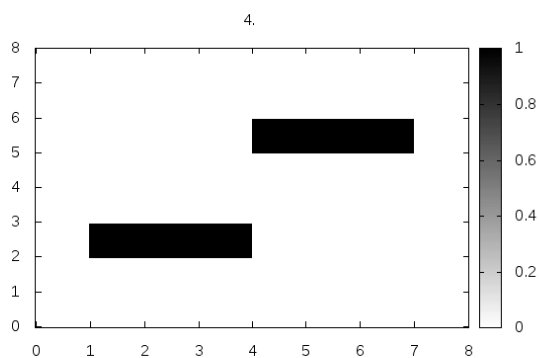
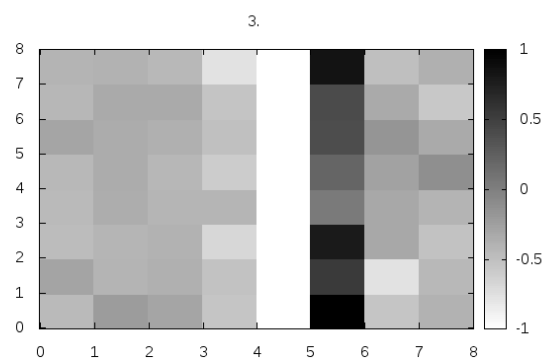
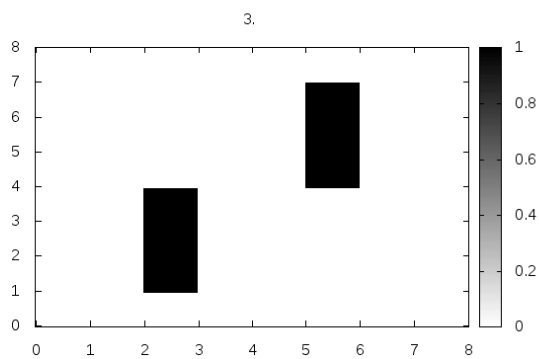
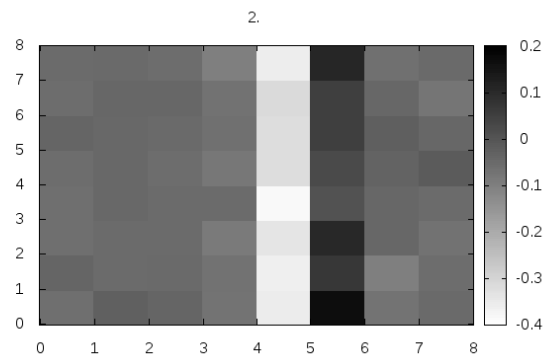
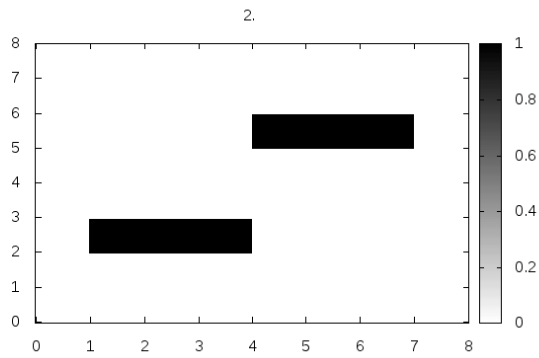
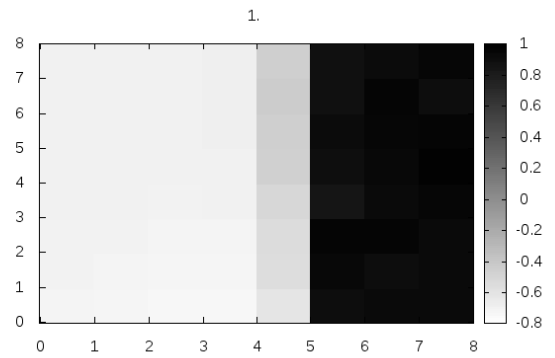
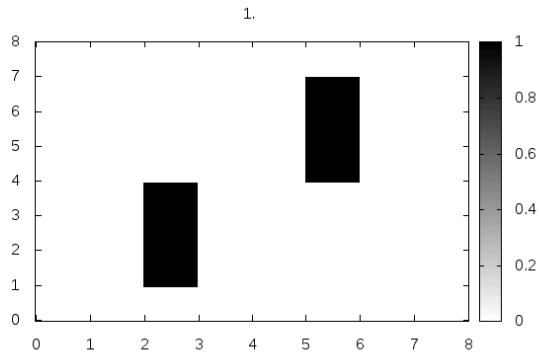


Fig. 4: Cellular Automaton - simulating the Game of Life, with the initial state: Bipole (doubled). This is an oscillator pattern, with period number 2. The first picture shows the initial state of the cells, and the others the forthcoming iterations. The reproduction of the appropriate pattern denotes: the local connections through the reflective way suitably accomplish their role.

Fig. 5: CNN simulation, with edge detection templates. The picture numbers as follows:
 1: The input - measured distances coded by grayscale codes (-1: close, 1: far)
 2: The output - after the 1st iteration
 3: The output - after the 6th iteration
 4: The output - after the 40th iteration

Investigating the possibilities of processing parallel resources with language statistical methods

László Laki

(Supervisor: Dr. Gábor Prószéky)

laklaja@digitus.itk.ppke.hu

Abstract—Nowadays the spread of the internet and the growing number of digital texts help to develop corpus linguistics. In this article we examined two tasks where the transformations between the two languages were carried out with statistical methods and the use of bilingual parallel corpus. As the first step we worked on the increasing the result of the statistical machine translation with deeper hybridization steps. Second our system was tested on a grammatically similar language pair, Hungarian-Gypsy. Finally my statistical text annotation system was installed and evaluated.

Index Terms— Bilingual parallel corpus, Language statistical method, Statistical Machine Translation (SMT), Text annotation

I. INTRODUCTION

The fast development of information technology opened wide spectrum of opportunities in almost all disciplines. This evolution could be detected on the field of linguistics as well. The handling of huge text materials becomes easier and the efficiency of these systems is increasing. This fact leads to the strengthening of computational linguistics, where the following main trends can be observed:

- The main point of the *rule-based linguistics* is that the generation of sentences is based on pre-defined rules and regularities that are built into the translator. The rules are not able to follow the changes of natural language since it is difficult to change them more over they define its limits.
- In contrast, *corpus linguistics* has a reverse approach to the problem, namely it studies the language systematically through real language data. This provides a more accurate description of the natural languages, whose main characteristics is the continuous change.

The spread of the Internet and the digitalization of printed texts led to the availability of the mass of textual data very different in style, theme and content. These investigations can lead to such observations and to the creation of systems that had never been possible before.

One of the challenges is that between the different languages – whether they are natural or artificial – there are such texts that have to be transformed from one to the other. It is easy to see that the usage of rule-based methods is quite difficult, because the set of rules could be determined hardly. It seems to be much more obvious to apply statistical methods to solve these tasks.

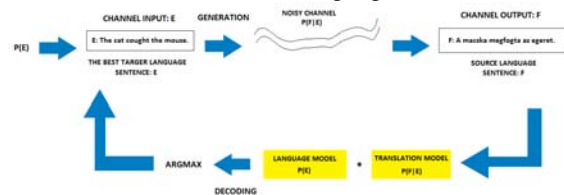
My goal was to study such kind of transformations between languages; to improve the existing methods; to assess the opportunities and make appropriate implementation.

II. STATISTICAL MACHINE TRANSLATION

Machine translation is a basic branch of statistical language processing. Statistical Machine Translation (SMT) has a great advantage over rule-based translation; namely that the knowledge of the grammatics of the language is not necessary to create the architecture of the machine translator. Only a bilingual corpus is needed to set up the system training set. From this corpus we can observe statistical observations and rules, which are the cornerstones of the translator.

According to the significant results achieved in foreign language pairs, we chose this method to create machine translation between English and Hungarian. The idea of the SMT method comes from the speech recognition system.

The only thing we know for sure in the beginning of the translation is the phrase which we want to translate – i.e. the source sentence. Therefore, the translator can be defined as a noisy channel. We pass through this channel the set of target sentences and compare the channel output with the source sentence. The result will be the phrase which provides the best match with the source language sentence.



A. Evaluation

An important issue in the case of translator systems is the evaluation of different translations and the comparison of various systems. During this research the BiLingual Evaluation Understudy – BLEU score – was used. The essence of this method is that the translations are compared with the reference sentences. The system calculates the result score in the interval 0-1.

B. Corpora

We used an English-Hungarian parallel corpus, built up from two parts of the Hunglish corpus: literature and magazines.

The Hunglish corpus is a free Hungarian-English parallel corpus, which consists of 54.2 million words and 2.07 million sentences. This corpus (from now on referred to as Lit-Mag) was prepared by the contribution of the BME MOKK and Research Institute for Linguistics department of Hungarian Academy of Sciences. The Lit-Mag is considered to be a small size corpus containing 654 939 sentences and 9 425 911 words.

C. Framework

Several methods were studied, which are able to obtain information from parallel corpora. Finally we decided to use IBM models, because these are relatively accurate, and the used algorithm was adaptable to our task. According to these reasons we started to use the MOSES framework [2], which implements the above mentioned models. This system includes algorithms for the preprocessing of the parallel corpus, the setup of translation and language models, the decoding and the optimization to the BLEU score. After the training this basic system resulted in 0.1085 (10.85%) BLEU scores.

D. Problems and solutions

The achieved results were compared with the results of the EuroMatrix. The aims of the EuroMatrix project is to solve the machine translation between the languages of the European Union.

It is evident from these results that the machine translation between other language pairs gives much better outcomes compared to the English-Hungarian system. For example: English-French 32%, Spanish-Catalan >40%, English-Spanish 30%.

We have to note, that between the above listed language pairs. There is great similarity both on grammatical and word level. Only small differences can be found in word order, grammatical structure and the usage of the language. As a result of these similarities the machine translation is able to create the transformations between these languages more certainly, and the outcoming translations will give better evaluation compared to the reference sentences.

In contrast, between Hungarian and English there is a very significant difference. The most obvious deviation is that Hungarian language is agglutinative, while English is fusional. This means that the same system applied to the English-French pair will provide much better results as in the case of English-Hungarian. To eliminate these differences we need to exploit the specific grammatical features. Therefore, instead of simple statistical-based system a hybrid translator should be used to improve the quality of the translation. The hybrid system is a transition between statistical and rule-based machines translation.

E. Hybrid systems

1) Adding vocabulary to the corpus

During the evaluation we noticed, that the sentence alignment finds the related text parts hardly, if the phrases are far apart because of the grammatical structure.

The first idea is to improve the quality of the original

corpus by adding a bilingual dictionary. We hoped that the accurate translation of the words and phrases will not only help to find the appropriate phrases but also reduces the number of OOV (Out Of Vocabulary) words. The freely accessible QED English-Hungarian database was used for this task (<http://www.math.u-szeged.hu/~bognarv/qed/qed.pdf>).

The dictionary was attached to the corpus several times in order to raise the number of occurrence of the appropriate phrases in the text. Therefore a greater weighting of proper phrases was expected in the translation model. We used the dictionary five times we relying on previous observations reaching the best result. If we used the dictionary less times, it would not be so effective; if we applied it more times, the translation would be too specific.

TABLE I

System	BLUE score
Base system	0.1085
Litmag+5*dictionary	0.1087
5*dictionary+joshua	0.1100

The first table contains the results we obtained. It is visible that compared to the basic framework we reached some improvement (0.02%). If we thoroughly analyze the results we could notice that on the 1-gram level this improvement is more than 0.7% and in case of the 2-grams is 0.15%; also on the level of the other n-grams a lesser degree of improvement occurs. This success is due to that dictionary consists 1- to 2-grams and short phrases.

These results are not deemed to be a huge breakthrough, but point out that the quality of machine translation can be improved by developing the alignment system.

2) JOSHUA

The next step was to analyze the possibilities of deeper hybridization. We can see that if we want to translate remote languages – like Hungarian and English -, over the phrase-based statistics we have to use other characteristic features, like grammatical rules.

The SMT system JOSHUA [3] could be a proper solution for this task. This framework not only applies word or phrase level statistics, but it takes into account the morphological characteristics of the language. Chomsky's generative grammars are able to help us in this case. The languages that could be described with grammatical rules belong to the class of regular languages and context-free grammars (CFG). Consequently, natural languages could be built up with CFG rules.

The great advantage of the JOSHUA system is that it is able to translate between these CFG rules in such a way, that rules can be specified for both source and target languages, furthermore the likelihood of the transformation into each other.

As a result, JOSHUA can be successfully applied for the translation between languages, which are far from each other morphologically and also syntactically.

In this study we used the following rules:

[S] ||| [X,1] ||| [X,1] ||| 0 0 0
 [S] ||| [S,1] [X,2] ||| [S,1] [X,2] ||| 0.434294482 0 0

We have chosen this general grammar in order to estimate whether the method is appropriate or not to solve the task.

The system had been trained with the corpus described in part C. It was necessary in order to get authentic/reliable comparison of the different systems. The evaluation provided the following result:

TABLE II

System	BLUE score
Base system	0.1085
Litmag+JOSHUA+OOV	0.0985
Litmag+JOSHUA	0.1106

The first line of the second table describes the result of the base system. The JOSHUA system joins an OOV tag to every word which is not included in the dictionary. We can observe that it gives worse result than the base system. This is because proper names had been tagged as well with OOV tag during the process. This mistake spoils the translation, although it could be good. To avoid this deterioration we deleted these OOV marks and got the result appearing in the last line. Significant improvement could be reached in this way.

EXAMPLE I

English reference sentence:
 " for a little while only , " said the voice quietly .
Hungarian reference translation:
 - csak egy kis ideig - mondta a hang csendesesen .
Translation of the base system:
 - egy darabig csak - mondta a hang .
Translation with JOSHUA system:
 - csak egy kis ideig nyugodtan - mondta a hang .

The previous example is able to show that the application of a simple rule could provide significant changes in the evaluation. The phrase 'for a little while only' was translated to 'egy darabig csak' with the base system; but the recursion rule of JOSHUA found phrase 'csak egy kis ideig', which – according to the reference – is the correct translation. Through this example it can be seen that for human evaluation both translations are acceptable; but machine evaluation resulted in completely different scores (the first minimal, the second maximal points).

F. Gypsy-Hungarian SMT system

As it had been mentioned in the introduction new opportunities revealed to us in the digital era. For example in December 1998 the 'lovári-Roma' translation of The Holy Bible by Vesho-Farkas Zoltán was published. This gave the idea to try our SMT system with 'lovári Roma'-Hungarian language pair, which are grammatically closer to each other. Both are morphologically rich agglutinative languages. The used corpus was the New Testament of Vesho-Farkas translation, and Gyorgy Kaldi's Catholics translation.

Table III shows the reached results after evaluation:

TABLE III

System	BLUE score
Cigány-Magyar (MOSES)	0.3053
Cigány-Magyar (MERT)	0.3176
Cigány-Magyar (JOSHUA)	0.2920
Magyar-Cigány (MOSES)	0.3038
Magyar-Cigány (MERT)	0.3716
Magyar-Cigány (JOSHUA)	0.3588

We reached far better results as in the case of Hungarian-English language pairs. This success has more reasons. First that both training and testing sets were established from the New Testament. Therefore the SMT system becomes too topic-specific. Furthermore in the case of the Gospels the content and the text occur repetitively; the evangelists describe the same story very similarly. Therefore it is possible that between the test translations we could find 100% correct ones, because these might appear in the corpus elsewhere.

EXAMPLE II

Gypsy reference translation:
 le but manusha pale tele sharadine penge gada po drom ,
 kavera pale kranzhi phagenas tele pa kasht haj po drom
 rispisarnaslen .
Hungarian reference translation:
 a hatalmas tömeg pedig leterítette ruháit az útra , mások meg
 ágakat vagdostak a fákról és az útra szórták .
MOSES translation:
 a nép pedig le terítették ruháikat az úton , mások pedig
 ágakat phagenas le a fa , és az úton rispisarnaslen .
JOSHUA translation:
 a nép pedig le terítették ruháikat az úton , mások pedig
 ágakat phagenas le a fa és az úton rispisarnaslen .
MERT translation:
 a nép pedig letakarták ruháikat az úton , mások pedig ágakat
 phagenas le a fa és az úton rispisarnaslen .

This translation gives better results compared with Hungarian-English SMT system. In addition to the better BLEU scores, the readability is also better for this system.

III. TEXT ANNOTATION

Our research had another direction as well. During the study of the SMT systems, we need a morphological analysis of the corpus. We realized that the text annotation could be formulated in such a way that from plain text we transform to analyzed text with appropriate rules. Thus, for this task we could also apply statistical methods.

There are several methods for text annotation; the two most widely-used are the rule-based technique and the method based on the machine learning. Both methods have many advantages, but they have similar problems as rule-based SMT systems. It is very difficult to set up the proper rules for this system that covers all possibilities and the process needs serious attention.

The other method has same difficulties. The training could be very precise, but if we want to train all rules, we would get a too complex and slow system.

In contrast the statistical method is able to find out all the rules – it just needs an appropriate corpus –, and in addition we get an online system. Therefore, we wanted to investigate what result could be obtained with this method.

We used the Szeged Corpus 2 [1], which was made by the Language Technology Group of University Szeged. This XML-based database contains both plain texts and their part-of-speech clarified version. The advantage of the corpus is that it was manually controlled, therefore is a very accurate data set. Further benefit is that it is general and not topic specific. The only fault is that we get a relatively small corpus.

The results we got after the training shown in the Table IV. We have got 91.06% BLEU score, which means that our analysis is similar to the reference analysis in a very high percentage.

TABLE IV

System	BLUE score
Szeged+MOSES	0.9097
Szeged+ MOSES +MERT	0.9106
Szeged+JOSHUA	0.9096

To

compare our system with others, we have done an error-rate evaluation (Table V).

TABLE V

System	%
Szeged+MOSES+correct%	90.2873
Szeged+MOSES+incorrect%	9.7127

According to this evaluation the 90.28% of the test sentences was analyzed properly and only 9.72% was incorrect. The main deterioration point was the analysis of proper nouns, as these were not included clearly in the corpus (especially the phrases built up from more words). Therefore the translator spoiled these phrases.

We made an experiment, where the test set was not part of Szeged Corpus. We achieved surprisingly good results as well. The most obvious problem was the small size of the training set; many words were not included in the corpus and were left without analysis. However, the system could be optimized such a way, that in these cases it would always give a proposal for the analysis.

IV. CONCLUSION

In this article I presented few tasks where parallel corpora were processed by statistical methods. In this article we examined two tasks where the transformations between the two languages were carried out with statistical methods and the use of bilingual parallel corpus. The result we can obtain will not be much worse or events sometimes better than that off the rule based methods', but much easier to train the system and we have an online system. In the future I would like to continue to find other possibilities to use statistical methods. I want to compare my text annotation system with other ones, and I want to raise the quality of mine.

V. REFERENCES

[1] Csendes D., Hatvani Cs., Alexin Z., Csirik J., Gyimóthy T., Prószéky G., Váradi "Kézzel annotált magyar nyelvi korpusz : a Szeged Korpusz" *Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2003) kiadványa*, Szeged, pp. 238-247., december 10-11, 2003

[2] Koehn P., "MOSES -a Beam-Search Decoder for Factored Phrase-Based Statistical Machine Translation Models, User Manual and Code Guide", (2009)

[3] Zhifei Li, Chris Callison-Burch: "Joshua: An Open Source Toolkit for Parsing-based Machine Translation" *Proceedings of the Fourth Workshop on Statistical Machine Translation* , pages 135–139, Athens, Greece, March, 2009

[4] Varga D, Halácsy P, Kornai A, Nagy V, Németh L, Trón V, „Parallel corpora for medium density languages”, *Recent Advances in Natural Language Processing Conference*, pp: 590-596, (2005)

Information-retrieval from medical diagnoses and anamneses with text mining algorithms

Ferenc Ott
(Supervisor: Dr. Gábor Prószéky)
ottfe@digitus.itk.ppke.hu

Abstract – In this presentation I introduce a concept of a system and its modules that can be helpful in text mining medical texts, mostly diagnoses and anamneses. A lot of diagnoses and anamneses are reachable in digital format. Text mining can be very helpful for doctors and researchers, but the texts to be used should guarantee anonymity. For this reason some pre-processing steps are needed to substitute real proper names of the documents. New information found with the help of medical text mining can help in identification of diseases in question.

Index Terms – medical diagnoses, anamneses, text mining in medical texts, Latin morphological parsing, medical ontology.

I. INTRODUCTION

The base of the text mining methods is the rich representation of the texts. There are different approaches: rule-based and statistical methods. The representation of the knowledge is the gate between processing structured information and unstructured information.

Four main approaches of text mining methods can be distinguished:

- **Data-oriented definition:** text mining is an operation on symbols of the texts; it tries to find associations with statistical, data-mining methods.
- **Structure-oriented definition:** this method uses morphological approaches.
- **Intelligence-oriented definition:** this method uses ontologies and taxonomies as representations.
- **Business-oriented definition:** this concept is much more pragmatic; it is a system built from modules that work together to grab human readable information from databases, diagnoses, anamneses.

Medical text mining is a special area of general text processing, because medical texts contain a lot of Latin terms or Medical Latin expressions that can be handled by

automatically methods. The Medical Latin language is a special language differing from the Classical Latin language. To analyze a Medical Latin diagnosis or anamnesis, we have to have special methods and pre-processing steps, which I've sold with automatically methods. My aim is to build a system, which can be helpful – by analyzing diagnoses and anamneses – for doctors and for researchers to get more information about diseases, and to be supported in their decision-making. Some typical Medical Latin expressions I've retrieved from existing medical dictionaries.

II. INFORMATION RETRIEVAL FROM MEDICAL DICTIONARIES

I have worked with the following dictionaries:

1. Hungarian-Latin morphological dictionary
M-80500 M 00 805-808 LAPHÁM TUMOROK
M-80502 L 01 Carcinoma papillare in situ
M-80502 V 01 Papillaris in-situ carcinoma
2. Hungarian-Latin topographical dictionary:
T-00400 M 05 nyálkahártyai
T-00400 L 01 tunica mucosa
T-00400 M 01 nyálkahártya
3. Hungarian-Latin procedure dictionary:
P1-95348 M 01 háromosztatúideg-kimetszés
P1-95348 V 01 trigeminálisideg-excizió
4. Hungarian-Latin disease dictionary:
SD4-00A41\$ L 01 multiplicata epiphyseal dysplasia
SD4-00A41\$ V 01 multiplex epifzeális diszplázia
SD4-00A41\$ M 01 többszörös csővescsonti porcvégi rendellenes fejlődésű
5. Multilingual anatomical dictionary:
LA cervicalis [-e]
EN cervical; C. (pertaining to the neck)
DE zervikal; Zervix-; Hals- (zum Hals gehörend)
HU cervikális; nyakhoz tartozó

III. THE MEDICAL LATIN PARSER

The parser is partially written by me. I added special medical affixes to it, like *-isis* or *-aris*. I have also developed a taxonomy-building ability: an XML database has been from the above dictionaries. The parser's input and output are simple text files (see Figure 1.)

```

Latin jelentések:
Első latin jelentés:          abdomen
Második latin jelentés:      alvus
Harmadik latin jelentés:
Negyedik latin jelentés:

Magyar jelentések:
Első magyar jelentés:       has
Második magyar jelentés:
Harmadik magyar jelentés:
Negyedik magyar jelentés:
-----
A(z) filamentum
elemzése és fordítása:
-----
Fordítási eredmény:
Szótag:      filament
Szó:         filamentum
Képző:
Képzőfajta:
Toldalék:   um

```

Figure 1: parsing of word “*abdomen*”

IV. BUILDING A MEDICAL LATIN CONCEPT DATABASE

To make the program understand the Medical Latin expressions, I have built an XML-based expressions database containing the expressions of the above medical (anatomical and pathological) dictionaries. The result is human readable (Figure 2).

```

- <reszSzo>
  <szoto>intervertebral</szoto>
  <szo>intervertebralis</szo>
  <eset>Acc</eset>
  <nem>M</nem>
  <szofaj>M</szofaj>
  <toldalék>es</toldalék>
  <szam>Tobbes</szam>

```

Figure 2: parsing of word “*abdomen*”

In this XML file every expression or a simple word is a <kifejezes>; every part of which is a <reszkifejezes>. If <reszkifejezes> contains more than one word, that is, the expression contains more subexpressions with more than one word, it is a <reszszo> (Figure 2).

V. APPLYING THE BIOLEXDB

One of the biggest medical ontologies, the BioLex database is also available for my research. I am going to use it to identify medical terms in the corpus. This ontology contains linguistic information for millions of English medical words and expressions. (It's been a challenge to convert this database from text format into MySQL), but it works already.) The database is structured in three **layers**:

1. **Target tables:** this layer contains the BioLexicon tables i.e. tables that directly instantiate the BioLexicon DTD.
2. **Staging tables** are tables storing and managing input data before loading them into target tables. They represent a middle tier between the input XML file and the target tables. This layer is dedicated to data cleaning and consistency.
3. **Dictionary tables** are tables used to configure and manage both staging and target tables.

I want to use this ontology to get better performance of medical parsing with my Medical Latin parser and the XML database.

VI. CORPORA

Three corpora will be used to train and test the software. These are English medical corpora for research projects. They are also in XML format, with a training data set of 978 documents, and a testing data set of 976 documents, once with medical codes, and once without these codes. The corpus is easy to work with, because its format is as Figure 3. shows.

```

<text>
<text origin="CCHMC_RADIOLOGY" type="CLINICAL_HISTORY">Follow up right renal duplication and hydronephrosis.</text>
<text origin="CCHMC_RADIOLOGY" type="IMPRESSION">Minimal residual lower pole pyelectasis in the duplex right kidney.</text>
</texts>

```

Figure 3: Format of [10.] English medical corpus

I've started to expand this corpus from medical internet sites. I've written a PHP script to download relevant sites and a script to clean the downloaded files (removing html tags and unusable data). The below sites have been chosen: www.medicinenet.com, www.rare diseases.about.com, www.virtualmedicalcentre.com, www.merck.com

VII. STANFORD PARSER

As a syntactic parser I've used Stanford parser [8.]. It provides the grammatical structure of sentences in a special dependency format. I've found that most important segments of a sentence consist of groups of substantives and a main verb of the sentence. To grab out this parts and analyze them morphologically and statistically, this parser can be very useful. This lexicalized probabilistic parser implements a factored product model, with separate PCFG phrase

structure and lexical dependency experts, whose preferences are combined by efficient exact inference, using an A* algorithm. Another use of the software is running it as an accurate unlexicalized stochastic context-free grammar parser. Either of these yields a good performance statistical parsing system. A GUI is provided for viewing the phrase structure tree output of the parser (Figure 4).

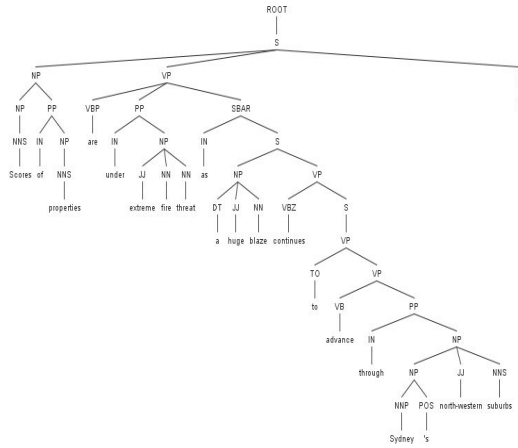


Figure 4: A sample output of the Stanford parser

The parser also has a textual output which can be easily handled with the Prolog logical programming language.

VIII. PROLOG

Prolog is a logic programming language associated with artificial intelligence and computational linguistics. Prolog has its roots in formal logic, and unlike many other programming languages, Prolog is declarative. The program logic is expressed in terms of relations, represented as facts and rules. A computation is initiated by running a *query* over these relations. Prolog can be used to parse natural language texts (Figure 5).

```

<sentence> ::= <stat_part>
<stat_part> ::= <statement> | <stat_part> <statement> <statement>
::= <id> = <expression>
<expression> ::= <operand> | <expression> <operator> <operand>
<operand> ::= <id> | <digit>
<id> ::= a | b
<digit> ::= 0..9
<operator> ::= + | - | *

```

Figure 5: Language description in Prolog

Some Prolog implementations, notably SWI-Prolog, support server-side web programming with support for web protocols, HTML and XML. There are also extensions to support semantic web formats such as RDF and OWL. I use

Prolog for managing the output of the Stanford parser. Because the output is in tree-format, the words and the grammatical information are in brackets, it can easily be handled with this programming language.

IX. WORKING WITH SWI-PROLOG IN TEXT-MINING

SWI-Prolog is a freely usable Prolog implementation. I have tried it on a Hungarian corpus, with a Hungarian morphological parser [9.]. The parser's output was similar, like the Stanford parsers output, so both software result can be handled with the Prolog's methodic. An example of the Prolog's input file (that is the morphological parsers output file) is shown by Figure 6.

```

csomo('COMPL', (null), (null), ['num'-'SG', 'pers'-'P3', 'case'-'ACC'], [
csomo('NP-FULL', 'érték', (null), ['num'-'SG', 'pers'-'P3', 'case'-'ACC'], [
csomo('NP', 'érték', (null), ['num'-'SG', 'pers'-'P3', 'case'-'ACC'], [
csomo('NP-FULL', 'portfolió', (null), ['num'-'SG', 'pers'-'P3', 'case'-'DAT'], [
csomo('NP', 'portfolió', (null), ['num'-'SG', 'pers'-'P3', 'case'-'DAT'], [
csomo('ADJP', 'tőzsde', (null), ['num'-'ni', 'case'-'NOM'], [
csomo('ADJX', 'tőzsde', (null), ['num'-'ni', 'pers'-'ni', 'case'-'NOM'], [
csomo('ONAD', 'tőzsde', 'tőzsdei', ['case'-'NOM', 'pers'-'P3', 'num'-'SG'], [
csomo('NX', 'portfolió', (null), ['num'-'SG', 'pers'-'P3', 'case'-'DAT'], [
csomo('N', 'portfolió', 'portfoliójának', ['case'-'DAT', 'pers'-'P3', 'num'-''], [
csomo('NP', 'érték', (null), ['num'-'SG', 'pers'-'P3', 'case'-'ACC'], [
csomo('NX', 'érték', (null), ['num'-'SG', 'pers'-'P3', 'case'-'ACC'], [
csomo('N', 'érték', 'értékét', ['case'-'ACC', 'pers'-'P3', 'num'-'SG'], [

```

Figure 6: Morphological output

The Prolog program grabs out the relevant words from the text and writes them out to a text file (part):

```

mondat_fonev([A|L], Gyujto, MondatFonevk) :-
A = csomo(Tagok, Szotari, _, _, Gyerekek),
(Tagok == N,
hozzaad(Szotari, Gyujto, Gyujto1),
!,
mondat_fonev(Gyerekek, Gyujto1, Gyujto2),
mondat_fonev(L, Gyujto2, MondatFonevk)
);
!,
mondat_fonev(Gyerekek, Gyujto, Gyujto1),
mondat_fonev(L, Gyujto1, MondatFonevk)
).

```

X. CONCLUSION

My goal is to develop an efficient algorithm and system to automatically process medical texts. The system will consist of the following modules:

- a Medical Latin parser (which I have finished),
- an XML-based expressions-database, which I have also finished)
- the MySQL-based BioLexDB
- the Stanford-parser (what I have tested this year)
- a medical corpus (I have started to build partly from internet sites and partly from still ready corpora)

The downloader script and the text-cleaning script are ready. The parser is also a good tool to analyze the data. My Latin parser and my medical XML database from medical Latin dictionaries serve as a basis of the system. The BioLexDB can be a good add-in, the Prolog NLP functionality makes easier to deal with the output of the morphological analyzer.

REFERENCES

- [1.] Sholom M. Weiss, Nitin Indorkhya, Thong Zang, Fred J. Damerou: *Text Mining, Predictive Methods for Analyzing Unstructured Information*, Springer 2005.
- [2.] Brian Roark and Richard Sproat: *Computational Approaches to Morphology and Syntax* 2008 Massachusetts Institute of Technology, September 2008, Vol. 34, No. 3, Pages 453-457.
- [3.] Tikk Domonkos, Biró György, Szidarovszky Ferenc P., Kardkovács Zsolt T., Héder Mihály és Lemák Gábor. *Magyar internetes gazdasági tematikájú tartalmak keresése*. In IV. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY-06) pp. 3–14, Szeged, 2006.
- [4.] György Szarvas, Richárd Farkas, Róbert Busa-Fekete: *State-of-the-art anonymisation of medical records using an iterative machine learning framework*. Journal of the American Medical Informatics Association, 2007 Vol. 14. pp 574-580
- [5.] Nagy József: *Orvosi latin nyelvi alapismeretek*, Medicina, 2009
- [6.] D. Tikk, Zs. T. Kardkovács, Z. Andriská, G. Magyar, A. Babarczy, and I. Szakadát. *Natural language question processing for Hungarian deep web searcher*. In Proc. of ICC-04, 2nd IEEE Int. Conf. on Computational Cybernetics, pp. 303–309, Vienna, Austria, 2004.
- [7.] Pestian, JP, C Brew, P Matykiewicz, DJ Hovermale, N. Johnson K. Bretonnel Cohen, W. Duch; *A Shared Task Involving Multi-label Classification of Clinical Free Text*, *Proceedings ACL:BioNLP*, Prague, June 2007
- [8.] <http://nlp.stanford.edu/software/lex-parser.shtml>
- [9.] Morphologic HumorESK C library
- [10.] <http://www.computationalmedicine.org/>

Fast computation of particle filters on topographic processor arrays

András Horváth

(Dr. Miklós Rásonyi and Dr. Tamás Roska)

horan@digitus.itk.ppke.hu

Abstract—During the first two semesters WE have developed a new variant of the particle filter algorithm for estimating a signal from noisy observations. It suits ideally implementation on a cellular processor array. The error of the new algorithm is essentially the same as that of the old one but it runs much faster, especially when there is a large number of particles to be simulated.

Index Terms—Particle filter, Processor array

I. INTRODUCTION

Sequential Monte Carlo methods arose for the computation of optimal estimates in nonlinear and non-Gaussian state-space models where analytic solutions are not available. They found applications in diverse areas such as localization, navigation, tracking, robotics and signal processing, see e.g. [8] for a representative sample. More recently they have been applied in financial mathematics (to stochastic volatility models and to the computation of credit losses, see [9] and [4]). For the mathematical theory, consult [7].

Here we restrict ourselves to particle filters with resampling (the so-called bootstrap filters). These filters and their various ameliorations (see e.g. [1]) proved to be an efficient tool for computing certain conditional expectations.

The study of particle filters is of particular interest as they are inherently unsuitable for parallel computing (see, however, [2] for related results). We have found a new variant of this algorithm that could be implemented on an array of processors and, using parallelism and local communication, could greatly enhance computational speed without substantial loss in precision.

I will describe the models we are dealing with. Then I present the results of test runs which have been carried out on a virtual machine, emulating a processor array that could actually be manufactured. I also review the test runs we have performed on existing architectures (Xenon chip).

It is already clear that the basic idea of the new algorithm may be applied to a much more general class of interacting particle systems. This is subject of current research.

II. HIDDEN MARKOV MODELS AND PARTICLE FILTERS

A. Model specification

The investigated model can be described by two stochastic processes: one contains the hidden states x_t of the system at time $t = 0, 1, \dots$ and the other the corresponding series of

observations y_t , $t = 1, 2, \dots$. We assume that x_t follows a Markovian dynamics given by the recursion

$$x_{t+1} = \varphi(x_t, e_1(t+1)) \quad (1)$$

where $\varphi : \mathbf{R}^2 \rightarrow \mathbf{R}$ is a fixed (nonlinear) function and $e_1(t)$ is an IID sequence independent of the initial state x_0 . The transition law of x_t will be denoted by $Q(v, dw)$, that is

$$P(x_1 \in A | x_0 = v) = \int_A Q(v, dw)$$

for sets $A \subset \mathbf{R}$ and for any $v \in \mathbf{R}$.

The observations are assumed to be a function of the system state blurred by additive noise, that is

$$y_t = \psi(x_t) + e_2(t) \quad (2)$$

for some (nonlinear) function $\psi : \mathbf{R} \rightarrow \mathbf{R}$ and an IID noise sequence $e_2(t)$, independent of $e_1(t), x_0$. We denote by $r(w)$ the density function of the law of $e_2(t)$. Then $r(y_t - \psi(x))$ is the “likelihood function” expressing how x is likely to be the state of the system at time t when the observation is y_t .

The pair of processes (x_t, y_t) is a typical example of a hidden Markov model, see [10]. In applications one tries to compute

$$E[x_t | y_t, \dots, y_1],$$

which is the best least-squares estimate of the hidden state x_t based on the information y_t, \dots, y_0 available at time t .

There are various implementations of particle filters, but they usually contain the following four steps.

B. Initialization (step 0)

A more detailed description can be found in [1] or in [9]. We will simulate N particles whose trajectories follow the state dynamics but are subject to a selection mechanism based on observations. We first draw initial values

$$\xi_0^i = \zeta_i \quad (3)$$

Each particle will have a weight that represents its accuracy/distance from the real state, we first set $w_0^i = 1/N$ so all the initial weights are assumed to be equal. (We do not have any observation about the system yet, hence this seems to be a reasonable choice.)

C. Error calculation (step 1)

Let us assume that we have already generated the trajectories of the particles ξ_s^i , $i = 1, \dots, N$ and $s \leq t$. We now have to calculate the “fitness” of each particle, based on the next observation. We set $E_t^i = y_t - \psi(\xi_t^i)$ for the “error” of particle i in the light of the observation y_t .

D. Resampling (step 2)

Set the new weights for every particle according to the likelihood function above:

$$w_t^i = r(E_t^i) = r(y_{t+1} - \psi(\xi_t^i)), \quad (4)$$

then normalize the weights:

$$w_t^i = \frac{w_t^i}{\sum_{j=1}^N w_t^j}. \quad (5)$$

(The normalization makes the resampling easier. The weights will be set back to equal values at step 3.)

We choose our new set of particles by drawing from our sample:

$$\hat{\xi}_t^i := \xi_t^{\eta(U_i)}, \quad (6)$$

where the U_i are IID random variables uniformly distributed on $[0, 1]$ and $\eta : [0, 1] \rightarrow \{1, \dots, N\}$ assigns a particle (an index) to every random variable such that $P(\eta(U_i) = j) = w_t^j$, for $i, j = 1, \dots, N$.

E. Iteration (step 3)

Make one step ahead with all the particles according to the rule in the model:

$$\xi_{t+1}^i = \varphi(\hat{\xi}_t^i, e_1^i(t+1)). \quad (7)$$

here the $e_1^i(t+1)$ are N IID copies of $e_1(t+1)$.

Often it is more effective to generate ξ_{t+1}^i using a law other than that of $e_1(t)$ and then correct this bias by assigning appropriate weights to these new particles. With this neat trick (called “importance sampling”) we can ‘lead’ the particles towards a prescribed region where we are the most interested in their behavior, see e.g. [1].

Then We reset the weights to their original (equal) values:

$$w_{t+1}^i = \frac{1}{N}. \quad (8)$$

After this we return to steps 1-3 and generate ξ_t^i for all the time points $0, \dots, T$ and for all $i = 1, \dots, N$. If T and N are large enough then the (discrete) distribution of the particles ξ_T^i , $i = 1, \dots, N$ is hoped to approximate μ_T fairly well. In formulas, denoting by δ_w the one-point mass at w , we have

$$\frac{1}{N} \sum_{i=1}^N \delta_{\xi_T^i} \approx \mu_T.$$

Hence one may take

$$\frac{1}{N} \sum_{i=1}^N \xi_T^i \approx E[x_T | y_T, \dots, y_1]. \quad (9)$$

to estimate the hidden state x_T .

III. PARALLELIZED PARTICLE FILTER

The algorithm sketched in the previous section has the drawback that comes from the resampling step, where all the weights/states of the particles has to be collected. This makes the algorithm non-parallelizable and hence slow, especially when a large number of particles need to be simulated. This occurs in the case of a high-dimensional state vector x_t .

We are now introducing an algorithm where the resampling step is based only on local communication:

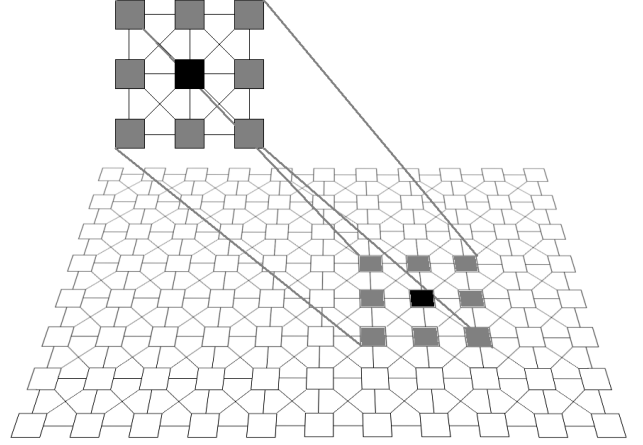


Fig. 1. The small set, the neighborhood of a given processor (representing a particle during the algorithm), that determines the calculations. The grid structure represents how the information spreads out from processor to processor, from neighborhood to neighborhood.

The resampling happens between small sets N_i of particles only. Every particle i communicates with its $|N_i| - 1$ neighbouring/surrounding particles (that can be reached in one clock cycle). When particles (represented by one processor each) are imagined to sit on a rectangular grid, a generic processor (particle) will communicate with 8 neighbors, so $|N_i| = 9$, see Figure III. (Those on the sides of the rectangle obviously communicate with less.) Initialization, error calculation and iterations steps are carried out in exactly the same way as described before.

The only, but crucial alteration takes places at step two. This tiny modification is extremely important, because it makes the algorithm parallelizable.

A. Resampling (step 2) in the localized particle filter

We have to calculate the sum of the weights in a neighborhood (this takes place in processor i based on the information sent to it from processors in N_i):

$$W_t^i = \sum_{j \in N_i} r(E_t^j), \quad (10)$$

and set the new weights $h_t^j(i) := r(E_t^j)/W_t^i$ for $j \in N_i$.

Processor i will select a new particle from ξ_t^j , $j \in N_i$ using the weights $h_t^j(i)$:

$$\hat{\xi}_{t+1}^i = \xi_t^{\theta_i(U_i)}, \quad (11)$$

where U_i are IID uniformly distributed on $[0, 1]$ and $\theta_i : [0, 1] \rightarrow N_i$ selects one of the particle from the neighborhood such that $P(\theta_i(U_i) = j) = h_i^j(i)$, $i = 1, \dots, N$ and $j \in N_i$.

The “best” or “fittest” particles will thus be gradually diffused over the whole grid.

Remark. One of the main advantages of the localized particle filter reveals itself after the previous steps and calculations. At this point we have all the particles suitably distributed to approximate μ_t distribution, but to get the desired result we have to count the average of the states, see (9). This step is unavoidable indifferent of the method we had used to generate ξ_T^i .

Such a calculation can be done fast and easily on a cellular-like architecture. We start a ‘horizontal’ wave from the left border of the grid, and spread it to the other end of the chip summing all the states, with this we make an average at the right border. After this we can spread a similar ‘vertical’ wave that will calculate the average in one of the corners. With this method the calculation of (9) can be done in $O(\sqrt{N})$ steps, which is much better than the usual $O(N)$.

IV. RESULTS

We tested the new algorithm on two commonly used “benchmark” models, well described in other papers, e.g. [1].

The algorithms we compared were the particle filter with resampling (PFR henceforth, also known as bootstrap filter; see section II above); our new variant of it in section III (NEW) and the method of [3] (BOL). The first algorithm was implemented on a simple dual-core processor PC (intel T6570); the second on a virtual machine emulated on the same PC. The virtual machine could physically be realized as a square (or rectangular) array of appropriate processors. The number of processors coincides with the number of particles to be simulated.

The tests were carried out and averaged on 1000 different simulated trajectories of x_t, y_t with $t = 0, \dots, 100$. We took as a measure of error the root of the sum of the squared differences between the true trajectories and their estimated counterparts generated by the respective algorithms.

The test runs were also carried out on an existing CNN-type architecture : the emulated version of the Xenon-v3 chip, see [11] and [12]. Although this chip is designed for image processing purposes, and can process data only with 8-bit accuracy, the given two examples shows, that this still gives comparable performance thanks to the robustness of the algorithm.

The first table contains the results for the first model:

particles	BOL	PFR	NEW	Xenon
16	87.43	66.42	77.38	90.59
36	75.75	55.44	61.61	71.70
49	67.55	53.66	57.17	68.56
64	64.41	52.56	54.65	66.44
81	61.02	51.65	53.00	65.04
100	59.14	50.98	51.58	64.42
144	56.31	50.51	49.69	63.39
225	53.83	50.01	48.44	62.16
400	51.82	49.49	47.61	62.14
625	51.09	49.16	47.08	62.11
900	50.60	49.13	47.01	62.10

The first column contains the number of the particles, the rest contains the mean-square errors of the previously mentioned methods (in order from left to right: ‘distributed resampling algorithm with non-proportional allocation and local exchange’; the ‘classical’ particle filter, the simulation of the localized particle filter on a 32-bit architecture; the results of the localized particle filter on the Xenon architecture).

The results for the second model:

particles	BOL	PFR	NEW	Xenon
16	1.120	0.618	0.584	0.659
36	0.888	0.564	0.538	0.582
49	0.835	0.557	0.529	0.570
64	0.797	0.550	0.522	0.562
81	0.785	0.547	0.518	0.554
100	0.762	0.545	0.516	0.551
144	0.706	0.542	0.511	0.5467
225	0.702	0.539	0.508	0.5442
400	0.701	0.538	0.505	0.5440
625	0.693	0.537	0.504	0.5440
900	0.680	0.536	0.502	0.5438

We can see that the new algorithm outperforms all the other variants.

Our method’s special advantage is that, with almost the same accuracy, its speed does not depend on the number of particles. This latter is limited only by the architecture of the chip (by the number of processors). Calculations of individual particles can be done separately in a parallel way, and the information can be locally distributed amongst the cellular connections for the censoring step.

The runtime of the above calculations is given below:

part	T PFR	T NEW	PFR/NEW	T Xen	PFR/Xen
36	0.143	0.215	0.67	12	0.01
100	0.421	0.615	0.68	12	0.04
225	0.901	1.330	0.68	12	0.08
4096	16.417	24.495	0.67	12	1.37

The first column contains the number of the particles, the second column contains the estimation time (in seconds) for 100 steps for one trajectory, with the ‘regular’ particle filter, the third column contains the same time for the new particle filter, on a regular architecture, without any parallelization (calculating the algorithm on N cores the processing time would be decreased to TNEW/N, because the algorithm is fully parallelizable). The fourth column contains the ratio between

the two method. /The time measurements are basically the same for the two models/ The fourth column contains the ratio between the two previous runs. The fifth contains the run-time of the algorithm if it were implemented on a real Xenon chip. According to the compiler of the chip, one step of the calculation could be done with 1248 clock cycles, which is approximately 120 ms, so proceeding through 100 steps would take about 12 seconds. The last column shows the ratio between the 'original' algorithm and our version implemented on a multi-parallel architecture.

It can be seen, that with lower number of particles the calculation of 'regular' particle filter method takes less time, however our methods outperforms it in tasks where high number of particles should be used, because the processing time is independent from the number of particles, it based only on the architecture. The estimation of a quite complex system (especially in several dimensions) requires an increased number of particles to preserve the accuracy, and this is the most serious drawback of the original particle filter algorithm. That's why we have also included the case of 4096 particles in the above table. Such large number may become necessary when the state x_t is high-dimensional (see e.g. [14] where a real-life example requires 27 dimensional state vector). The present note is about highlighting the novelty of the algorithm, real-life multi-dimensional models will be subject of our future research. Nonetheless one can see that even on the emulator (where there are no parallel chips) a runtime comparable to the original algorithm can be observed(especially with high number of particles). This shows that, in the case of a large number of particles, our algorithm is worth implementing.

We also have to underline, that the Xenon chip was not designed for state approximation. With a special-purpose processor containing all the necessary operations in an optimized form the processing time could be decrease drastically.

We have also tested or algorithm on a new real-life problem. with a three-dimensional state-space. Our method outperformed the regular particle-filter even in this case (we had to apply a few technical tricks, but these can not be described in details because of the lack of space). The error of our method and in case of their regular method can be seen in the following table

num	Schon	New N4	New N7	New I4	New I7
E625	102.68	80.99	73.51	81.14	74.24
T625	3.84	10.86	26.02	7.76	12.34
E900	94.30	79.44	71.41	79.67	72.02
T900	5.55	16.03	38.85	11.19	17.80
E1225	89.34	78.71	70.09	78.89	70.53
T1225	7.56	21.18	50.28	15.43	24.51
E1600	85.49	77.94	69.26	78.16	70.05
T1600	9.89	27.65	65.65	20.35	32.35

V. CONCLUSION

To sum up, our method operates with parallelism and local interactions. Essentially the same approximation can be reached while decreasing the runtime drastically. The algorithm takes advantage of the structure of cellular neural networks, see [13] and [6].

The present study shows, that this algorithm – implemented on cellular architecture like the Xenon-v3 or on a more specific locally coupled chip network – could provide a solution to many problems where the state of a complex system should be estimated with high accuracy within a very strict time limit, e.g. when estimating the position of an aircraft.

This method shows that with the proper parameters (number of particles, and neighborhood size), we can create an extremely good balance between information preserving -to avoid particle impoverishment-, and information spreading -to create good estimations-. The optimal parameters can be easily found with a few simulations for a known model. With this method we can outperform the regular particle filters, in error rate, we can ensure a lower mean square error, our for a practical task, we can make our estimation faster with the same error, our create better estimation with the same temporal resolution.

The algorithm can be used even in case of regular single-core architectures, but they are inherently designed for multi-core, paralell architectures with local interconnections, where the method can be boosted, without increasing the error rate.

REFERENCES

- [1] M. S. Arulampalam, S. Mksell, N. J. Gordon, and T. Clapp. A tutorial on particle filters for online non- linear/non-Gaussian Bayesian tracking. In [8], 174–188.
- [2] Bolić, M. *Architectures for the implementatin of particle filters*. PhD thesis, Stony Brook University, Stony Brook, NY, USA, 2004.
- [3] Bolić, M., Djurić, P. M. and Hong, S. Resampling algorithms and architectures for distributed particle filters. *IEEE Trans. Signal Processing*, Vol. 53, 2442–2450, 2005.
- [4] Carmona, R., Fouque, J.-P. and Vestal, D. Interacting particle systems for the computation of rare credit portfolio losses. *Finance and Stochastics*, Vol. 13, pp. 613–633, 2009.
- [5] Chigansky, P., Liptser, R. and van Handel, R. Intrinsic methods in filter stability. *To appear in "Handbook of Nonlinear Filtering"*, Oxford University Press, 2009.
- [6] Chua LO and Roska T. The Cnn Paradigm . *IEEE Transactions on Circuits and Systems I - Fundamental Theory and Applications* 40:(3) pp. 147–156, 1993
- [7] Del Moral, P. *Genealogical and Interacting Particle Systems with Applications Feynman-Kac Formulae* . Springer, 2004.
- [8] Djurić, P.M. and Godsill, S. J., guest editors). Speical issue on Monte Carlo Methods for Statistical Signal Processing *IEEE Transactions on Signal Processing*, Vol. 50, Feb. 2002.
- [9] Doucet, A. and Johansen, A. M. A tutorial on particle filtering and smoothing: fifteen years later. *To appear in: Oxford Handbook of Nonlinear Filtering*, Oxford University Press, 2009.
- [10] Ephraim, Y. and Merhav, N. Hidden Markov Processes. *IEEE Transactions on Information Theory*, Vol. 48, 1508–1569, 2002.
- [11] Földesy, Péter, Zarándy, Ákos, Rekeczky, Csaba, Roska, Tamás, Digital implementation of the cellular sensor-computers *International Journal of Circuit Theory and Applications*
- [12] Földesy, Péter, Zarándy, Ákos, Rekeczky, Csaba, Roska, Tamás, High performance processor array for image processing, *IEEE International Symposium on Circuits and Systems*. New Orleans, 2007
- [13] Roska T, Chua LO, The CNN universal machine: an analogic array computer, 3rd ed. *IEEE Transactions on Circuits and Systems II-Analog and Digital Processing* 40:(3) pp. 163-173, 1993.
- [14] Schön, T., Gustaffson, F. and Nordlund, P.-J. Marginalized Particle Filters for Mixed Linear/Nonlinear State-Space Models. *IEEE Transactions on Signal Processing*, 53 (7), pp. 2279–2289, July 2005.

Mapping Mathematical Expressions into FPGA Devices Using Data-Flow Graphs

Csaba Nemes

(Supervisors: Dr. Péter Szolgay and Dr. Tamás Roska)
nemcs@digitus.itk.ppke.hu

Abstract—In this paper a new design methodology is presented which targets to optimize the control units used in FPGA implementation of mathematical expressions. Nowadays as the computation time of an execution unit is comparable with the communication delay it is not enough to find the optimal netlist of execution units which implements a given mathematical expression but also the control units shall be optimized. The usage of distributed control units results in a speed gain contrast to the global control where the fan out of the wiring can slow down the operation of the whole circuit. In the presented design methodology the execution units are partitioned and locally distributed control units are assigned to each partition. An optimization problem is described and an algorithm is developed which targets to find the optimal partitioning where fast local control units can be used with relatively small area increase. The optimal solution of the partitioning problem is NP complete[1] but a reasonable algorithm can be constructed for practical engineering applications. The operation of the algorithm is demonstrated on a few test cases.

Index Terms—hypergraph partitioning; FPGA; distributed control unit

I. INTRODUCTION

Having a computational problem defined on 2D or 3D array ($N \times M$, $N \times M \times L$) and the operation on every element is described as a mathematical expression, acyclic data flow graph or UMF diagram [2]. The problem to be solved is how to map a computational problem on a virtual array to a given physical FPGA where area/processor (logic slices, DSP slices), on-chip memory (BRAM) and off-chip memory bandwidth are limited. Depending on the complexity of the operator a small amount of physical execution units can be implemented $n \ll N \times M$ (in 2D case) or $N \times M \times L$ (in 3D case). The operator can be decomposed into small basic blocks which use either the logic resources (such as adders) or the dedicated resources (embedded multipliers) of the FPGA. The result of this process is a Physical Cellular Machine optimized for the given application. The optimization can be focused on speed, area, accuracy etc. Main components are the on-chip memory and the specialized execution unit.

Using current high speed DDR2/3 SDRAM and SRAM memories data read and write operations can be carried out in consecutive bursts. Additionally the available memory bandwidth might be fluctuating, therefore the execution unit should be halted during the computation if no data available.

The simplest and most area efficient solution of this problem is to use one global control unit to monitor the state of the I/O buffers and enable the operation of the entire system by using

a global enable signal. The global enable signal has very high fan-out and is hard to route even if global wires are available on FPGA. As wire delay dominates over gate (LUT) delay on the current state-of-the-art FPGAs this solution results in very low operating frequency.

One possible solution of this problem is to create a data driven pipeline where a basic processing unit is halted automatically when no input data is available or the results cannot be processed by the next unit. Therefore local control unit can be added to every operator (adder, multiplier). In this case the control units are the simplest, but area requirements are significantly increased by the large number of FIFOs.

Alternative solution is to share the control unit among several basic processing units thus the FIFO buffers inside the groups can be eliminated significantly reducing area requirements. Determining the parameters of the groups carefully significant loss in operating frequency can be avoided.

II. RESOURCES ON AN FPGA

The main configurable element of the new Xilinx Virtex family[3] is the Advanced Silicon Modular Block (ASMBL). The architecture is column based where each ASMBL column has specific capabilities, such as logic, memory, I/O, DSP, hard IP and mixed signal. By using different mix of the ASMBL columns domain specific devices can be manufactured. Currently four families are available optimized for different application areas: logic intensive (LX), logic intensive with serial transceiver (LXT), high performance DSP with serial transceiver (SXT), and embedded processing (FXT). Due to the smaller transistor dimensions the total net delay is mainly determined by the wire delay, hence the CLBs of the Virtex-5 architecture are completely redesigned. In the new architecture traditional 4-input LUTs are replaced by 6-input LUTs. Each CLB is divided into two slices and every slice contains 4 6-input LUTs, 4 registers, and carry logic.

In the new FPGAs the simple multipliers are replaced by complex DSP blocks called XtremeDSP (DSP48E) slices. The heart of the DSP48E is a 25bit by 18bit 2's complements signed multiplier. It also contains a 48bit ALU unit with optional registered accumulation feedback. Additionally, hard-wired 17 bit shift capability simplifies the construction of large multipliers, while optional pipeline registers enable even 550MHz operation. The currently available largest Virtex-5 device contains 1056 DSP48E slices, while the largest

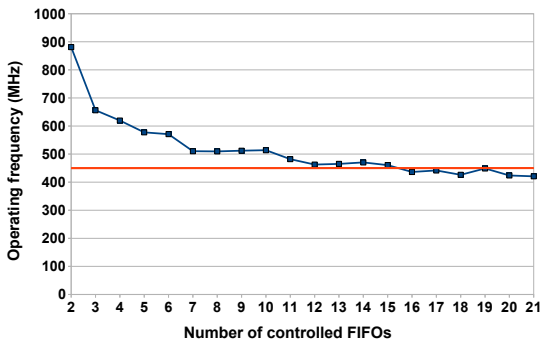


Fig. 1. Operating frequency of the control block. Red line indicates the operating frequency of the multiplier unit.

member of the recently introduced Virtex-6 family contains 2016 DSP48E slices.

III. SYSTEMC AND MATHEMATICAL REPRESENTATION

Using a high-level description language the computationally intensive algorithm can be efficiently described by using the data-flow model. This model can be transformed to an abstract mathematical graph representation, where operations and connections are represented by nodes and arcs of the graph respectively.

In case of distributed control units data driven pipelines are created where basic processing units are halted automatically when no input data is available or the results cannot be processed by the next unit. Synchronization of the processing elements is done by using FIFO buffers.

If we assign one control unit to every processing unit and attach a FIFO to each of its output we obtain a restricted case of the Kahn process networks [4] where independent processes are communicating over bounded FIFO channels. Each processing unit can be treated as a process because it reads its input from a FIFO attached to previous processor or the memory interface and writes the updated result into a FIFO. The enable signal and the FIFO control signals are local to the given processor; their state depends on the state of the connected FIFOs.

However it is more practical to partition the processing units into groups where each group has one control unit. In this case FIFOs inside the locally controlled groups can be omitted. The disadvantage of this solution is that each control unit should handle more FIFOs, which results in more complex control logic and smaller operating frequency as shown in Figure 1.

There is a design trade of between the speed of the control unit and the area requirements of the circuit. Larger partitions have more inputs and outputs which results in a slower control unit, on the other hand small partitions increase the area requirements of the circuit because of the overhead generated by the larger quantity of FIFOs. We can formulate an optimization problem where we would like to enlarge the

size of the locally controlled groups as long as the operating frequency of our control blocks does not limit the operation of the entire circuit.

Mathematical formulation of the partitioning problem:

- 1) Optimization: Make partitions from the nodes of a directed acyclic hyper graph where the number of cut edges are minimal. (In this case a hyperlink can be cut many times.)
- 2) Constraint: Any given partition shall have less than or equal connection to the other partitions than a given upper threshold.

Finding the optimal solution is an NP-complete problem, however our goal is to find a reasonable solution in polynomial time. The result of the algorithm will be an optimized graph, where the control is local and data driven. According to our experiments one control unit can handle 10 input/output FIFOs without decreasing the expected 450MHz operating frequency of the entire data path significantly (see Figure 1).

In our implementation the input of the algorithm is an initial solution of a computationally intensive task implemented in C++ via the SystemC library [5]. The only restriction for the implementation is that the classes of the modules have to inherit our interfaces as well. This is an elegant way to extend the SystemC model with some extra information without the modification of the library itself. While the modeling features of the SystemC library are still available, the extra information is used to build up the graph representation via the Lemon Graph Library [6].

IV. THE PROPOSED PARTITIONING ALGORITHM

Hereby I propose a heuristic greedy algorithm for the previously described partitioning problem. While the pseudo code of the algorithm is shown in Algorithm 1. the key steps are also summarized below:

- 1) The input of the algorithm is a directed acyclic hyper-graph.
- 2) Based on the delays of the arithmetic units different levels can be assigned to the nodes. These levels indicate the number of cycles required for the data to reach the given node through the pipeline.
- 3) Based on the levels the nodes are numbered. The order of two nodes on the same level are arbitrary but nodes on lower level always have lower numbers than the ones on higher levels.
- 4) The algorithm starts from the lowest-numbered node and a subgraph is grown from this node. After the subgraph cannot be grown further the first partition is created from the nodes of the subgraph.
- 5) In every upcoming iteration the lowest-numbered node id selected which has not been partitioned yet. If there are no more unpartitioned node the algorithm ends.
- 6) The subgraph is also created iteratively. In the first iteration the subgraph contains only one node. In every upcoming iteration the algorithm tries to deepen the subgraph by selecting one node "below" the subgraph

Algorithm 1

function runPartitioning(G,T):

- 1: Assign a number to each node, which indicates the number of cycles required for the data to reach the given node through the pipeline.
- 2: Sort the nodes based on the assigned numbers.
- 3: **repeat**
- 4: Create a new partition P.
- 5: Move the lowest-numbered node, which is unpartitioned to P.
- 6: growSubgraph(P,T)
- 7: **until** there is any unpartitioned node

function growSubgraph(P,T):

- 1: **for all** N nodes which gets input from one of the nodes of P **do**
 - 2: $P1 := P$
 - 3: Move N to partition $P1$.
 - 4: Move all ancestor nodes of N which is unpartitioned to $P1$.
 - 5: **end for**
 - 6: **if** the number of the incoming and outgoing arcs of the subgraph $\leq T$ **then**
 - 7: $P := P1$
 - 8: growSubgraph(P,T)
 - 9: **break**
 - 10: **end if**
-

and includes all the ancestors of the selected node which have not been partitioned yet.

- 7) Extension of the subgraph is only successful if the number of incoming and outgoing arcs of the subgraph does not exceed the upper threshold.
- 8) If no nodes can be selected below the subgraph which is suitable to a successful extension the subgraph cannot be extended and the algorithm is continued from step 4.

V. EXAMPLES

A. Simple SystemC code and its graph representation

Simple SystemC code fragment and an equivalent graph representation is shown in Figure 2 and Figure 3 respectively. Sum and Prod modules are base modules which are already implemented in SystemC. Instances of these modules will be the nodes of the graph. Signal1 and signal2 are wires corresponding to the interconnections.

B. Base template operation of the CNN state equation

Our first test case was the frequently used 3x3 template operation of the CNN state equation [7]. It can be regarded as a convolution with a 3x3 kernel where 9 state values should be multiplied by 9 template parameters. The final result is computed by summing the partial results and the old state value. However this example is relatively simple and has a limited size, it is ideal for the demonstration of our algorithm. The result of the algorithm is shown in Figure 4. The first

```
1: ...
2: Prod<INSIZE,INSIZE,OUTSIZE> p1,p2;
3: Sum<INSIZE,INSIZE,OUTSIZE> s1;
4: ...
5: p1->c(signal1);
6: p2->c(signal2);
7: s1->a(signal1);
8: s1->b(signal2);
9: ...
```

Fig. 2. Simple SystemC code snippet

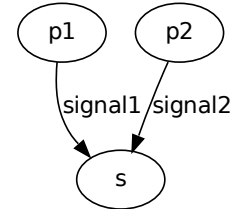


Fig. 3. Simple data-flow graph generated from Figure 2.

partition (partition#1) is grown from the top-left node of the graph (node 1). The algorithm extends the partition by stepping one level down and recursively selecting all the ancestors of the newly founded node in every iteration, therefore nodes 2 and 12 are added to the partition in the first iteration. Partitions are extended as long as the number of input/output arcs are still smaller than 10, which was determined earlier as an upper threshold. In this example the first two extensions are successful. the third extension is failed because the extended partition would have 17 input/output arcs. The second partition (partition#2) is grown from the next non-partitioned node (node 5) which has the lowest number.

Area requirements and operating frequency of the optimized control unit are shown on Table I. The optimized control unit requires less additional area than fully distributed control unit but operates on higher frequency than global control unit. The area of input/output and inside FIFOs is shown separately to emphasize the difference between them. Input/output FIFOs are used in both global and locally distributed control unit however inside FIFOs are only used in the last one.

TABLE I
IMPLEMENTATION RESULTS OF THE CONTROL UNIT IN CASE OF CNN
STATE EQUATION

		Global control unit	Fully distributed control unit	Optimized control unit
FIFO Area (slice)	Input/output	549	549	549
	Inside	0	855	98
	All	549	1404	647
Clock frequency (MHz)		423,908	881,057	512,032

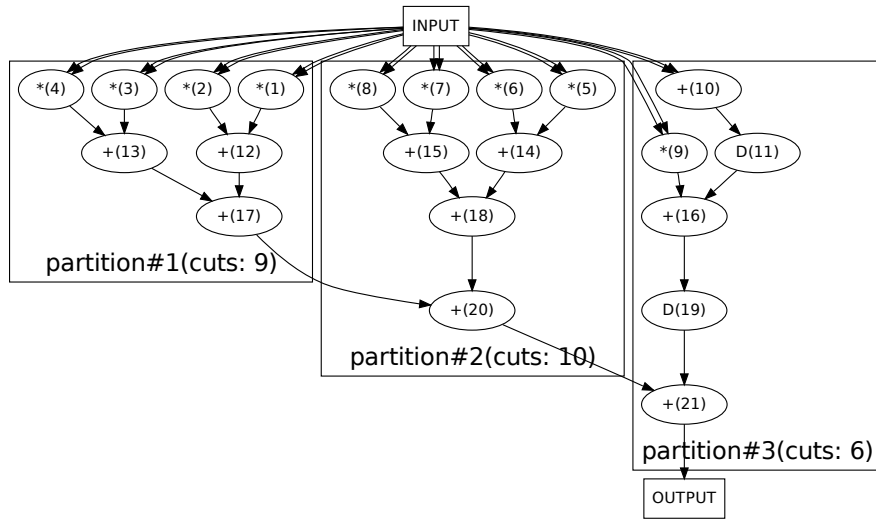


Fig. 4. Result of the partitioning algorithm on the graph of the CNN template operator. The total number of cut arcs is 23.

C. Computational Fluid Dynamics(CFD)

The second test case was a representative part of the arithmetic unit implemented for simulation of computational fluid dynamics(CFD) in [8]. The data-flow graph representation of this example is more complex containing hyperarcs as well. Hyperarcs represent data connections where one source module supplies several target modules with the same parameter.

The figure of the partitioned graph is not presented here because of its relatively big size, however the area requirements and operating frequency of the optimized control unit are shown on Table II. The same conclusion can be drawn as in the previous example. The optimized control unit requires less additional area than fully distributed control unit but operates on higher frequency than global control unit. The area of the input/output FIFOs is different in each case because hyperarcs with more than two nodes are interpreted differently. In the global control unit every hyperarc is replaced with only one FIFO, as they represent one input parameter which can be controlled globally. In the distributed control unit hyperarcs are replaced with as many FIFOs as the number of the different partitions which get the given input parameter.

VI. CONCLUSION AND FUTURE WORK

A design methodology is described to implement customized high-performance data-flow architectures using dis-

tributed control unit. A data-flow graph of a mathematical expression constructed from a high-level language description is given. An optimization problem with a special constraint has been described to efficiently partition the execution units between control structures without significantly increasing the area of the circuit or decreasing the operating frequency. A heuristic algorithm has been proposed to give an affordable solution. The operation of the algorithm was presented by optimizing the implementation of two practical examples. In both cases the number and the size of the FIFOs are optimized without limiting the speed of the overall circuit.

The netlist partitioning in VLSI design is well studied in the literature, however the introduced special constraint is unique and not applicable directly with the available partitioning methods. In the future I am planning to modify and test the available partitioning methods[9][10] including heuristic move-based and spectral algorithms.

REFERENCES

- [1] *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman, 1979.
- [2] T. Roska, "An overview on emerging spatial wave logic for spatial-temporal events via cellular wave computers on flows and patterns," *Proc. of NOLTA 2008*, pp. 98–100, 2008.
- [3] "Xilinx product homepage," <http://www.xilinx.com>, 2010.
- [4] T. M. Parks, "Bounded Scheduling of Process Networks," Ph.D. dissertation, University of California at Berkeley, 1995.
- [5] "The open systemc initiative," <http://www.systemc.org/>, 2010.
- [6] "Lemon graph library," <http://lemon.cs.elte.hu/trac/lemon>, 2010.
- [7] P. Nagy, Z. Szolgay, "Configurable multilayer cnn-um emulator on fpga," *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on*, pp. 774 – 778, 2003.
- [8] S. Kocsárdi, Z. Nagy, A. Csík, and P. Szolgay, "Simulation of two-dimensional supersonic flows on emulated-digital cnn-um," *EURASIP J. Adv. Signal Process*, vol. 2009, pp. 1–11, 2009.
- [9] C. J. Alpert and A. B. Kahng, "Recent directions in netlist partitioning: a survey," *Integr. VLSI J.*, vol. 19, no. 1-2, pp. 1–81, 1995.
- [10] D. A. Papa and I. L. Markov, "Hypergraph partitioning and clustering," in *In Approximation Algorithms and Metaheuristics*, 2007.

TABLE II
IMPLEMENTATION RESULTS OF THE CONTROL UNIT IN CASE OF CFD

		Global control unit	Fully distributed control unit	Optimized control unit
FIFO Area (slice)	Input/output	967	1392	1342
	Inside	0	1932	460
	All	967	3324	1802
Clock frequency (MHz)		<403,551	881,057	512,032

Standard C++ Compiling to GPU with Lambda Functions

Ádám Rák

(Supervisors: Dr. Tamás Roska and Dr. György Cserey)

rakad@digitus.itk.ppke.hu

Abstract—In this paper, a new method of a compiler application to GPU is introduced. In this method, a hybrid executable is generated from the C++ lambda function based code. My compiler plug-in creates GPU accelerated subroutines from code using my library. A C++ run-time library is designed embedding the generated GPU code into the original project.

I. INTRODUCTION

New generation hardware contains more and more processors and the trends show that these numbers will intensely increase in the future. The question is how could we program these systems and may we port earlier codes on them? There is a huge need for this today as well as in the forthcoming period. My new approach of the automation of software development may change the future techniques of computing science.

Exploiting the advantages of the new architectures needs algorithm porting which practically means the complete redesign of the algorithms. New parallel architectures can be reached by “specialized” languages (CUDA, OpenCL, Verilog, VHDL, etc.), for successful implementation, programmers must know the fine details of the architecture. After a twenty years long evolution, efficient compiling for CPU does not need detailed knowledge about the architecture, the compiler can do most of the optimizations. Can we develop as efficient GPU (or other parallel architecture) compilers as the CPU ones? Will it be a two decade long development period again or can we make it in less time?

The specification of a problem describes a relationship from the input to the output. The most explicit and precise specification can be a working platform independent reference implementation which actually transforms the input from the output. Consequently, we can see the (mostly) platform independent implementation, as a specification of the problem.

Parallelization must preserve the behavior in the aspect of specification to give the equivalent results, and should modify the behavior concerning the method of the implementation. Automated hardware utilization has to separate the source code (specification) and optimization techniques on parallel architectures.

There are different trends and technical standards emerging. Without the claim of completeness, the most significant contributions are the following: OpenMP [1] - supports multi-platform shared-memory parallel programming in C/C++ and FORTRAN, practically it uses pragmas for existing codes. OpenCL [2] - is an open, standard C-language extension for the parallel programming of heterogeneous systems, also handling memory hierarchy. Threading Building Blocks of Intel [3] - is a useful optimized block library for shared memory CPUs, which does not support automation. One of

the automation supported solution providers is the PGI Accelerator Compiler [4] of The Portland Group Inc. but it does not support C++. There are problem-software or language specific implementations on many-core architectures, one of them is a GPU boosted software platform under Matlab, called AccelerEyes’ Jacket [5]. Overlooking the growing area, there are successful partially solutions, but there is no universal product and still there are a lot of open problems.

My aim is machine learning boosted OpenCL parallelization of any standard C++ source code by separating programming and parallelization meta-programming. This presentation shows that the basic technological problems (OpenCL source code generation, host code generation and insertion) are manageable: a C++ library is introduced, which can be compiled with every C++0x standard [6] compatible compiler, and produces CPU code. My compiler plug-in and C++ library creates GPU accelerated executables. This approach is methodically one step after the Intel Thread Building Blocks, because the parallelization schemes and memory access patterns are still fixed and provided by my library, but the building blocks themselves become completely user defined in the form of lambda functions.

This paper is organized as follows. After the introduction, in section 2 a general overview of the architecture of the new generation GPUs is given. The lambda functions in the new standard C++ are depicted in section 3. In section 4 I introduce the Minotaurus project which is a gcc based C++ compiler plug-in. Results and working demonstrations are presented in section 5.

This work was presented on the CNNA conference [7] on a demo session, and will be presented on NOLTA [8] too.

II. GPU ARCHITECTURE

Complex real-time 3D rendering needs considerable computing power, orders of magnitude greater than what one CPU can provide. But fortunately the algorithms are all data-parallel, which means that the same code must be executed on all the threads, just the processed data is different. These requirements gave rise to the massively SIMD parallel GPU architectures nowadays. Most of the parts of a normal CPU are sacrificed to place the maximum amount of processing units on the chip. In most cases one core is completely reduced to a simple 32bit FPU / ALU pair, and many cores use the same execution control units on the chip. The pipelines are generally very deep, further allowing more optimization of the architecture. While CPUs need serious trickery, both in hardware (branch prediction, instruction reordering) and sometimes in software too (compilers) to deal with deep

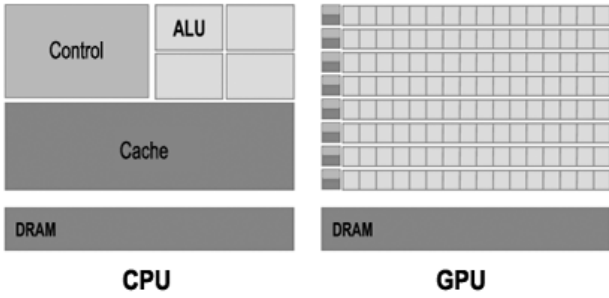


Fig. 1. In the case of a GPU, most of the parts of a normal CPU are sacrificed to place the maximum amount of processing units on the chip. In most cases, one core is completely reduced to a simple 32bit FPU / ALU pair, and many cores use the same execution control units on the chip.

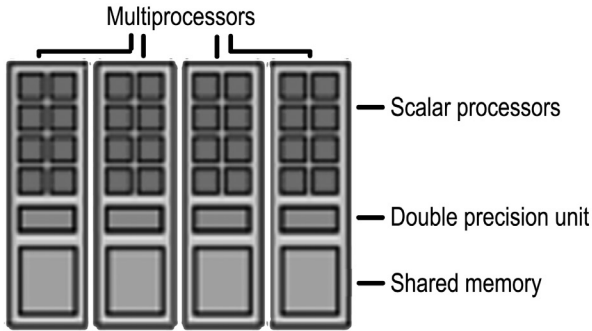


Fig. 2. NVidia GPU architecture usually contains 30 Streaming Multiprocessors (SM), where each SM contains 8 scalar processors, 1 double precision unit, 2 special function units, 16K spared memory and 64K registers.

pipelines, GPUs do not need this because rendering specific algorithms utilize massively huge amount of threads, much more than the number of cores, which makes it very easy to fill the pipelines. This is possible because every thread runs independently on different data, so there is no dependency between them, so on every core, on every pipeline stage a different thread can be executed. The scheduling of threads is done in the hardware to reduce the overhead.

OpenCL provides us an abstraction of the massively parallel hardwares, where both the computing resources (cores) and the memory is hierarchical. This approach was introduced by the hardware manufacturers and it seems that the multicore industry is heading this way. It is suspected [9] that currently this is the optimal trade-off between programmability and performance, where the highest performance is represented by the FPGAs (Field Programmable Gate Array) where everything is parallel, and the maximal programmability by the single core CPU with a single thread running. An important feature is that the memory can be accessed in 1D, 2D or 3D topography, accelerated by the 2D aware hardware caching, and the virtual indexing of the threads can also follow this scheme.

III. LAMBDA FUNCTIONS IN C++

The use of "lambda" originates from functional programming and lambda calculus, where a lambda abstraction defines

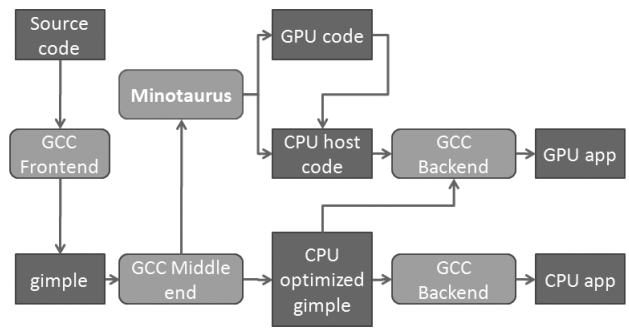


Fig. 3. This flowchart shows the main components of the compiler using my plug-in called Minotaurus. As usually, the gcc compiler has three parts, a front-end, a middle end and the back-end. Minotaurus connects to the middle end using the inner representation of a GCC compiler besides the GPU application output, generating an OpenCL code based GPU-accelerated application output.

an unnamed function. In the new standard of C++ (known also as C++0x) the syntactic element of lambda function is introduced to improve functor usability in templates. The lambda function is an inline expression of a functor object. It is nameless, only a few syntactic units can be given: the captured variables, the parameters, the type of the returning value, and the function body. The created functor will have the captured variables as members, and the constructor will assign the values. The operator() will be created with the parameters and the given function body. The function body is limited to use local variables, the parameters, and the captured variables. There is a convenient way to capture all of the local stack variables in the context as well.

Lambda functions are designed to be used where functors are passed but there is no need to reuse the functor class anywhere else, and building a whole class in order to fulfill syntactic requirements for only one use is circuitous.

IV. MINOTAURUS PROJECT

The flowchart in Figure 3 shows the main components of the compiler using my plug-in called Minotaurus. As usually, the GCC compiler has three parts, a front-end, a middle end and the back-end. Minotaurus connects to the middle end using the inner representation of a GCC compiler besides the GPU application output, generating an OpenCL code based GPU-accelerated application output.

Simplified problem: the programmer specifies the code parts that can be run efficiently on GPU in lambda functions. Minotaurus compiles the lambda function to extract the data and control flow, and synthesize the OpenCL source code which is semantically equivalent to the lambda function. The programmer picks a template function to express the pattern of use (scan primitive for example), and gives the input data. The template function contains the host code which feeds the GPU kernel function. This is where I am standing right now.

A. Standard C++ code input

With Minotaurus, it is possible to compile CPU only executable and CPU-GPU mixed executable as well, using only

```

fill_matrix(in, [tick, shift, speed, x1, y1, x2, y2]
            (int x, int y) -> float {
    float in = tick/speed+shift;
    float jx=sin(in/11.0)*0.4;
    float jy=cos(in/5.0)*0.4;
    float xx=x1+float(xx)/(x2-x1);
    float yy=y1+float(yy)/(y2-y1);

    complex<float> c(jx, jy);
    complex<float> z(xx, yy);
    int k=0;

    do {
        z = z*z + c;
        k++;
    } while (real(z)*real(z)+imag(z)*imag(z) < 4.0
            && k < 1*256.0f);

    return k;
});

```

Fig. 4. This C++ code demonstrates the usage of the lambda function in my system. The fill_matrix() function works as a solution template. The solution template contains hardware specific parallelization schemes and the memory access patterns. The lambda function is defined as a parameter of the function. The implementation of the algorithm is coded by the lambda function. In this given case, it generates a julia-set demonstrating the exploitation of the C++ advantages.

standard C++ language elements in the common source code. This is useful if the debugging process is more complicated with GPU codes, which is usually far more complicated indeed. There are small differences between the resulting executables concerning floating point precision for example, but the theoretic correctness of the implemented method can be checked.

B. Using lambda functions to specify kernels

The fundamental benefit of using lambda functions for compiling to CPU-GPU mixed executable is the clear separation of function and parameter data. For the compilation of the general code, the compiler must explore the full data dependency of the given function to transport the required data to the GPU platform before code execution. This data dependency can be hard to follow because of reading global variables, pointers to globals, etc. Lambda functions are closed in this term, besides local variables, only the captured values and the parameters are accepted inside the function. These variables are given explicitly so any template function can handle the memory transfer to the GPU, so the data dependency of the GPU targeted code can be satisfied.

C. Automatic code generation

The function body of the lambda function may contain elements of C++, such as complex<> type, or references. Minotaurus can convert these elements to a semantically equivalent OpenCL code, using pointers instead of references for example. Lambda functions will be converted to OpenCL functions.

The host code is also generated, the memory transfer can be handled based on the lambda functions' members. The converted lambda functions are called from generated kernel functions based on the parameter set of the lambda function. The host code copies the actual data to the kernel functions,

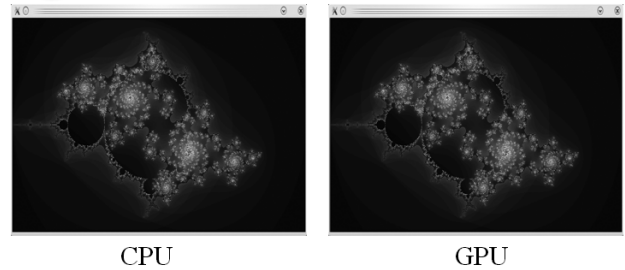


Fig. 5. These images show that the different CPU and GPU accelerated application output generates the same image result, but the GPU version has a significant 25-times performance speed-up.

enqueues the kernel, and reads GPU memory to the returning variables.

D. Extendable technology to other languages

Since Minotaurus works in the inner representation of the compiler, most of the functionality does not depend on the input language.

E. Towards automatic GPU code generation

Some of the hard work is done on Minotaurus right now, but there is plenty of work ahead. Automatic data dependency exploration of any code segment is required for general CPU-GPU hybrid compilation. Functional dependency and guaranteed traces of control are needed in order to select a section of the code which can be compiled to GPU function (entry point of the generated function and the returning points - it is trivial in lambda functions).

F. GPU code generation is functional

We can now compile C/C++ code to OpenCL functions and host code, so we are able to create a hybrid executable from purely C++ code. The performance gain is heavily task dependent, it can be even 80x speed-up. This is notable since the conversion is purely mechanical, no additional tweaking is done with OpenCL local variables, and other sophisticated techniques yet.

V. RESULTS AND DEMONSTRATIONS

Figure 4. demonstrates the usage of the lambda function in my system. The fill_matrix() function works as a solution template. The solution template contains hardware specific parallelization schemes and the memory access patterns. The lambda function is defined as a parameter of the function. The implementation of the algorithm is coded by the lambda function. In this given case, it generates a julia-set demonstrating the exploitation of the C++ advantages (see Figure 5.).

The GPU accelerated version reached up to 25x performance gain on the same source code, utilizing the parallel GP-GPU technology (NVIDIA GTX 280) compared to the OpenMP Intel i7 4 cores implementation. This approach provides C++ support in the kernel code and shows a proof of concept for automatic GPU code (OpenCL) generation.

VI. CONCLUSION

A novel way to program parallel architectures in C++ was demonstrated. With the aid of the OpenCL code generator it is possible to reach any OpenCL capable hardware. This was also a proof of concept implementation paving the way to the automatic parallelization system.

ACKNOWLEDGMENT

The Operational Program for Economic Competitiveness (GVOP KMA), the support of NVIDIA Professor Partnership Program and the Bolyai János Research Scholarship is gratefully acknowledged. The authors are also grateful to Professor Tamás Roska for discussions, his suggestions and his never ending patience.

REFERENCES

- [1] L. Dagum, R. Menon, and S. Inc, "OpenMP: an industry standard API for shared-memory programming," *IEEE Computational Science & Engineering*, vol. 5, no. 1, pp. 46–55, 1998.
- [2] A. Munshi, "The OpenCL specification version 1.0," *Khronos OpenCL Working Group*, 2009.
- [3] J. Reinders, "Intel threading building blocks," 2007.
- [4] M. Wolfe, "Implementing the PGI Accelerator model," in *Proceedings of the 3rd Workshop on General-Purpose Computation on Graphics Processing Units*, pp. 43–50, ACM, 2010.
- [5] AccelerEyes, "Jacket: a GPU engine for MATLAB," 2009.
- [6] P. Becker, "Working draft, standard for programming language C++," *ISO/IEC, Tech. Rep.*, vol. 2798, 2009.
- [7] A. Rak, G. Feldhoffer, G. Soós, and G. Cserey, "CPU-GPU Hybrid Compiling for General Purpose: Case Studies," in *Proceedings of 12th IEEE CNNA - International Workshop on Cellular Nanoscale Networks and their Application*, 2010.
- [8] A. Rak, G. Feldhoffer, G. Soós, and G. Cserey, "Standard C++ Compiling to GPU with Lambda Functions," in *Proceedings of 2010 International Symposium on Nonlinear Theory and its Applications (NOLTA 2010)*, (Krakow, Poland), 2010. accepted.
- [9] K. Hawick, A. Leist, and D. Playne, "Mixing Multi-Core CPUs and GPUs for Scientific Simulation Software," tech. rep., Technical Report CSTN-091, Computer Science, Massey University, 2009.

2D and 3D Level-Set Algorithms on GPU

Gábor János Tornai

(Supervisor: Dr. Tamás Roska and Dr. György Cserey)

torgaja@digitus.itk.ppke.hu

Abstract—Locating object boundaries, is a challenging task in many applications such as computer vision, object detection, image segmentation and tracking. In this paper the implementation of 2D and 3D algorithms based on the level sets using the advantages residing in today’s common GPUs is shown. One main goal of this paper is to contribute a development and give one new local-parallel implementation of a fast level set based algorithm via the locally organized processing elements and memory. This algorithm can model and detect any object with arbitrary complex shape and can be applied to situations where no or very few a priori information is available.

Our accelerated implementation can handle more initial curves and surfaces which can fuse or merge according to the requirements. This might be a good base to achieve fast and robust detection, segmentation or tracking in medical or autonomous tasks.

Index Terms—GPU, level sets, object detection

I. INTRODUCTION

During the past five years new many core architectures have emerged and became common. A new phenomenon can be observed both in the industry and the academic mainstream: the problem and challenge of the many core locally organized computing elements. Mapping various problems, algorithms on these architectures are non trivial tasks. Even so the new generation of GPUs provide easier programmability and increased flexibility.

General purpose computing on GPUs is not so young, but the past decade have brought a dramatic change. Vendors have opened the gates for the developers and researchers by providing APIs based on the C programming language (NVIDIA CUDA SDK, ATI Stream SDK) and in 2009 a new free standard the OpenCL was released. This makes the programming and hopefully also the portability easier.

Locating object boundaries in various datasets is still an interesting task in many imaging problems such as segmentation and tracking. The level set method has become a very popular in the past years. The basic idea comes from two mathematician Osher and Sethian [1] – they investigated the flame front propagation. This front propagation can depend on various properties but the introduced model handles topological changes automatically.

To solve the level set problem we must face partial differential equations (PDEs) of the nonlinear type. To overcome this difficulty a lot of proposals have been made. There are works [2], [3] proposing the evaluation of the PDE only near the zero level set. These methods are referred to as narrow band and sparse field technics. In the GPU implementation a fast level set based algorithm [4] was used which approximates

the solution of the PDEs by constructing a special level set function represented by two sets.

Several researchers and projects succeeded to find solution for 2D image processing problems using level set based algorithms implemented on cellular architecture [5], [6]. Nowadays many-processor architectures and the presence of modern medical imaging systems inspires the development of systems and algorithms, which can process 3D volumetric data flows in real time.

This paper is organized as follows. In section II a general overview of the topic is given. Then some details of the used device is unfold. In section III the details of the 2D and 3D level set algorithms on CPU and CNN as well. The mapping and implementation issues are depicted in section IV while results, time measurements and comparisons are presented in section V. Finally, some conclusions are made in section VI.

II. BACKGROUND

A. Numerical methods of nonlinear PDEs

Evaluating a nonlinear PDE requires aware schemes to compute. The first naive scheme called “central difference” fails automatically after a few steps. I’ve investigated four explicit schemes the Lax-Friedrichs (LF), Lax-Wendroff (LW), Beam-Warminig (BW) and the Min-Max (MM). The MM scheme approximates in the first order the exact solution. It preserves the shocks, rarefaction waves and gives only 2-4 point wide numerical diffusion. The LF scheme is a very robust first order scheme but gives a notable numerical diffusion depending on the time (τ) and space step (h) ratio ($\lambda = \frac{\tau}{h}$). This diffusion can corrupts totally the solution if $\lambda < 0.5$. The LW scheme is second order accurate works only when a tiny diffusion term is given to the equations to smooth out relative heavy Gibbs oscillations on sharp corners. BW is second order accurate and can handle only positive directions. Behavior can be seen on Fig. 1.

B. Curve evolution

The original equation of the level set method is quite a simple one:

$$\frac{\partial \phi}{\partial t} + F|\nabla \phi| = 0 \quad (1)$$

In equation (1) $F = F(L, G, I)$ can be any arbitrary function depending on local (curvature or normal direction), global (shape, position) or independent properties (underlying fluid velocity). Typically, this means there are a few new parameters should be tuned properly for the specific application. F is

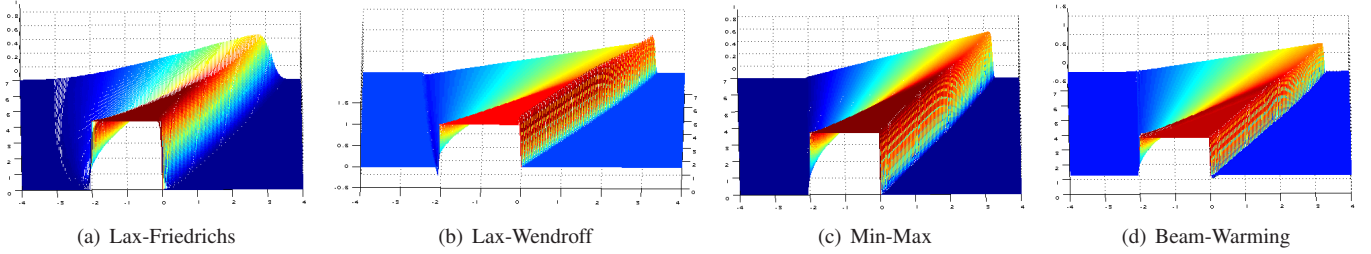


Figure 1. Computed evolution of the burger's equation with different schemes. One can see the heavy diffusion in the LF, and the tiny oscillation on BW and LW scheme. The shock and the rarefaction wave is preserved qualitatively by all schemes.

going to be referred to *speed*. In this case the curve evolution equation is the following:

$$\frac{dC}{dt} = F \cdot \vec{n} \quad (2)$$

where \vec{n} is the normal pointing outward. On a given point the curve motion is controlled by the sign of F . There are a lot of tasks, when F can be divided into two independent terms, F_{ext} and F_{int} . F_{ext} stands for the data dependent external speed and F_{int} for speed depending on local and global properties of ϕ . A popular choice for F_{int} is the curvature of ϕ [7], and if ϕ is chosen as the signed distance function, than F_{int} component is proportional to the Laplacian of ϕ . This term can be approximated by filtering ϕ by a gaussian kernel [8].

C. GPU Architecture

One can easily sum up the characteristics of new generation GPUs. The first main aspect is the hierarchy of the processing cores. By NVIDIA a device [9] contains 8 to 30 *streaming multiprocessors* each consists of 8 or 32 *scalar processors*, an *instruction unit*, and a *transcendental unit*. Every processing element in a stream processor executes the same function called *kernel*. When a kernel is called it is executed in N times in parallel by N different *threads*. Naturally execution of kernels are also hierarchical. Threads are organized into work-groups, and these work-groups are executed in a grid specified by the application.

The second main aspect is the memory hierarchy. Each processor core has its own *registers*. Processor cores from the same stream processor share a fixed amount of *on chip memory* which is accessible from all the threads in a work-group. On chip memory has just a few clock cycle latency and to preserve consistency threads can be synchronized. The third level is the device's DRAM.

Now we describe some details of the NVIDIA GeForce 280 GTX card. This device uses the GT200 architecture released in 2008 June. The GT200 has eight scalar processors in a multiprocessor and 30 multiprocessors. The number of registers per multiprocessor is 16384. The cache working set for constant memory and texture memory per multiprocessor is 8KB (read only). The amount of shared memory per multiprocessor is 16KB, organized into 16 banks. The 1GB amount of global memory is reached through a GDDR3 interface.

III. FAST LEVEL SET ALGORITHM ON REGULAR GRID

As mentioned in the introduction we give a new parallel implementation of a fast level set based algorithm [4] on GPU. The sequential algorithm is based on the key observation that the motion of the curve C can be described by switching points from two sets describing the neighborhood of the zero level set. A detailed description of the algorithm can be found from Y. Shi in [4], [10]. A CNN implementation was published in [11].

Let us assume that ϕ is a real valued function defined over a domain $D \in \mathbb{R}^K$ ($K \geq 2$) which is discretized into a grid. Without the loss of generality we assume that this grid is sampled uniformly. Two "neighboring" sets L_{in} and L_{out} can be defined on this grid as follows:

$$L_{in} = \{\mathbf{x} | \phi(\mathbf{x}) < 0 \text{ and } \exists \mathbf{y} \in N(\mathbf{x}) \text{ that } \phi(\mathbf{y}) > 0\} \quad (3)$$

$$L_{out} = \{\mathbf{x} | \phi(\mathbf{x}) > 0 \text{ and } \exists \mathbf{y} \in N(\mathbf{x}) \text{ that } \phi(\mathbf{y}) < 0\} \quad (4)$$

Where $N(\mathbf{x})$ is defined as:

$$N(\mathbf{x}) = \{\mathbf{y} \in D | \sum_{k=1}^K |y_k - x_k| = 1\} \forall \mathbf{x} \in D \quad (5)$$

The level set function itself is an approximated signed distance function near the zero level set:

$$\phi(x) = \begin{cases} -3, & \text{if } \mathbf{x} \text{ is inside } C, \mathbf{x} \notin L_{in} \text{ inner points} \\ -1, & \text{if } \mathbf{x} \in L_{in} \\ 1, & \text{if } \mathbf{x} \in L_{out} \\ 3, & \text{if } \mathbf{x} \text{ is outside } C, \mathbf{x} \notin L_{out} \text{ outer points} \end{cases} \quad (6)$$

A. CPU Implementation

By switching elements from L_{in} to L_{out} and vice versa: motion of C can be obtained. Of course the level set function should be updated step by step. This can be seen on Figure 2. For efficient implementation this two sets are represented as linked lists. The algorithm itself consists of 4 steps. Interested readers can find further information in [4] or in [10].

The complexity of this implementation is $O(n)$ because there is a fix number of scanning of the sets L_{in} and L_{out} . Where n is the size of the linked lists in other words the number of pixels, voxels next to the zero level set. This set of pixels will be denoted as active fronts. It is essential to

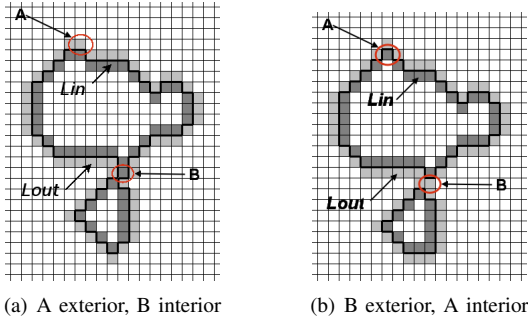


Figure 2. Curve C (solid black line) is implicitly represented by L_{in} , and L_{out} . Motion of C is obtained by switching elements from the two sets. A topological change (a split) can be seen in the two image.

keep the list size under a not so well defined boundary. Under this boundary the list size is linear with respect the image dimension minus one. Exceeding this boundary means that n will become proportional to the number of pixels or voxels in the image. At the same time if the active fronts on the initializing ϕ are very “far” from the object to be found then the number of iterations to converge to the real boundary is increasing according to the Hausdorff distance of the active front and the object to be detected.

One more important notice to the CPU implementation. This implementation does not requires ordered datasets but the data inside the lists can be totally random, unorganized.

B. CNN Implementation

The CNN implementation works with global operators. It has 5 phases:

- 1) initialization
- 2) preprocessing
- 3) accretion phase
 - evolution
 - update outer points
 - eliminate redundant parts from L_{in}
 - update inner points
- 4) diminution phase
 - evolution
 - update outer points
 - eliminate redundant parts from L_{out}
 - update outer points
- 5) check stopping condition if false go back to step 3.

This implementation uses only feed-forward (B template) operators (threshold, logic, arithmetic and binary morphology), the total time required for the operations in step 3, 4, and 5 are less then 70τ . The only exception is the smoothing. It is implemented in the preprocessing phase because a simple diffusion operator for a specific time is equivalent to Gauss filtering the image with a standard deviation proportional to the running time of the diffusion template.

There is a good chance to give a feed-back (A template) operator based solution for this algorithm considering the wave nature of the evolution of the level set’s active front.

IV. ADAPTATION ON GPU

As described in subsection III/A, the CPU implementation uses linked lists for L_{in} and L_{out} to implicitly represent the curve C and arrays to represent ϕ and F_{ext} . On the GPU F_{ext} , force derived from the image, or the image itself does not change in runtime, so it can be represented by a cached read only image object. As a consequence of the cache mechanism reading from a read-only image object the hardware loads the neighbouring pixels too. This means only one read from the global memory and later reads are served by the cache memory.

There can be two different ways to represent the level set function. The first possibility is to use global linear memory buffer (one separate read or write means 300-400 clock cycle latency). Read the buffer to the shared memory space (with coalesced loads) where all active threads in a multiprocessor can reach it within 4 clock cycle. Second possibility is to create the level set function ϕ on the GPU, write it in the shared memory. In this case one global memory transaction is spared. Time measurements are done according to the second described possibility. It was mentioned in subsection II/B that computing units and memory on a device has more levels of organization indicating the cellularity inside the GPU. A smaller collection of threads executing the same kernel invoked on a device are called work-group. A work-group itself can be organized into one, two or three dimensions (this data is going to be referred as local size and local id), shares a fixed amount on chip memory. Our implementation assigns an image segment (4×4 pixel) to a single thread so the scope of a specific thread is restricted on this tiny part of the image and the level set function. Every thread in a work-group works on this small part of ϕ in the shared memory space. The local sizes were set to 16 in one dimension and 4 to the other. These sizes were chosen according to the half warp size of the card. This topographic alignment of the threads and the image implies that there is no need to separately represent the two sets L_{in} and L_{out} . They are represented in natural way on the level set function. The reason for this alignment resides in the GPU’s different memory access latency pattern. It was mentioned in subsection III/A that initializing the level set function with too many small active fronts will cause the algorithm to visit the majority of the pixels in the image which is *not desired* when working on CPU. On GPU this is totally different! By creating huge number of small initializing active fronts the required number of iteration is decreasing and the computing width of the GPU is utilized. Operation of the implemented 2D OpenCL kernel can be divided into 3 parts.

- 1) initialize variables and create/load ϕ in shared memory
- 2) evolve ϕ
 - sweep through the image segment aligned to the thread
 - if $\phi(x) \in L_{in}$ and $F(x) < 0$ then change the value of $\phi(x)$ to L_{out} and add neighbors to L_{in}
 - if $\phi(x) \in L_{out}$ and $F(x) > 0$ then change the value of $\phi(x)$ to L_{in} and add neighbors to L_{in}

Table I

TIME MEASUREMENTS OF THE IMPLEMENTED ALGORITHM ON INTEL CORE2 DUO 3.1 GHZ AND NVIDIA GTX 280 GPU. PRESENTED RESULTS ARE THE MEAN VALUE AND STANDARD DEVIATION OF 20 RUNS.

Data size	Kernel start	Iter. time	Copy	CPU
64×64	44±0.5 us	68±0.4 us	29±1.3 us	0.74 ms
128×128	44±0.5 us	68±0.3 us	46±1.7 us	2.72 ms
256×256	47±0.5 us	79±4.6 us	115±3 us	10.85 ms
512×512	86±1.9 us	170±22 us	390±13 us	69.84 ms
1024×1024	156±2.1 us	592±4.4 us	1553±167 us	626.05 ms

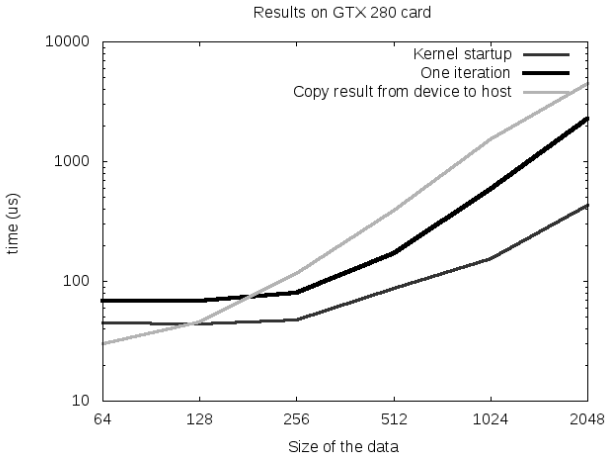


Figure 3. Time measurements on GTX 280 GPU. It can be seen that the required time to copy the result back to the host follows more or less a linear function, while the kernel startup time and the time of one iteration follows the linear after the 256×256 data size. Probably the reason is that the computing width of the GPU was not utilized totally.

- if $\phi(x) \in L_{out}$ and all neighbors of x in ϕ or $F < 0$ set $\phi(x)$ to outer point
- if $\phi(x) \in L_{in}$ and all neighbors of x in ϕ or $F > 0$ set $\phi(x)$ to inner point

3) write out ϕ to a global buffer object.

In this section we described the details of the implementation of the 2D algorithm. With some minor changes in local and global works sizes, using 3D textures, proper shared memory array layout for the level set function, sufficient alignment between a thread and a tiny image size the algorithm can be converted to work on 3D datasets.

V. EXPERIMENTS AND ANALYSIS

All CPU code were implemented in C++ language compiled with gcc 4.4 using standard template library built-in datatypes. All running time measurements were done on Intel Core2 Duo @3.1GHz CPU with 8GB amount main system memory, running GNU/Debian linux, NVIDIA GTX 280 GPU with driver version 190.29. Results are shown in Table I the same data visualized in Figure 3. All GPU host code were implemented in C++ language using OpenCL API calls. All GPU code were implemented in OpenCL. All GPU related measurements were done by getting the GPU's own time

counter through the OpenCL API. The resolution of the device clock on GTX 280 card is 1 us. It can be seen that the copy from the device memory back to the host fits nicely to a linear function, while the kernel startup time containing steps 1 and 3 and the required time for one iteration follows the linear function after data size of 256. This is because the computing width of the device is not utilized. In the case of a 64×64 image the number of work groups started to cover the image is 1×4, similarly in the case of a 128×128 image the number of work-groups required are 2×8. The reason: both numbers are smaller then the number of multiprocessors in the GPU.

VI. CONCLUSION

According to Ryoo at al. [12] an application accelerated by GPU compared to the CPU implementation can achieve 1.16 to 431 speedup. Presented implementation become much faster but the exact rate is hardly measurable. This is because the basic conceptual differences of the initializing level set functions. Finding optimal initializing level set functions for GPUs are hard to found in addition it could vary from application to application.

ACKNOWLEDGEMENT

The Operational Program for Economic Competitiveness (GVOP KMA) and the Office of Naval Research (ONR) is gratefully acknowledged. The author is also grateful to *Barna Garay* for his discussions, suggestions.

REFERENCES

- [1] S. Osher, J. Sethian, and L. R. Center, *Fronts Propagating with Curvature Dependent Speed: Algorithms Based on Hamilton-Jacobi Formulations*. National Aeronautics and Space Administration, 1987.
- [2] D. Adalsteinsson and J. Sethian, "A fast level set method for propagating interfaces," *Journal of Computational Physics*, vol. 118, no. 2, pp. 269–277, 1995.
- [3] R. Whitaker, "A level-set approach to 3D reconstruction from range data," *International Journal of Computer Vision*, vol. 29, no. 3, pp. 203–232, 1998.
- [4] Y. Shi and W. Karl, "A fast level set method without solving PDEs," in *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing, (ICASSP 2005)*, vol. 2. Citeseer, 2005.
- [5] Gy. Cserey, Cs. Rekeczky, and P. Földesy, "PDE based histogram modification with embedded morphological processing of the level-sets," *Journal of Circuits, System and Computers*, vol. 12, no. 4, pp. 519–538, 2003.
- [6] D. Vilarino and C. Rekeczky, "Pixel-level snakes on the CNNUM: algorithm design, on-chip implementation and applications," *Int. Journal of Circuit Theory and Applications*, vol. 33, no. 1, pp. 17–51, 2005.
- [7] L. Vese and T. Chan, "A multiphase level set framework for image segmentation using the Mumford and Shah model," *International Journal of Computer Vision*, vol. 50, no. 3, pp. 271–293, 2002.
- [8] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 7, pp. 629–639, 1990.
- [9] "NVIDIA CUDA," <http://developer.nvidia.com/object/cuda.html>.
- [10] Y. Shi, "Object based dynamic imaging with level set methods," PhD, Boston Univ. College of Eng., 2005.
- [11] G. J. Tornai, G. Cserey, and A. Rák, "Spatial-temporal level set algorithms on CNN-UM," in *International Symposium on Nonlinear Theory and its Application, (NOLTA 2008)*, 2008, pp. 696–699.
- [12] S. Ryoo, C. I. Rodrigues, S. S. Baghsorkhi, S. S. Stone, D. B. Kirk, and W.-m. W. Hwu, "Optimization principles and application performance evaluation of a multithreaded GPU using CUDA," in *PPoPP '08: Proceedings of the 13th ACM SIGPLAN Symposium on Principles and practice of parallel programming*. New York, NY, USA: ACM, 2008, pp. 73–82.

Collision avoidance for UAVs using visual detection

Tamás Zsedrovits
(Supervisors: Tamás Roska and Ákos Zarándy)
zseta@digitus.itk.ppke.hu

Abstract—In this report the steps and the results of my work to make a simulation environment and a basic algorithm for the visual detection of approaching objects in air is summarized. This work is a part of the Office of Naval Research (ONR) project entitled “Sense and Avoid Problems of Unmanned Aerial Vehicle”. Measurements with FlightGear flight simulator driven by Simulink[3] are presented. The basics of the flight dynamics co-ordinate systems and the closed loop demo setup are introduced. The approaching airplane is detected from 4-5km by my MATLAB[3] based algorithm in real time simulation @25Hz.

Index Terms—Image processing, Aircraft detection and tracking, UAV, Mobile robot motion-planning

I. INTRODUCTION

In my first year, my technical work was dedicated for contributing to an ONR project titled “Sense and Avoid Problems of Unmanned Aerial Vehicle”. The final goal of the project is to develop a collision avoidance system for Unmanned Aerial Vehicles (UAVs). As the system will be on the board of a small aircraft we have to minimize the energy consumption in order to minimize the payload. The acceptable power consumption is 1-2W and the mass of the control system is maximum 300-500g. The first stage of the collision avoidance is the detection of the approaching object. In this report an algorithm for the detection of distant aircrafts in a real-time simulation environment is presented.

The organization of the report is as follows. In the first Section the planned sense and avoid system is presented. After that the simulation framework and the work with FlightGear flight simulator are shown. Then commonly used co-ordinate systems and the image processing algorithm are introduced. Finally the closed loop demo setup is shown and the conclusions are drawn.

II. THE SIMULATION FRAMEWORK

A. The sense and avoid system [1]

The goal of the project is to develop a complete autonomous “sense and avoid” flight control system for UAVs. In Fig. 1. the diagram of the control system is shown. This is a closed loop flight control system based on visual detection of the approaching object.

In the first part of the system is the visual sensory processing. The input pictures are recorded by the *Camera*. The recorded pictures are transmitted by *Image Acquisition* to *Preprocessing* block by which the pictures are filtered. The next step of the processing is *Detection*. The images are

processed by image processing algorithms to detect the approaching objects. *Data Association & Tracking* is responsible for the combination of the orientation and angle of attack data of the approaching object calculated by *Detection* and the own position and inertial data measured by onboard *INS/GPS* (Inertial Navigation System/Global Positioning System).

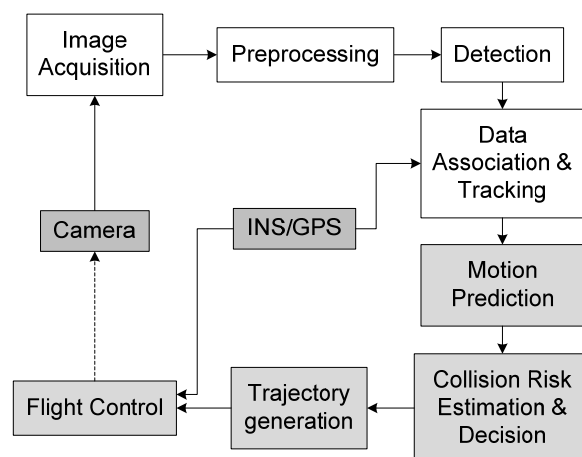


Fig. 1. Diagram of the control system

The second part is the flight control. According to the combined data the relative motion of the approaching object is predicted by *Motion Prediction*. If a risky situation is identified by *Collision Risk Estimation & Decision* a modified trajectory is generated by *Trajectory generation* and the avoiding maneuver is controlled by *Flight Control*.

B. Scene generation

Before the practical flying we have to prove the operability of our system. According to the real system we developed a simulation environment. The *Camera* and the *Image Acquisition* in Fig. 1. are the inputs of the motion prediction and are substituted by FlightGear flight simulator.[4] The FlightGear cooperative flight simulator is an open-source, multi-platform program. It can fetch real weather conditions, it contains more than 100 3D aircraft models and it contains the real geographical data of the half globe.

We used this program to visualize our aircrafts and the environment. We used an external mathematical model implemented by Simulink developed by University of Minnesota. There is a standard communication interface called Flight Dynamic Model (FDM) package by which FlightGear can be driven by Simulink. Additionally, in this way the waypoints navigation can be also implemented by

Simulink. Thus Simulink contains the flight model of the aircrafts and the waypoints.

This part communicates with two FlightGear cooperative instances over Ethernet with FDM packages (shown in Fig. 2. on the right hand side). These packages contain the important data of the aircrafts: state information and actuator deflections. In Fig. 2. on the left hand side the diagram of the communication is shown. Basically in a given time instance the Simulink model provides the data of the aircrafts and FlightGear renders a picture according to the data. The two aircrafts are handled by two FlightGears. The two FlightGears communicate with each other via Ethernet with User Datagram Protocol (UDP) packages on dedicated ports.

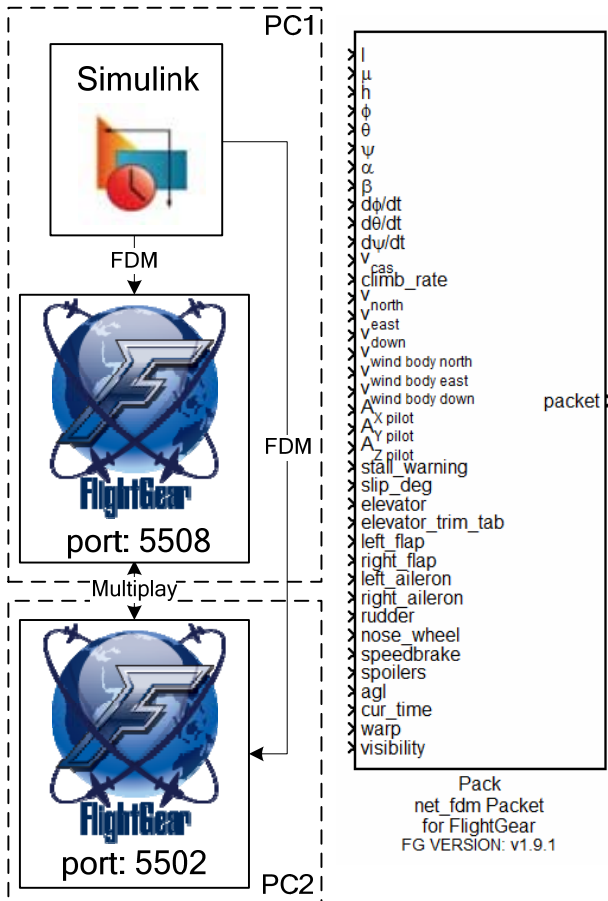


Fig. 2. Communication between Simulink and cooperative FlightGear instances (left); the Simulink FDM block (right)

By default FlightGear doesn't support the saving of rendered pictures in real time (@25Hz). But FlightGear driven by FDM renders the new picture only when it gets a new FDM package from Simulink. Between two incoming FDM packages the picture shown by FlightGear is frozen. If the image acquisition is done after the FDM package processor, in every time instance a newly rendered picture defined by the last processed FDM package is saved.

For motion prediction it is necessary to know the timestamp for each position to synchronize the time. Since FlightGear is an open source project with more than 100 developers, there are unimplemented parts. One of these is the time management via FDM; therefore this channel is used to have the timestamp. OpenGL functions are used to bring the pictures from the graphics card color buffer to the

hard disk (HDD) of the personal computer (PC). This way raw data from the graphics card to PC's hard disk in .ppm (portable pixel map) format are saved and each picture is labeled with its timestamp. The FlightGear source was modified and all the parts of the program were rebuilt.

C. Calibration

For the sake of calculating precise input data for the control algorithm we needed to calibrate the FlightGear program to see how it renders the pictures. First the Field of View (FoV) and the aspect ratio settings were measured. For the measurements Cessna 172P aircraft model was used because this is a very popular light weight airplane. UAVs share airspace with this type of aircrafts and most of them have no radar and use visual sensing for collision avoidance.

The wingspan of Cessna 172P is 11m. The FoV of the rendered image from the following model is calculated (Fig. 3):

$$FoV = \frac{\arctg\left(\frac{5.5}{d}\right) \cdot 2}{w_a} \cdot w \quad (1)$$

where:

- FoV is in degree
- d is the distance of the two aircrafts in meters
- w_a is the measured width of the aircraft in pixels
- w is the width of the rendered image in pixels

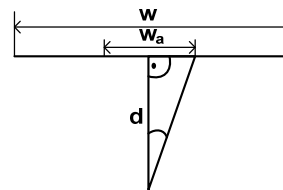


Fig. 3. Geometry of calculation

From the measurements it turned out, that two regions can be defined from rendering point of view: a far region ($d > 20m$), where this model can be used and a close region ($d < 20m$), where distortions of this model are observed. But in our case the far region is interesting because we are not dealing with the emergency situation yet. We have to detect the other aircraft in far enough to do the avoiding maneuver.

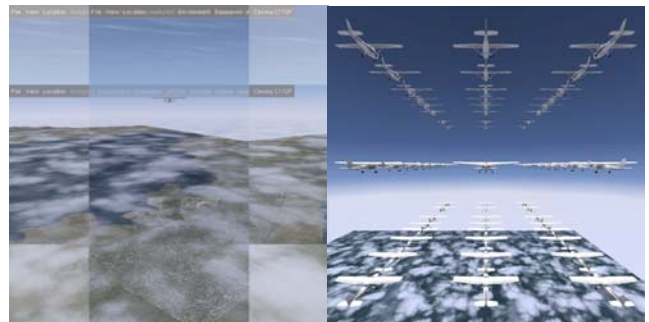


Fig. 4. Image crop in FlightGear, demonstrated with 3 overlapped images on the left hand side; First image was taken in 1000x1000, the second in 1000x500 and the third in 500x1000 resolution. The aspect ratio and FoV were the same in all cases. FlightGear cropped the center region of the 1:1 image in the second and in the third case. On the right hand side: Linear perspective used by FlightGear. Overlapped images

It also turned out that the FlightGear does not take care about the aspect ratio parameter. If geometry is not 1:1, the FoV is set to the bigger size and the image is cropped by FlightGear (Fig. 4). According to the measurements that are not detailed here, it can be asserted that the geometry used by FlightGear is linear perspective. (Fig. 4.).

D. Co-ordinate systems [2]

We use three co-ordinate systems: NED, Body and Camera. Fig. 5 shows an example when we are flying to west and we detect an airplane on the image plane. In the upper left corner the NED co-ordinate system is shown, P denotes the calculated point on the image plane. The 3D position of the detected aircraft is on the orange broken line.

- NED - Fixed in one (latitude,longitude) point. We assume flat Earth and the flying distance is short.
 - X axis - positive in the direction of north
 - Y axis - positive in the direction of east (perpendicular to X axis)
 - Z axis - positive towards the center of Earth (perpendicular to X-Y plane)
- Body - based about aircraft Centre of Gravity
 - X axis - positive forward, through nose of aircraft
 - Y axis - positive to right of X axis (through right wing), perpendicular to X axis
 - Z axis - positive downwards, perpendicular to X-Y plane
- Camera - The camera is fixed to the nose of the plane, but we assume that it doesn't rotate with the aircraft's body according to the NED
 - X axis - positive in the direction of west in NED
 - Y axis - positive to right of X axis, perpendicular to X axis
 - Z axis - positive downwards, perpendicular to X-Y plane

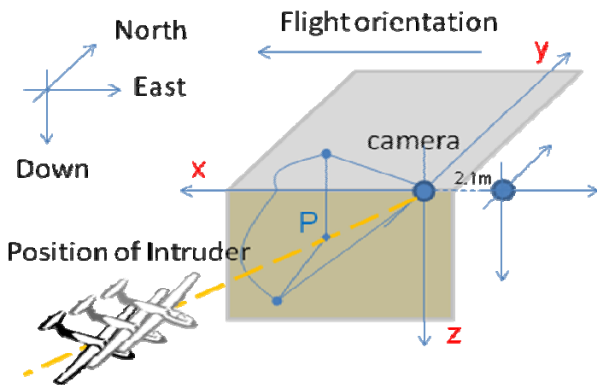


Fig. 5. Co-ordinate systems and diagram of the detection; NED co-ordinate system is shown in upper left corner. In the middle right the aircraft's centre of mass is shown with the inertial co-ordinate system. The camera is fixed 2.1m from the centre of mass in the forward direction. The measured point in camera's image plane is represented by P. The calculated 3D position of the detected aircraft is on the orange broken line.

To have right co-ordinates lever arm compensation is required. In the case of Cessna 172P the distance between the nose and the centre of mass is 2.1m.

E. Image processing algorithm

After the calibration out a basic image processing algorithm was developed. The conditions are the simplest.

The airplanes are at 11km high, there are no clouds in the background, the light conditions are good, the camera is fixed to the NED co-ordinate system and our plain doesn't rotate.

Three different situations were studied: the other plane comes over against us from the front or from left in 90° relative angle or with an arbitrary angle. In every time instance three unknowns has to be evaluated: the other plane's direction by the position on the image plane with azimuth and elevation and the size on the image plane. Later on the wing level information has also been evaluated. From that we can draw conclusions about the orbit of the aircraft.

On Fig. 6. the flowchart of the image processing algorithm is shown. The input images of the algorithm are at least 1 megapixel. As shown on Fig. 6. the first step is a space variant adaptive threshold[5] to filter out the slow transitions on a cut or a non-cut image according to that the current picture is the first or not. To reduce the input image size and speed up the computation, a window containing the intruder airplane according to the previous results is cut. In this way the information calculated by the previous step to the location of the other aircraft on the image plane is used. The adaptive threshold results a binary image containing some of the points of the aircraft. On this binary image a centroid calculation [6] is applied. The centre of Region of Interest (ROI) is determined by the centroid co-ordinates. The size of the ROI is determined by the previously calculated wingsize plus 20 pixels in each direction. In that way two images are cut: one from the original picture (colored ROI image) and one from the result of the adaptive threshold (binary ROI image).

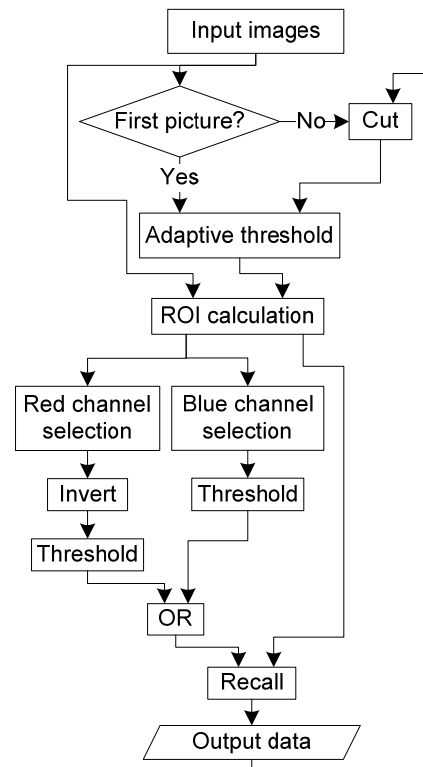


Fig. 6. Diagram of the image processing algorithm

The aircraft is composed by darker and brighter pixels than the original picture's intensity mean value. On the colored ROI image two thresholds are ran. The first one is

calculated on the inverse picture of the grayscale image created from the colored ROI image's red channel. With this threshold the brighter pixels than the original picture's intensity mean value are found. The result is a binary image with the brighter pixels. The other threshold is calculated on the blue channel of the colored ROI image and with it the darker pixels are found. I used the blue channel because the sky is blue and the difference between the background and foreground pictures is the biggest in this channel. The result is a binary image with the darker pixels. A logical OR is applied for the two threshold images. The result is a binary picture with the found pixels of the aircraft and with some other pixels. In some cases the parts of the airplane are not connected in this picture. A closing [7] is applied to connect the components. From the binary ROI picture we have an approximation for the aircraft and from the previously calculated picture we have the pixels of the whole airplane with some noise. A recall [8] is applied according to the binary ROI for the double threshold image.

A small picture with the shape of the airplane is obtained in this way. On this picture the centroid and the size of the aircraft in pixels is determined. Based on the measurements with this algorithm the position of the incoming aircraft can be determined with an error of few meters from 1km. A MATLAB application to demonstrate the image processing algorithm was made. In the application pictures are loaded from the hard disk drive, and are processed with MATLAB based image processing code and are shown on a GUI. The results are saved in a *mat* file and can be visualize in MATLAB.

F. Demo setup

In this month the FlightGear program was adapted to get the rendered pictures into the main memory of the PC. These pictures are sent to MATLAB Engine. MATLAB Engine is an interface provided by MATLAB to run MATLAB code from C++. The image processing algorithm presented in section E is done by this module.

A desktop PC with Intel Core 2 Quad Q9400 CPU @2.66GHz 4GB RAM Nvidia GeForce 210 graphics card and Windows 7 Professional operating system is used to run the modified FlightGear instance (*PC2* in Fig. 1.). A laptop PC with Intel Core 2 Duo T9300 CPU @2.5GHz 2GB RAM Intel GMA X3100 integrated graphics and Windows 7 Professional operating system is used to run Simulink and the other cooperative FlightGear instance (*PC1* in Fig. 1.). In the case of the demo setup the *Motion Prediction* (shown in Fig. 1.) is done by a Kalman filter realized in Spartan 3 FPGA (Field-programmable gate array).

In Fig. 7. the diagram of the demo setup is shown. The flight control is running on hardware in the loop system, shown at the upper left corner. This block communicates with the image processing PC via Ethernet. On the image processing PC a modified FlightGear is running, which is sending the rendered pictures to the MATLAB Engine and the results to the FPGA via USB. The FPGA realizes a Kalman filter and calculates the *Motion Prediction* data required by the control block. These data are forwarded to the control block by the image processing pc via Ethernet. Our aim is to implement image processing algorithm on

FPGA in a later stage of the project, because the requirements with respect to the power consumption and the system's mass and volume.

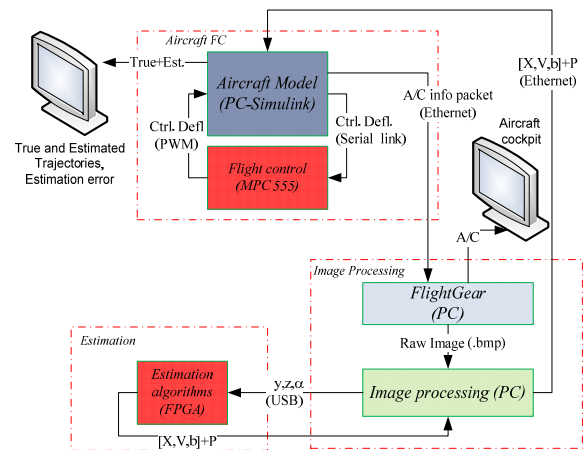


Fig. 7. Diagram of the Demo setup

III. RESULTS

The basic problem of the collision avoidance of UAVs was mentioned. The planned "sense and avoid" system, the work with FlightGear simulator, a basic algorithm for the detection and the diagram of the demo setup were presented. A journal paper is in preparation according to these results.

IV. ACKNOWLEDGMENT

The author would like to thank Ákos Zarándy, Bálint Vanek, Tamás Péni, László Füredi, Csaba Nemes, Zoltán Nagy, Tamás Fülöp, Gábor Tornai, Domonkos Gergelyi, Ádám Rák, Gergely Feldhoffer for their help.

V. REFERENCES

- [1] J. Bokor, T. Roska, P. Szolgay, Á. Zarándy, Z. Nagy, B. Vanek, T. Péni, L. Füredi, Cs. Nemes, T. Zsedrovits "Sense and Avoid Problems of Unmanned Aerial Vehicle", whitepaper for ONR meeting 2010.
- [2] R. M. Rogers, *Applied Mathematics in Integrated Navigation Systems, Second Edition*. Reston, Virginia: American Institute of Aeronautics and Astronautics, Inc., 2003, pp. 41–55.
- [3] MATLAB and Simulink are registered trademark of The MathWorks, Inc.
- [4] <http://www.flightgear.org/Docs/getstart/getstart.html>
- [5] William K. Pratt, *Digital Image Processing: PIKS Inside, Third Edition*, Los Altos, California: PixelSoft Inc., 2001, pp. 554.
- [6] William K. Pratt, *Digital Image Processing: PIKS Inside, Third Edition*, Los Altos, California: PixelSoft Inc., 2001, pp. 601-607.
- [7] William K. Pratt, *Digital Image Processing: PIKS Inside, Third Edition*, Los Altos, California: PixelSoft Inc., 2001, pp. 433-435.
- [8] "CNN Software Library, (Templates and Algorithms)", edited by T.Roska, L. Kék, L. Nemes, and Á. Zarándy, Comp. and Auto. Ins. of the Hung. Acad. of Sci. DNS-1-1997, Budapest, 1997.

A Redesigned Emulated Digital CNN Architecture for FPGAs

László Füredi

(Supervisor: Dr. Péter Szolgay)
furla@digitus.itk.ppke.hu

Abstract—Cellular Neural Network (CNN) is a prototype Single Instruction Multiple Data (SIMD) like architecture, where the basic operation of this architecture is the weighted sum calculation. The emulated digital CNN-UM architecture was implemented and tested on different kind of array computers, eg. Cell Broadband Engine (Cell BE)[1], Field-Programmable Gate Arrays (FPGAs)[2], Stream processors[3] and graphics processing units (GPUs)[4] for utilizing the high performance of the digital microprocessors.

The arithmetic unit of the original Falcon architecture was mainly optimized for the special features of the Xilinx Virtex-II architecture. Implementing the same architecture on the new Digital Signal Processor (DSP) optimized FPGAs will be inefficient. In order to achieve the highest possible performance the dedicated elements of the new FPGAs should be fully utilized. Therefore an improved arithmetic unit should be designed. According to the requirements of the new arithmetic unit the input data structure and the data-flow of the processor should be redesigned. Additionally the interconnection of the Falcon processing elements are optimized to utilize the specialized interconnect resources on the FPGA.

Compared to the original Falcon processor with the modified implementation on the new FPGA families the clock frequency can be improved by approximately 20 percent. Additionally the area requirement of the arithmetic unit is significantly reduced by utilizing the special features of the DSP blocks.

Index Terms—Array Computers, FPGA, Emulated Digital CNN-UM, FALCON.

I. INTRODUCTION

In high performance processors the operation delay and the wiring delay is comparable. This effect is explicable with the scaling down of the technology. The increase of clock frequency the signal does not have enough time to reach the destination in one cycle. The adjacent computational elements can communicate faster because in short range the wiring delay is not significant. The effective architecture design's prime aspect is the locality precedence. This precedence is studied in an emulated digital CNN-UM implementation.

A multi-layer CNN array can be used to solve the state equation of complex dynamical systems [5][6]. The CASTLE and Falcon emulated digital CNN chips were designed to reach this goal [7][8][9], where the accuracy, number of templates, template size, cell array size, the number of layers can be configured, the number and arrangement of the processor cores. In the FALCON processor array there are homogenous processor (Figure 1.), each processor works on a narrow slice of the image and each line of processor computes one CNN

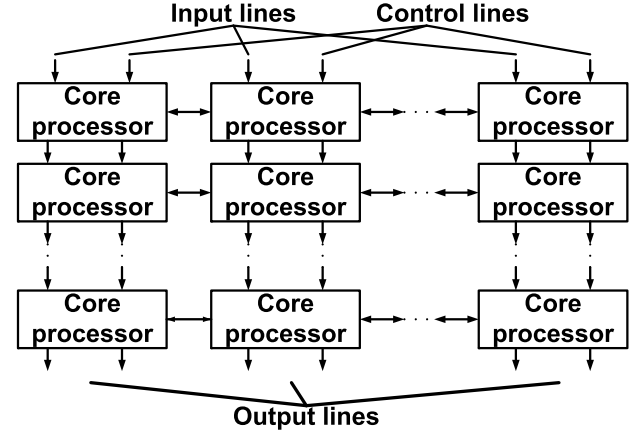


Fig. 1. The FALCON processor array

iteration. The results are shifted down to the next processor row.

This paper describes synthesis and implementation methods used for the modified Falcon processor array on Virtex-5, Virtex-6 and Virtex-7 FPGAs.

The Falcon architecture is designed to solve the full signal range model of the CNN cell [10][11].

$$\begin{aligned} \dot{x}_{i,j}(t) = & \sum_{k=0}^{2 \cdot n} \sum_{l=0}^{2 \cdot n} \mathbf{A}_{k,l} \cdot x_{i+k-n,j+l-n}(t) + \\ & + \sum_{k=0}^{2 \cdot n} \sum_{l=0}^{2 \cdot n} \mathbf{B}_{k,l} \cdot u_{i+k-n,j+l-n}(t) + I_{i,j} \end{aligned} \quad (1)$$

where x , u and I are the state, input and the bias values of the CNN cell, n is the neighborhood size, \mathbf{A} is the feedback, \mathbf{B} is the feed-forward template. The templates are $(2n+1) \times (2n+1)$ sized matrices. The state equation of the CNN array is solved on the Falcon architecture by forward Euler discretization. The h time step value can be inserted into the templates \mathbf{A} and \mathbf{B} , these modified templates are denoted by $\hat{\mathbf{A}}$ and $\hat{\mathbf{B}}$. Usually the input values do not change for several time steps so the state equation (1) can be partitioned into two parts, the feedback (2) and the feed-forward part (3).

$$x_{i,j}(m+1) = \sum_{k=0}^{2 \cdot n} \sum_{l=0}^{2 \cdot n} \hat{A}_{k,l} \cdot x_{i+k-n,j+l-n}(m) + g_{i,j} \quad (2)$$

$$g_{i,j} = \sum_{k=0}^{2 \cdot n} \sum_{l=0}^{2 \cdot n} \hat{B}_{k,l} \cdot u_{i+k-n,j+l-n} + h \cdot I_{ij} \quad (3)$$

The problem to be solved how to map the computational problem defined in (2) and (3) on a virtual array to a given physical FPGA where area/processor (logic slices, DSP slices), on-chip memory (Block Random Access Memory (BRAM)) and off-chip memory bandwidth are limited. Depending on the complexity of the operator a small amount of physical execution units can be implemented $n \ll N \times M$ (in 2D case) or $N \times M \times L$ (in 3D case). The operator can be decomposed into small basic blocks which use either the logic resources (such as adders) or the dedicated resources (embedded multipliers) of the FPGA. The result of this process is a Virtual Cellular Machine optimized for the given application. The optimization can be focused on area, accuracy, speed, dissipated power etc. Main components are on-chip memory and the specialized execution unit.

II. THE RESOURCES ON AN FPGA

The main configurable elements of the new Xilinx Virtex family is the Advanced Silicon Modular Block (ASMBL)[12]. The architecture is column based where each ASMBL column has specific capabilities, such as logic, memory, Input/Output, DSP, hard IP and mixed signal. By using different mix of the ASMBL columns domain specific devices can be manufactured. In the new architecture traditional 4-input Look-up Tables (LUTs) are replaced by 6-input LUTs. Each configurable logic block (CLB) is divided into two slices and every slice contains 4 6-input LUTs, 4 registers, selection circuitry (MUX, etc.) and carry logic.

In the new FPGAs the simple multipliers are replaced by complex DSP blocks called XtremeDSP (DSP48E) slices, it supports over 40 dynamically controlled operating modes including: multiplier, multiplier-accumulator, multiplier-adder/subtractor, three input adder, barrel shifter, wide bus multiplexers, wide counters, and comparators. The embedded registers in the DSP48E and its ability to change its operation on a clock-by-clock basis block save lots fabric Flip-Flops (FFs) and LUTs. A lot of functionality that would typically be taken out of the DSP48E block can be kept inside by using its registers and different modes of operation. The DSP48E enable adder-chain architectures for implementing complex algorithm efficiently.

The heart of the DSP48E is a 25bit by 18bit 2's complements signed multiplier with full precision 43-bit result. It also contains a 48bit Arithmetic Logic Unit (ALU) with optional registered accumulation feedback and support for SIMD operations. Additionally, hard wired 17 bit shift capability simplifies the construction of large multipliers, while optional pipeline registers enable even higher operation frequency.

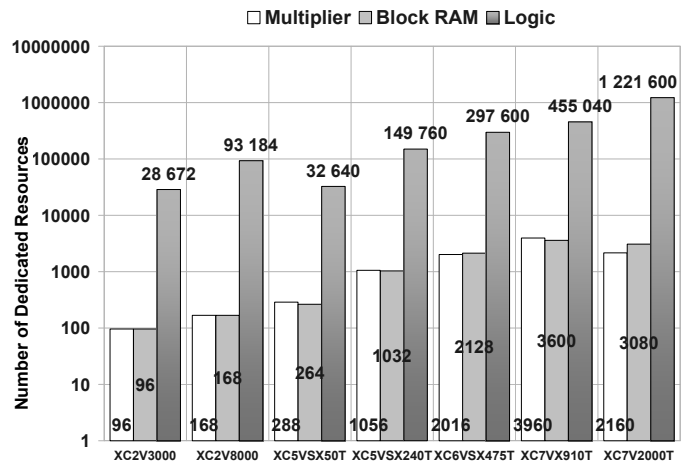


Fig. 2. The Resources of FPGAs

The number of DSP48Es is 1056 in a Virtex-5 SX240T, 2016 in a Virtex-6 SX475T, 3960 in a Virtex-7 VX910T, 2160 in a Virtex-7 V2000T FPGA. The other key configurable elements are the interconnect wires. In the contribution we especially focus on minimization of wire delays.

III. ARCHITECTURAL IMPROVEMENTS

The new FPGA families have much more resources than the Virtex-II FPGA which was used for the first implementation of the Falcon processor. For solving the discretized version of the CNN state equation a large number of multiplication is needed which can easily and efficiently implemented by using the dedicated elements (multipliers or DSPs) of the FPGAs. The available dedicated resources of the different FPGAs can be seen on Figure 2. Scaling up the original Falcon architecture on the new FPGAs in terms of the multipliers shows that on new FPGAs there are not enough configurable logic resources. If 32 original Falcon processor cores are implemented on the Virtex-II 3000 FPGA 94 percent of the configurable logic blocks and all of the multipliers can be utilized.

Examining the available resources on the Virtex-5 SX50T, Virtex-5 SX240T, Virtex-6 SX475T and Virtex-7 VX910T devices 228, 182, 140 and 180 percent of available logic block is required to implement the original Falcon processor when all multipliers are utilized (3 multiplier for one processor core) as shown in Figure 3. This number is different for Virtex-7 V2000T because this type of FPGA is optimized for high number of logic resources. If more multipliers (for example 9) is used for each processor cores, and this multipliers built up from logic resources, in that way the number of processor cores and the operation frequency of the system is reduced dramatically. The question is how to arrange the computation to use all multipliers while not overusing configurable logic blocks to implement the Falcon architecture on new FPGAs. The new modified type of architecture is shown on Figure 4, where the mixer and arithmetic units were changed. With these changes which will be described in the next sections all of the

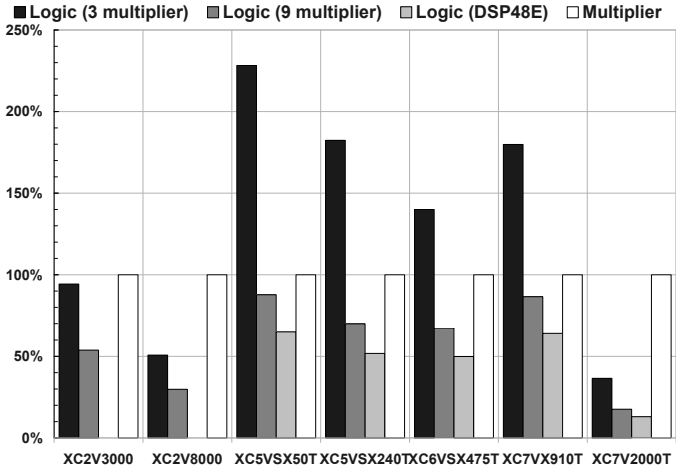


Fig. 3. The logic usage in Falcon

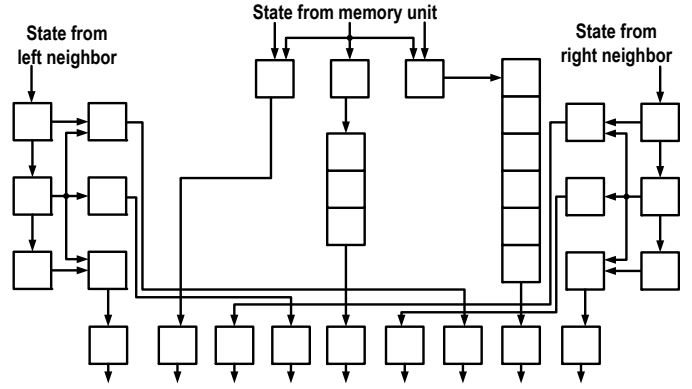


Fig. 5. Structure of the modified mixer unit

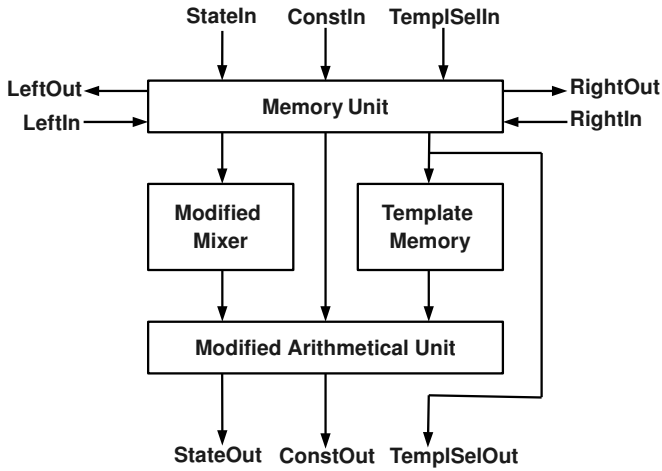


Fig. 4. Structure of one Falcon processor core

built-in DSP48E slices can be used and the configurable logic block requirement of the processor core is also reduced.

A. Modified Mixer Unit

The structure of the mixer unit is shown in Figure 5. This unit contains one block of shift registers to store a window around the currently processed cell and two additional blocks of shift registers which are used to store data from the left and right neighbors of the processor. The registers are connected serially and their outputs are also connected to the S_x inputs of the arithmetic unit. Communication between the neighboring processors is carried out through the left and right inputs without affecting the arithmetic unit. As a result the number of cycles required for the processing is reduced which increases the performance of the architecture and enables 100 percent utilization of the multipliers in the arithmetic unit.

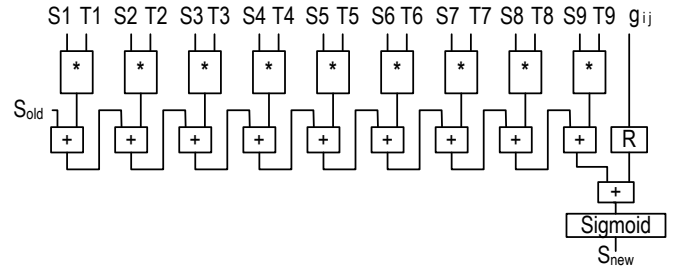


Fig. 6. Structure of the improved arithmetic unit

B. Modified Arithmetic Unit

The S_x inputs of the arithmetic unit are connected to the state value outputs of the mixer unit while the T_x template values are connected to the output of the template memory. The precision of the state values is 25bit while the inputs are 18 bit wide. All of the multipliers and adders are implemented inside the cascaded DSP48E slices. Using this structure the dedicated connections between the DSP48E slices can be utilized. Therefore the operating frequency of the arithmetic unit is the maximum that the DSP slices allow. Depending on the speed grade of the FPGA the operating frequency of the arithmetic unit can reach 550MHz on the Virtex-5, 650MHz on the Virtex-6 and 600MHz on the Virtex-7 FPGAs. Only one external element a register is required to store the feed-forward value of the computation and it comes from the memory unit.

The modified arithmetic unit is shown in Figure 6, and it can be used in pipelined mode.

IV. PERFORMANCE

Performance of the modified Falcon processor is compared to the speed of the software simulation. In the software simulation Intel Core 2 Duo E8400 and IBM CellBE processors with 8 Synergistic Processing Elements (SPEs) was used. To simulate CNN array functions of the Intel Performance

TABLE I
COMPARISON OF DIFFERENT IMPLEMENTATIONS

Implementation type	Implementations						
	Intel Core 2 Duo	Cell Processor 8 SPEs	FPGA				
			XC5VSX50T	XC5VSX240T	XC6VSX475T	XC7VX910T	XC7V2000T
	Software (Intel IPP)[13]	Software (Cell SDK)	FPGA	FPGA	FPGA	FPGA	FPGA
Technology (nm)	45	65	65	65	40	28	28
Clock Frequency (MHz)	3000	3200	550	550	650	600	600
Number of Processing Elements	2 Cores	8 SPE	32 FPE	117 FPE	224 FPE	440 FPE	240 FPE
Million cell iteration/s	400	3627	17600	64350	145600	264000	144000
Speedup	1	9	44	160	364	364	364
Power Dissipation (W)	65	85	~ 16	~ 59	~ 102	N/A	N/A
Area (mm ²)	107	2×253	N/A	N/A	N/A	N/A	N/A

Primitives - Image Processing Library (IPP IPL) was used to help to optimize image- and vector-processing tasks.

Performance of the software simulation depends on the size of the cell array. If the size is larger than 688×688 the performance drops to a lower level, due to the memory bottleneck and L2-cache memory occupancy. Even in a single Falcon processor configuration 38 percent performance improvement can be achieved compared to Intel Core 2 Duo processor.

The easy scalability of the array makes it possible to connect severally modified Falcon processor cores on one FPGA and get even more performance. Using the previously described architecture and utilizing all the 440 modified Falcon processors on the Virtex-7 FPGAs 264 billion cell iteration per second computing performance can be achieved.

The Virtex-7 FPGA based solution is 660 times faster compared to a high performance microprocessor, using all of the modified Falcon processors during the computation. Compared to a high performance Intel Core 2 Duo microprocessor, for a 1024 × 1024 pixel picture instead of 38 template execution per second, 25177 template execution per second can be used with the Virtex-7 Falcon implementation.

The system precision is 25 bit if better precision is needed, it is possible to use the architecture but the implementable processor cores will be reduced to half of the original one.

V. CONCLUSIONS

An improved emulated digital CNN-UM architecture implementation was successful on the prototyping boards, using the Virtex-5 SX50T and Virtex-5 SX240T FPGA from Xilinx Inc. and implementation in simulation using the Virtex-6 SX475T FPGA and hand calculation using the Virtex-7 VX910T and Virtex-7 V2000T FPGAs.

The solution was optimized to the special requirements of the Virtex-5, Virtex-6, and Virtex-7 FPGAs. The main parameters of the architecture is described and compared to the parameters of the software simulation of the CNN full signal range model running on high performance processors such as Intel Core 2 Duo and IBM Cell.

REFERENCES

- [1] Z. Nagy, L. Kék, Z. Kincses, A. Kiss, and P. Szolgay, "Toward exploitation of cell multi-processor array in time-consuming applications by using CNN model," *International Journal of Circuit Theory and Applications*, vol. 36, pp. 605–622, 2008.
- [2] Z. Vörösházi, A. Kiss, Z. Nagy, and P. Szolgay, "Implementation of embedded emulated-digital CNN-UM global analogic programming unit on FPGA and its application," *International Journal of Circuit Theory and Applications*, vol. 36, pp. 589–603, 2008.
- [3] L. Füredi and P. Szolgay, "CNN Model on Stream Processing Platform," *Proceedings of the European Conference on Circuit Theory and Design 2009*, pp. 843 – 846, 2009.
- [4] B. G. Soós, A. Rák, J. Veres, and G. Cserey, "GPU boosted CNN simulator library for graphical flow based programmability," *EURASIP Journal on Advances in Signal Processing*, p. 11, 2008.
- [5] Z. Nagy, Z. Vörösházi, and P. Szolgay, "Emulated Digital CNN-UM Solution of Partial Differential Equations," *International Journal of Circuit Theory and Applications*, vol. 34, pp. 445–470, 2006.
- [6] T. Roska, "An Overview on Emerging Spatial Wave Logic for Spatial-Temporal Events Via Cellular Wave Computers on Flows and Patterns," *International symposium on nonlinear theory and its applications*, pp. 98–100, 2008.
- [7] P. Keresztes, A. Zarándy, T. Roska, P. Szolgay, T. Hidvégi, P. Jónás, and A. Katona, "An Emulated Digital CNN implementation," *International Journal of VLSI Signal Processing*, vol. 23, pp. 291–303, 1999.
- [8] Z. Nagy and P. Szolgay, "Configurable Multi-layer CNN-UM Emulator on FPGA," *IEEE Transaction on Circuit and Systems I: Fundamental Theory and Applications*, vol. 50, pp. 774–778, 2003.
- [9] Z. Kincses, Z. Nagy, , and P. Szolgay, "Implementation of Nonlinear Template Runner Emulated Digital CNN-UM on FPGA," *Cellular Neural Networks and Their Applications, 2006. CNNA '06. 10th International Workshop on 28-30 Aug. 2006*, pp. 1–5, 2006.
- [10] L. O. Chua and L. Yang, "Cellular neural networks: theory," *IEEE Transactions on Circuits and Systems*, vol. 35, no. 10, pp. 1257–1272, 1988.
- [11] S. Espejo and et.al., "A VLSI-Oriented Continuous-Time CNN Model," *International Journal of Circuit Theory and Applications*, vol. 24, pp. 341–356, 1996.
- [12] "Xilinx product homepage," <http://www.xilinx.com>, 2010.
- [13] "Intel integrated performance primitives homepage," <http://software.intel.com/en-us/intel-ipp/>, 2010.

Emulated Digital Cellular Neural Networks for Accelerating CFD Simulations

András Kiss

(Supervisor: Dr. Péter Szolgay)

kissa@digitus.itk.ppke.hu

Abstract—The analog CNN-UM can be used to solve the Navier-Stokes equations quite fast. But using in engineering applications it can not be sufficiently accurate and reliable because noises from the environment, such as power supply noise or temperature fluctuation. With the proper Field Programmable Gate Array (FPGA) we can gain sufficient computation speed with high precision. The dedicated hardware elements of the FPGA can highly accelerate the computations on curved surface. Consequently it can be used in industrial applications where fluid flow simulation around complex shapes is required. In the paper the implementation and optimization of a new Computational Fluid Dynamics (CFD) solver architecture, which can work on Body Fitted Mesh geometry, on FPGA is described. The proposed new architecture is compared to existing solutions in terms of area, speed, accuracy and power dissipation.

I. INTRODUCTION

However, most real life applications of CFD require handling more complex geometries, bounded by curved surfaces. A popular and often an efficient solution to this problem is to perform the computation over non-uniform, logically structured grids for example to use body fitted mesh geometry. Although the standard 2D scheme over Cartesian geometry can be put to work, it is computationally much more demanding, due to the expensive operations related to coordinate transformation.

Array computers can easily solve a couple of numerical spatiotemporal problems such as partial differential equations (PDE). There are a number of different implementations of array processors commercially available from different vendors [2][3][4]. But, reconfigurable devices seems to be the most versatile devices to implement array processors. Flexibility of the FPGA devices enable to use different computing precisions during the solution of PDEs and evaluate different architectures quickly[5].

In the paper we extend the previous work to the solution of 2D Euler equations over structured quadrilaterals defined by a single block of body fitted grid. On the FPGA we can make a more specific structure for the CFD with better performance in terms of the area and dissipation with a variable accuracy considering to use it in real life applications.

II. FLUID FLOWS

To simulate gas or fluid flows over complex obstacles requires the enormous computing power of today's supercomputers. In engineering applications the temporal evolution of non-ideal, compressible fluids is quite often modeled by

the system of Navier-Stokes equations. It is based on the fundamental laws of mass-, momentum- and energy conservation, extended by the dissipative effects of viscosity, diffusion and heat conduction. By neglecting all the above non-ideal processes, and assuming adiabatic variations, we obtain the Euler equations [6], describing the dynamics of dissipation-free, inviscid, compressible fluids. The equations are a coupled set of nonlinear hyperbolic partial differential equations, in conservative form expressed as

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \quad (1)$$

$$\frac{\partial (\rho \mathbf{v})}{\partial t} + \nabla \cdot (\rho \mathbf{v} \mathbf{v} + \hat{I} p) = 0 \quad (2)$$

$$\frac{\partial E}{\partial t} + \nabla \cdot ((E + p) \mathbf{v}) = 0, \quad (3)$$

where t denotes time, ∇ is the Nabla operator, ρ is the density, u, v are the x- and y-components of velocity vector \mathbf{v} , respectively, p is the pressure of the fluid, \hat{I} is the identity matrix, and E is the total energy density defined as

$$E = \frac{p}{\gamma - 1} + \frac{1}{2} \rho \mathbf{v} \cdot \mathbf{v}, \quad (4)$$

where γ is the ratio of specific heats.

III. DISCRETIZATION OF THE GOVERNING EQUATIONS

Since logically structured arrangement of data is fundamental for the efficient operation of the FPGA based implementations, we consider explicit finite volume discretizations of the governing equations over structured grids employing a simple numerical flux function. Indeed, the corresponding rectangular arrangement of information and the choice of multi-level a temporal integration strategy ensure the continuous flow of data through the CNN-UM architecture. In the followings we recall the basic properties of the mesh geometry, and the details of the considered first- and second-order schemes.

A. The geometry of the mesh

The computational domain is composed of $n \times m$ logically structured quadrilaterals, called finite volumes as shown in Figure 1. Indices i and j refer to the volume situated in the i^{th} column and the j^{th} row. The corners of the volumes can be described practically by any functions X and Y of indices (i, j) , provided that degenerated volumes do not appear:

$$x_{i,j} = X(i, j), y_{i,j} = Y(i, j), \quad (5)$$

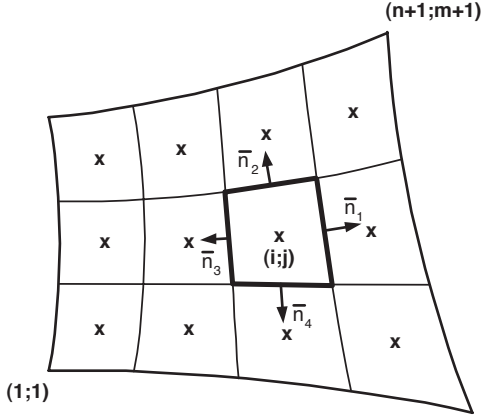


Fig. 1. The computational domain

where $x_{i,j}$ and $y_{i,j}$ stand for the x - and the y - component of corner point (i, j) , respectively, $i \in [1, n + 1]$ and $j \in [1, m + 1]$. In real life applications X and Y follow from the functions describing the boundaries of the computational domain. Consider a general finite volume with indices (i, j) , presented in Figure 1. Its volume is labeled by $V_{i,j}$, while \mathbf{n}_f represents the outward pointing normal vector of face f scaled by the length of the face.

B. The First-order Scheme

The simplest algorithm we consider is first-order both in space and time [1]. The application of the finite volume discretization method leads to the following semi-discrete form of governing equations (1-3)

$$\frac{dU_{i,j}}{dt} = -\frac{1}{V_{i,j}} \sum_f \mathbf{F}_f \cdot \mathbf{n}_f, \quad (6)$$

where the summation is meant for all the four faces of cell (i, j) , \mathbf{F}_f is the flux tensor evaluated at face f , and \mathbf{n}_f is the outward pointing normal vector of face f scaled by the length of the face. Let us consider face f in a coordinate frame attached to the face, such that its x -axes is normal to f . Face f separates cell L (left) and cell R (right). In this case the $\mathbf{F}_f \cdot \mathbf{n}_f$ scalar product equals to the x -component of $\mathbf{F}(F_x)$ multiplied by the area of the face. In order to stabilize the solution procedure, artificial dissipation has to be introduced into the scheme. Following the standard procedure, this is achieved by replacing the physical flux tensor by the numerical flux function F^N containing the dissipative stabilization term. A finite volume scheme is characterized by the evaluation of F^N , which is the function of both U_L and U_R . In the paper we apply the simple and robust Lax-Friedrichs numerical flux function defined as

$$F^N = \frac{F_L + F_R}{2} - (|\bar{u}| + \bar{c}) \frac{U_R - U_L}{2} \quad (7)$$

and bar labels speeds computed at the following averaged state

$$\bar{U} = \frac{U_L + U_R}{2}. \quad (8)$$

In the last equation c is the local speed of sound, $F_L = F_x(U_L)$, $F_R = F_x(U_R)$.

The last step concludes the spatial discretization. Finally, the temporal derivative is discretized by the first-order forward Euler method:

$$\frac{dU_{i,j}}{dt} = \frac{U_{i,j}^{n+1} - U_{i,j}^n}{\Delta t}, \quad (9)$$

where $U_{i,j}^n$ is the known value of the state vector at time level n , $U_{i,j}^{n+1}$ is the unknown value of the state vector at time level $n + 1$, and Δt is the time step. By working out the algebra described so far, leads to the following discrete form of the governing equations:

$$\begin{aligned} \rho_C^{n+1} &= \rho_C^n - \\ & - \frac{\Delta t}{\Delta x} \left(\left(\frac{\rho u_C^n + \rho u_E^n}{2} - (|\bar{u}| + \bar{c}) \frac{\rho_E^n - \rho_C^n}{2} \right) \right. \\ & - \left(\frac{\rho u_W^n + \rho u_C^n}{2} - (|\bar{u}| + \bar{c}) \frac{\rho_C^n - \rho_W^n}{2} \right) \\ & + \left(\frac{\rho v_C^n + \rho v_N^n}{2} - (|\bar{u}| + \bar{c}) \frac{\rho_N^n - \rho_C^n}{2} \right) \\ & \left. - \left(\frac{\rho v_S^n + \rho v_C^n}{2} - (|\bar{u}| + \bar{c}) \frac{\rho_C^n - \rho_S^n}{2} \right) \right) \end{aligned} \quad (10a)$$

$$\begin{aligned} \rho u_C^{n+1} &= \rho u_C^n - \\ & - \frac{\Delta t}{\Delta x} \left(\left(\frac{(\rho u^2 + p)_C^n + (\rho u^2 + p)_E^n}{2} - (|\bar{u}| + \bar{c}) \frac{\rho u_E^n - \rho u_C^n}{2} \right) \right. \\ & - \left(\frac{(\rho u^2 + p)_W^n + (\rho u^2 + p)_C^n}{2} - (|\bar{u}| + \bar{c}) \frac{\rho u_C^n - \rho u_W^n}{2} \right) \\ & + \left(\frac{\rho u v_C^n + \rho u v_N^n}{2} - (|\bar{u}| + \bar{c}) \frac{\rho u_N^n - \rho u_C^n}{2} \right) \\ & \left. - \left(\frac{\rho u v_S^n + \rho u v_C^n}{2} - (|\bar{u}| + \bar{c}) \frac{\rho u_C^n - \rho u_S^n}{2} \right) \right) \end{aligned} \quad (10b)$$

$$\begin{aligned} \rho v_C^{n+1} &= \rho v_C^n - \\ & - \frac{\Delta t}{\Delta x} \left(\left(\frac{\rho u v_C^n + \rho u v_E^n}{2} - (|\bar{u}| + \bar{c}) \frac{\rho v_E^n - \rho v_C^n}{2} \right) \right. \\ & - \left(\frac{\rho u v_W^n + \rho u v_C^n}{2} - (|\bar{u}| + \bar{c}) \frac{\rho v_C^n - \rho v_W^n}{2} \right) \\ & + \left(\frac{(\rho v^2 + p)_C^n + (\rho v^2 + p)_N^n}{2} - (|\bar{u}| + \bar{c}) \frac{\rho v_N^n - \rho v_C^n}{2} \right) \\ & \left. - \left(\frac{(\rho v^2 + p)_S^n + (\rho v^2 + p)_C^n}{2} - (|\bar{u}| + \bar{c}) \frac{\rho v_C^n - \rho v_S^n}{2} \right) \right) \end{aligned} \quad (10c)$$

$$\begin{aligned} E_C^{n+1} &= E_C^n - \\ & - \frac{\Delta t}{\Delta x} \left(\left(\frac{(E+p)u_C^n + (E+p)u_E^n}{2} - (|\bar{u}| + \bar{c}) \frac{E_E^n - E_C^n}{2} \right) \right. \\ & - \left(\frac{(E+p)u_W^n + (E+p)u_C^n}{2} - (|\bar{u}| + \bar{c}) \frac{E_C^n - E_W^n}{2} \right) \\ & + \left(\frac{(E+p)v_C^n + (E+p)v_N^n}{2} - (|\bar{u}| + \bar{c}) \frac{E_N^n - E_C^n}{2} \right) \\ & \left. - \left(\frac{(E+p)v_S^n + (E+p)v_C^n}{2} - (|\bar{u}| + \bar{c}) \frac{E_C^n - E_S^n}{2} \right) \right) \end{aligned} \quad (10d)$$

Complex terms in the equations were marked with only one super- and subscript for better understanding, for example $(\rho u^2 + p)_C^n$ is equal to $\rho_C^n (u_C^n)^2 + p_C^n$. Notations $|\bar{u}|$ and $|\bar{c}|$ represent the average value of the u velocity component and the speed of sound at an interface, respectively.

A vast amount of experience has shown that these equations provide a stable discretization of the governing equations if the

time step obeys the following CFL condition:

$$\Delta t \leq \min_{(i,j) \in ([1,M] \times [1,N])} \frac{\min(\Delta x, \Delta y)}{|u_{i,j}| + c_{i,j}}. \quad (11)$$

C. The Second-order Scheme

The overall accuracy of the scheme can be raised to second-order if the spatial and the temporal derivatives are calculated by a second-order approximation. One way to satisfy the latter requirement is to perform a piecewise linear extrapolation of the primitive variables P_L and P_R at the two sides of the interface in (7). This procedure requires the introduction of additional cells with respect to the interface, i.e. cell LL (left to cell L) and cell RR (right to cell R). With these labels the reconstructed primitive variables are

$$P_L = P_L + \frac{g_L(\delta P_L, \delta P_C)}{2}, P_R = P_R - \frac{g_R(\delta P_C, \delta P_R)}{2}, \quad (12)$$

with

$$\delta P_L = P_L - P_{LL}, \delta P_C = P_R - P_L, \delta P_R = P_{RR} - P_R \quad (13)$$

while g_L and g_R are the limiter functions.

The previous scheme yields acceptable second-order time-accurate approximation of the solution, only if the variations in the flow field are smooth. However, the integral form of the governing equations admits discontinuous solutions as well, and in an important class of applications the solution contains shocks. In order to capture these discontinuities without spurious oscillations, in (12) we apply the *minmod* limiter function, also:

$$g_L(\delta P_L, \delta P_C) = \begin{cases} \delta P_L & \text{if } |\delta P_L| < |\delta P_C| \\ & \text{and } \delta P_L \delta P_C > 0 \\ \delta P_C & \text{if } |\delta P_C| < |\delta P_L| \\ & \text{and } \delta P_L \delta P_C > 0 \\ 0 & \text{if } \delta P_L \delta P_C \leq 0 \end{cases} \quad (14)$$

The function $g_R(\delta P_C, \delta P_R)$ can be defined analogously.

IV. IMPLEMENTATION ON FALCON CNN-UM ARCHITECTURE

The Falcon architecture [5] is an emulated digital implementation of CNN-UM array processor which uses the full signal range model. On this architecture the flexibility of simulators and computational power of analog architectures are mixed. Not only the size of templates and the computational precision can be configured but space-variant and non-linear templates can also be used.

The Euler equations are solved by a modified Falcon processor array in which the arithmetic unit was redesigned according to the discretized governing equations. Since each CNN cell has only one real output value, four layers are required to represent the variables ρ , ρu , ρv and E in case of Lax-Friedrichs approximation. In the first-order case the non-linear CNN templates acting on the ρu layer can easily be taken from (10b). Equations (15)-(17) show templates, in which cells of different layers are connected to the cell of layer ρu at position (i, j) .

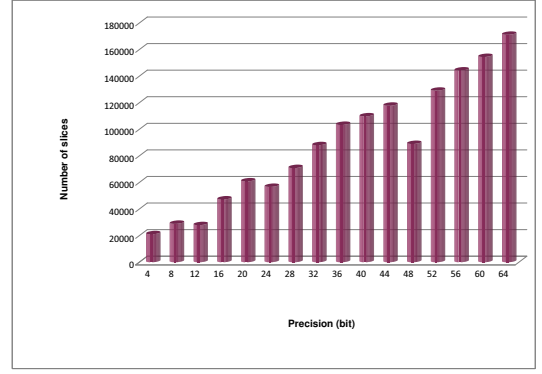


Fig. 2. Number of slices in the arithmetic unit

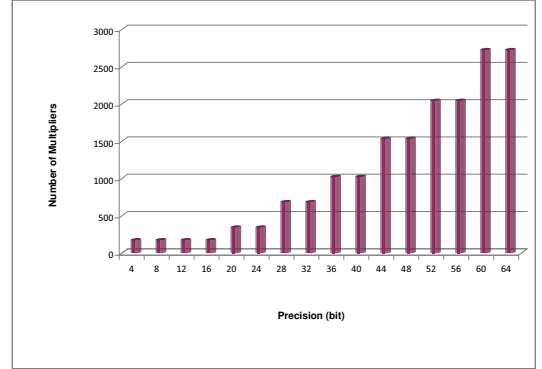


Fig. 3. Number of multipliers in the arithmetic unit

$$A_1^{\rho u} = \frac{1}{2\Delta x} \begin{bmatrix} 0 & 0 & 0 \\ \rho u^2 + p & 0 & -(\rho u^2 + p) \\ 0 & 0 & 0 \end{bmatrix} \quad (15)$$

$$A_2^{\rho u} = \frac{1}{2\Delta x} \begin{bmatrix} 0 & -\rho w & 0 \\ 0 & 0 & 0 \\ 0 & \rho v & 0 \end{bmatrix} \quad (16)$$

$$A_3^{\rho u} = \frac{1}{2\Delta x} \begin{bmatrix} 0 & \rho v & 0 \\ \rho u & -2\rho u - 2\rho v & \rho u \\ 0 & \rho v & 0 \end{bmatrix} \quad (17)$$

The template values for ρ , ρv and E layers can be defined analogously.

In the second-order case limiter function should be used on the primitive variables and the conservative variables are computed from these results. The limited values will be different for the four interfaces and cannot be reused in the computation of the neighboring cells. Therefore, this approach does not make it possible to derive CNN templates for the solution. However a specialized arithmetic unit still can be designed to solve it directly.

V. RESULTS AND PERFORMANCE

Our previous results solving the Euler equations on a rectangular grid shows that only few specialized arithmetic unit can be implemented even on the largest FPGAs. In

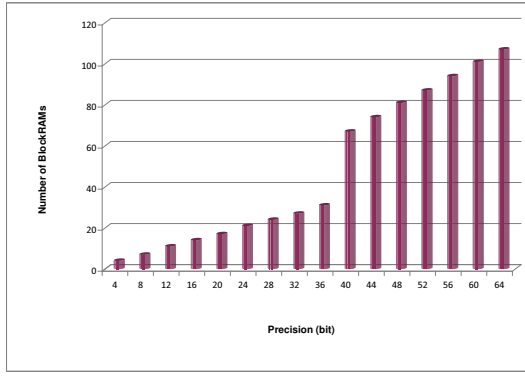


Fig. 4. Number of block-RAMs in the arithmetic unit

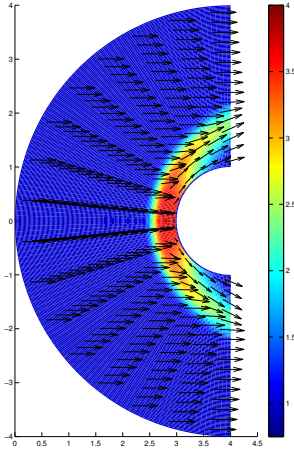


Fig. 5. Simulation around a cylinder in the initial state, 0.25 second, 0.5 second and in 1 second

the body fitted case additional area is required to take into account the geometry of the mesh during the computation. Symmetrical nature of the problem, which can be seen on the templates (15)-(17), enable further optimization of the arithmetic unit which compensates the area increase due to coordinate transformations. The number of slices, multipliers and block-RAMs of the arithmetic unit can be seen in Figure (2)-(4) respectively.

To show the efficiency of our solution a complex test case was used, in which a Mach 3 flow around a cylinder was computed. The direction of the flow is from left to right and the speed of the flow at the left boundary is 3-times the speed of sound constantly. The solution contains a symmetrical bow shock flow around the cylinder. Therefore, only the upper half of the region should be simulated. This problem was solved on a 128×256 grid, which was bent over the cylinder using 2ms timestep.

Result of the computation after 1s of simulation time is shown in Figure 5.

The experimental results of the average computation time

TABLE I
COMPARISON OF DIFFERENT HARDWARE IMPLEMENTATIONS

	Implementations		
	FPGA SX240T	Intel Core2Duo	Cell Processor 8 SPEs
Clock Frequency (MHz)	410	2000	3200
Million cell iteration/s	2500	1.3004	313.0319
Computation Time on 128×512 1 step (μs)	13.11	25197.92	104.68
Computation Time on 128×512 65536 steps (s)	0.86	1651.37	6.86
Speedup	1922.448	1	240.7151
Power Dissipation (W)	~ 30	65	85
Area (mm^2)	389	143	253

are compared to a Intel Core2Duo microprocessor is shown on Table I.

VI. CONCLUSION

The governing equations of two dimensional compressible Newtonian flows on body fitted mesh geometry were solved by using different kind of Xilinx Virtex 5 FPGAs. During the discretization body fitted mesh is defined, which can be more efficiently used to describe complex geometries, such as a cross section of a wing or a rocket nozzle. The first- and second-order Lax-Friedrichs scheme was used during the solutions. The main advantage of this method over the forward Euler method, which is used extensively in the computation of the CNN dynamics, is that this approximation is more robust in case of complex computational geometries and in presence of shock waves in the solutions.

Our solution was optimized for the dedicated and the general resources of the Xilinx Virtex 5 FPGA. Performance comparison showed that about a three order of magnitude can be achieved with respect to a high performance microprocessor and an order of magnitude can be achieved to the IBMs Cell processor.

The 2D part of the supersonic flow simulator was successfully implemented on the Xilinx Virtex 5 FPGA and significant performance improvement was achieved. We plan to extend our solution into 3D in the future to get a more usable simulator for real-life simulations.

REFERENCES

- [1] S. Kocsárdi, Z. Nagy, Á. Csík, and P. Szolgay, "Simulation of two-dimensional inviscid, adiabatic, compressible flows on emulated digital CNN-UM," *International Journal of Circuit Theory and Applications*, vol. DOI:10.1002/cta.565, 2008.
- [2] (2008) ClearSpeed Inc. [Online]. Available: <http://www.clearspeed.com/>
- [3] (2008) MathStar Inc. [Online]. Available: <http://www.mathstar.com/>
- [4] (2008) Tiler Inc. [Online]. Available: <http://www.tiler.com/>
- [5] Z. Nagy, Z. Vörösházi, and P. Szolgay, "Emulated Digital CNN-UM Solution of Partial Differential Equations," *IEEE J CASI*, vol. 34, no. 4, pp. 445-470, 2006.
- [6] J. D. Anderson, *Computational Fluid Dynamics - The Basics with Applications*. McGraw Hill, 1995.

Realizing large time constant in implantable neural signal recording application

Zoltán Kárász

(Supervisor: Dr. Péter Földesy)

karzo@digitus.itk.ppke.hu

Abstract – This work try to introduce the difficulties of the implantable neural recoding devices design. After to subject some of the main problem of this area, try to give the some possible solutions. Helps to understand the optimization steps that needed to reach the state of the art specification. Particular attention the large time constant filtering solutions, which are ought to sense the low frequency input signal.

Index Terms – Large time constant, pseudo-resistor, neuro-amplifier

I. INTRODUCTION

The biomedical field is one of the most dynamically developing research area in the analog IC design, especially the implantable solutions with battery less and low-power implementations. Creating a brain-computer interface gives a powerful tool for the medical science to observing several neurological diseases, like the epilepsy. The examination procedures needs more time for the functional result than available using other observing techniques as the FMRI, but need more detailed result than using simple EEG. Although the portability of the measuring instrument is not an important issue for the animal studies, but in the human experiments rather it is.

For the purpose to overstep this toughness the latest medical research attempt to implant micro-sensors. Our interest indeed the cortical arrays, which causes minimal structural damages in the analyzed region. From the engineer's aspect measuring the brain activity could be simplified to an electrical connection between the brain tissue and the electrode. After the perception it's necessary to convey the information. Instead of implement a naive measuring subsystem it is possible to do a preprocessing and spare the energy when we transfer only the minimal data about the measured brain signals. The implantable neural recoding devices have to achieve strict specifications. Including the power consumption, noise and distortion requirements, defined maximal thermal dissipation and specified input frequency range. The basic amplifier architecture is an OTA based capacitive feedback single input differential output amplifier. Before presented the actual implementation lets start the features of input signal, this are necessary to understand the requirements of the amplifier.

Neural Signal

However a neuron can produce 100 mV internal voltage changes relative to the extracellular fluid that can be recording directly only with patch-clamp electrodes, but the in-vivo chronic recordings using multi-electrode arrays which able to utilize the smaller extracellular potentials from several micrometers from the cell.

The amplitude of the signal is on the order of 100 μ V (Fig. 1)[3].

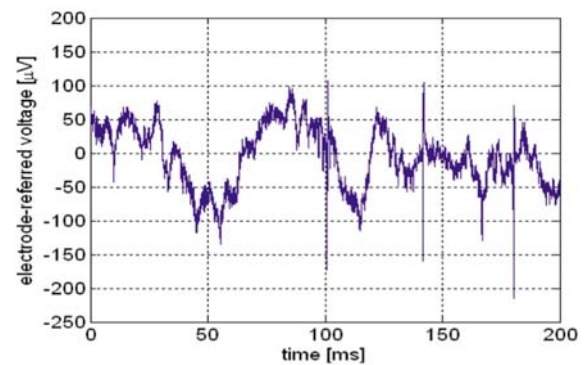


Fig. 1 Typical neural signal

The neural action potentials called „spikes”. The neurons are rarely fire more rapidly than 100 spikes per second, although the rapid bursts of several spikes are possible. Neurons produce spikes of nearly identical amplitude and duration and information is encoded in the timing of spikes.

The low-frequency oscillations are (under 200 Hz) known as Local Field Potentials (LFPs). It is arise from the synchronous activity of many neurons in one region of the brain. These neurons are far away from the electrode for their individual action potentials to be detected, but the many neighboring cells create a large signal that is easily detected. The energy of LFP signal is correlates with specific event like sleeping (0.5 - 25 Hz) or pathological state as the epilepsy (8 - 40 Hz) and the Parkinson prediction (15 - 30 Hz) [2]. LFPs are a robust signal. In some experiments using electrode arrays and scar the tissue around microelectrode tips. This scar tissue tends to attenuate spike signals from nearby neurons, but LFP signals are less affected.

In many applications, it is desirable to separate LFP and spike signals so they may be analyzed separately. This is easily accomplished by linear filtering since LFPs occupy frequencies from approximately 0.5 – 200 Hz, while spikes have energy concentrated in the 300 Hz – 7 kHz range. When multi-electrode arrays are placed in the brain, it is common for some electrodes to detect spikes from two to four distinct neurons, while other electrodes may see no resolvable spikes.

Amplifier Requirements

The realization of large time constants is fundamental for design filters with very low cut-off frequencies especially in implantable biomedical sensors. The filters are required to be tunable. In addition, realizations with low power dissipation and small size are also critical.

Several approaches for the design of integrators with very large time constants have been reported. [1, 16, 20] The most obvious is the trivial solution to employ on-chip physical resistor and capacitor, conversely this would require large chip area and it would not be tunable. The possible solutions can be categorized into pseudo-resistor implementations [1, 2, 5 - 11], switched-capacitor (SC) methods [16] and operational trans-conductance amplifier capacitor (OTA-C) techniques with very small trans-conductance's [20] to allow the on-chip capacitance to be kept manageably low. The exact value of the time constant is not critical.

MOS Pseudo-Resistor

The most prevalent solutions contains pseudo-resistor. That is construing the traits of this solution, like the minimal size, simplicity and the outstanding effective resistance.

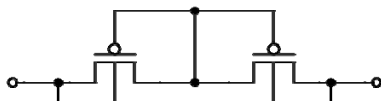


Fig. 2 Schematic of the pseudo-resistor element

The basic symmetric element contains two transistors that are connected as a MOS diode and a parasitic source-bulk diode connected in anti-parallel. If the voltage across the device is small enough, then neither diode will conduct strongly, and the effective resistance is very large ($> 10 \text{ G}\Omega$).

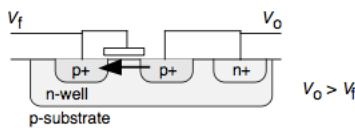


Fig. 3 Diode-connected MOS transistor

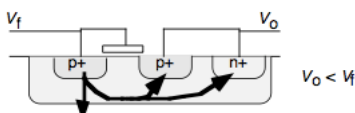


Fig. 4 PN junction is forward-biased

For voltage polarity $V_o > V_f$ across the element (Fig. 3) the side of V_o in the MOS case acts as the source of the transistor. For the opposite polarity (Fig. 4), the driven side is a forward-biased source-gate junction.

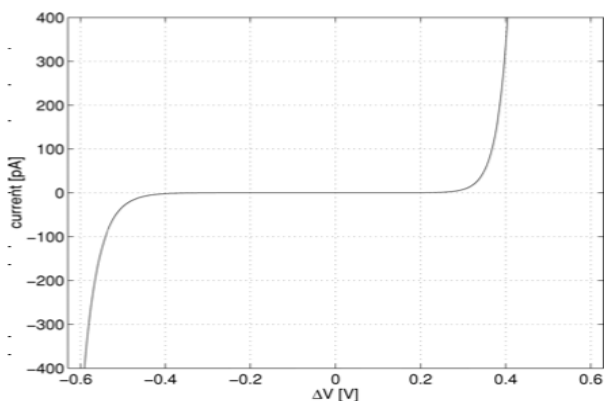


Fig 5. Current voltage relation

The current-voltage relationship (Fig. 5) [3, 7] of the expansive element means that the effective resistance of the element is huge for small signals and small for large signals. Therefore the adaptation is slow for small signals and fast for large signals.

Unfortunately this kind of resistance exhibits a nonlinear behaving. The variation of the resistance in the feedback loop means the transfer-function would not be permanent at the whole working period. If the cut of frequency is altered that would increase the whole distortion. This effect impair significant in the lower frequency range (under 100 Hz).

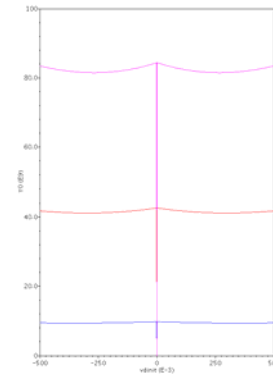


Fig. 6 Corner variation

Another relevant problem with this solution has large impact to the technological parameter. Although using any active element cause similar problem in the corner state. The biomedical applications have strict operating requirement about the temperature ($30\text{-}44 \text{ C}^\circ$) that actually reduce the corner variation, but still has $+40\%$ and -20% .

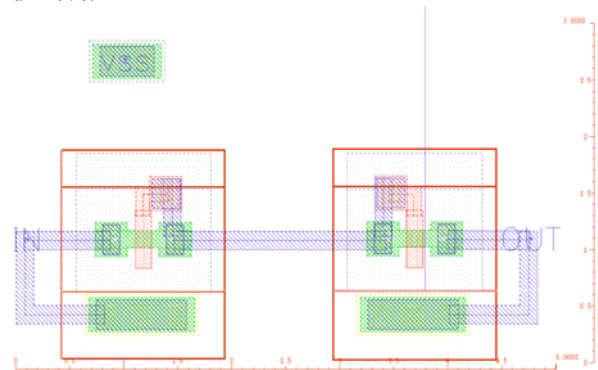


Fig. 7 Layout for the 2 transistor pseudo resistor

In case of the two transistors pseudo resistor the total parasitic capacitance is 365 aF, that is negligible up to the other element of the circuit.

For summation the pseudo-resistance has good size and parasitic values, but it also has some serious non-ideal behavior, which means poor robustness and bad distortion in the LFP range.

II. BASIC NEURO-AMPLIFIER TOPOLOGY

The amplifier is based around an operational transconductance amplifier (OTA) that produces a current applied to its input. A capacitive feedback network consisting of C_1 and C_2 capacitors sets the mid-band gain of the amplifier. The input is capacitively coupled through C_1 , so any dc offset from the electrode-tissue

interface is removed. C_1 should be made much smaller than the electrode impedance to minimize signal attenuation. The R_2 elements shown in the feedback loop set the low-frequency amplifier cutoff.

The approximate transfer function is given by

$$\frac{v_{out+} - v_{out-}}{v_{in}} = \frac{C_1}{C_2} \frac{1 - \frac{sC_2}{G_m}}{\left(\frac{1}{sR_2C_2 + 1}\right) \left(s\frac{C_L C_1}{G_m C_2} + 1\right)} \quad (1)$$

The midband gain A_M is set by the capacitance ratio C_1/C_2 , and the gain is flat between the lower and upper cutoff frequencies f_L and f_H . The product of R_2 and C_2 determines the lower cutoff frequency, while the upper cutoff is determined by the load capacitance C_L , the OTA trans-conductance G_m , and the mid-band gain. Capacitive feed introduces a right-half-plane zero at f_z , but this zero can be very at high frequency by setting

$$C_2 \ll \sqrt{C_1 C_L} \quad (2)$$

so that it has little practical effect on amplifier operation. The OTA contributes noise primarily between f_L and f_H . Below a particular frequency, the noise contribution from v_{nR} will dominate; we denote this frequency f_{corner} . If R_2 is implemented as a real resistor so that its noise spectral density is

$$v_{nR}^2(f) = 4kTR_2 \quad (3)$$

and $C_1 \gg C_2, C_{in}$, then f_{corner} is approximately

$$f_{corner} \approx \sqrt{\frac{3C_L}{2C_1}} f_L f_H \quad (4)$$

A similar result is obtained for pseudo resistor element used as R_2 in. To minimize the noise contribution from the R_2 elements, we should ensure that $f_{corner} \ll f_H$.

If the noise contribution from R_2 is negligible and $C_1 \gg C_2, C_{in}$, then the output rms noise voltage of the neural amplifier is dominated by the noise from the OTA.

$$v_{nia}^2 = \frac{16kT}{3g_{m1}} \left(1 + 2\frac{g_{m3}}{g_{m1}} + \frac{g_{m7}}{g_{m1}}\right) \quad (5)$$

where g_{m1} is the trans-conductance of the input devices M_1 and M_2 . The noise of the cascode transistors is negligible.

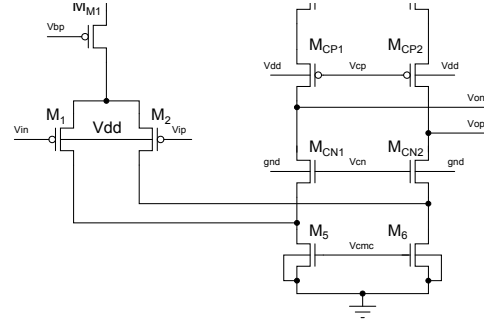


Fig. 9 Differential input cascoded OTA

That case the load capacitance is determined by

$$C_L = \frac{4KT}{V_{mi}^2 3A_M} \quad (6)$$

III. Proposed Series-Connected Digitally Controllable Pseudo-Resistor

As we can see it before to avoid the additional distortion it is necessary to minimize the resistance variation. It is possible to utilize the tradeoff between the noise and distortion. Using more pseudo resistor element in series helps the decreasing this effect (*Fig. 10*).

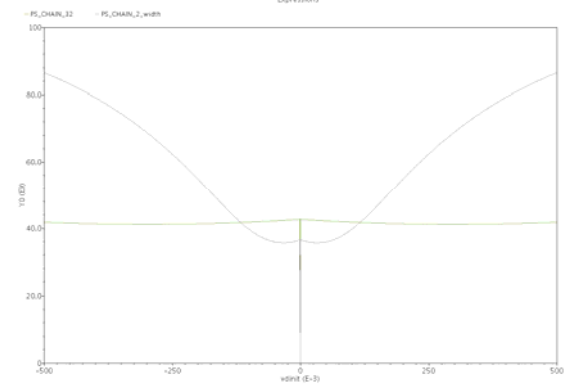


Fig. 10 Resistance variation, PS2 vs. PS32

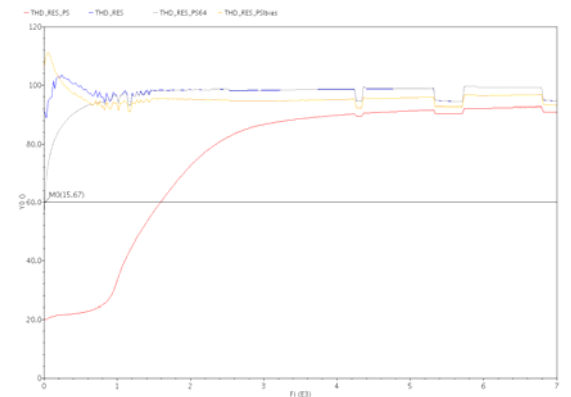


Fig. 11 THD of the Harrison-topology with different resistor implementation

To able to fulfill the accuracy requirements in the whole system we need satisfy the total harmonic distortion (THD) enough good on the every frequency.

For the 8-bit accuracy we need keep at least the 60 dB level. In other solutions, which interest only the AP-s could be use less number of element (Fig 11).

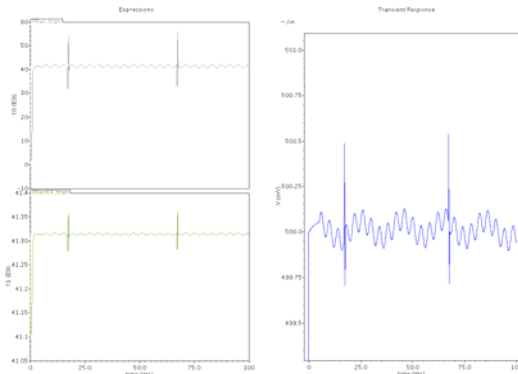


Fig. 12 Resistance variance by time for “real” input (PS2 vs. PS64)

Because of the high corner deviations and the frequency tune-ability another important aspect in the design is the resistance controlling.

Gated Pseudo-Resistance

It possible to give controllability to the resistance if we use switches to shortcut the remaining part of the chain (Fig. 13). This gated structure needs to be pre-calibrate at least the required resistance plus the corner variation. Using more pseudo resistor the overall resistance will be increasing as the distortion is decreasing. It is not needed to worry about the minimal resistance case, because that's the AP range not the LFP one. Otherwise the large number of the series connected pseudo resistor still doesn't have large overhead in the area occupation neither the parasitic.

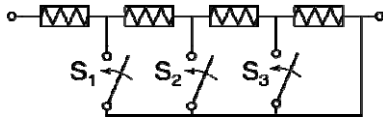


Fig. 13 Gated pseudo-resistance

For practical reasons we used transfer gates for switches. They necessary to have large impedance that could be commensurable the pseudo-resistances, otherwise leakage will reduce the overall resistance. The transfer gates must be to optimize to the OFF resistance oppositely the general usage.

It doesn't effective to use identical resistors if we like to tuning and compensating with the same chain (Fig. 14). The exact distribution is depending on the required cut-off frequency and the degree of the corner deviations.

Finally we got a programmable solution that helps us to increase the robustness against the technology parameter variation, to reduce the significant distortion and gives possibility to choose the cut-off frequency.

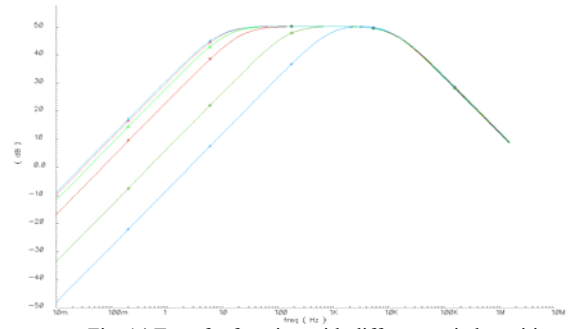


Fig. 14 Transfer function with different switch position

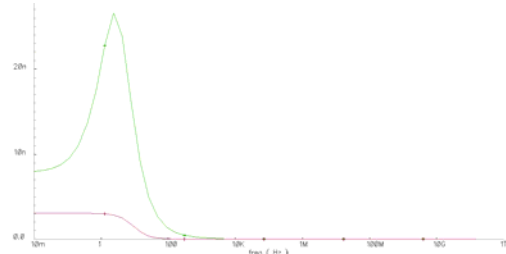


Fig. 15 Input-referred noise (ideal vs. real)

V. RESULTS

In this paper presented an integrated low noise amplifier circuit for the battery less implantable neural recording, and reviewed the most important design considerations. The MOS pseudo resistor chain is genuine innovation which is not used any other solutions on this area. The comparison between the switched-capacitance, the pseudo resistance and the modified OTA topologies as generally are not definite. As long as the current cancellation and division generate a continuously current consumption and not gives any chance for tuning the transfer-function, till then the switched capacitor provide a fine tuning method but generates high distortion. The basic MOS pseudo resistance not able the handle the low frequency input because the bad distortion and sensitive for the corner variation as a SC resistances. Finally the gated chain could be the optimal solution. It gives the tuning range to decreasing the corner effect and to handle the local field potential range.

VI. REFERENCES

- [1] Gozzini et al. Linear transistor with rail-to-rail input swing for very large time constant applications. Electronics Letters AB - ER - (2006) vol. 42 (19) pp. 1069- 1070
- [2] Rieger et al. A 230-nW 10-s time constant CMOS integrator for an adaptive nerve signal amplifier. Solid-State Circuits, IEEE Journal of (2004) vol. 39 (11) pp. 1968- 1975
- [3] Harrison and Charles. A low-power low-noise CMOS amplifier for neural recording applications. Solid-State Circuits, IEEE Journal of (2003) vol. 38 (6) pp. 958- 965
- [4] Zou et al. A 1-V 1.1- μ W sensor interface IC for wearable biomedical devices. Circuits and Systems, 2008. ISCAS 2008. IEEE International Symposium on (2008) pp. 2725-2728
- [5] Avestruz et al. A 5 μ W/Channel Spectral Analysis IC for Chronic Bidirectional Brain-Machine Interfaces. Solid-State Circuits, IEEE Journal of (2008) vol. 43 (12) pp. 3006-3024
- [6] Yin and Ghovanloo. A Low-Noise Preamplifier with Adjustable Gain and Bandwidth for Biopotential Recording Applications. Circuits and Systems, 2007. ISCAS 2007. IEEE International Symposium on (2007) pp. 321-324

Power Amplifier at Low Frequency with Low Output Power

László Tamás Kozák

(Supervisors: Dr. Péter Földesy and Dr. Tamás Roska)

kozla@digitus.itk.ppke.hu

Abstract—In the increasing number of wireless biological applications where the main goals are to increase the battery lifetime and decrease the heat, to avoid the death of the cells, new effective topologies are necessary. In many cases these systems work as low output power as let the designer avoid the stereotype Power Amplifier (PA) topology. However the efficiency of this kind of circuit without large and high quality factor inductor is quite low. In this paper a possible way is presented how the efficiency can be increased at low frequency where the size of the passives become too large to let the inductor be on-chip.

Index Terms—Power Amplifier, Drain Efficiency, Power Added Efficiency, Distortion

I. INTRODUCTION

Power amplifiers (PA) are the last blocks of RF transmitters. The main challenges on these blocks to maximize the output power (P_{out}) and the efficiency. The second is fairly important since PA may consume more power than the other parts of the circuit together. If the requirement does not let the designer use large passive off-chip elements the efficiency (and linearity) decrease significantly. Additionally, if the required output power is low enough the usage of on-chip reactant elements can not give much better result and if the frequency is low the size of the inductor (BFL) included by the amplifier must be increased significantly for the same efficiency.

In this paper a procedure is presented how the efficiency can be improved without passive elements when the frequency and the necessary output power is low.

II. SIMPLE INVERTER

If the designer do not use passive elements only linear PA versions can be taken into account since it does not need output filtering (except DC decoupling). However the theoretical maximum efficiency of a linear PA is 50% [1]. With some modification this value can be improved (discussed later).

An other important property of a linear PA, we have to deal with, is the linearity. The elimination of higher order harmonics at the output of is important because of two things.

- The phase difference between the fundamental and higher harmonics may result significant P_{out} drop.
- Generation of higher harmonics means more power consumption for signals are not necessary.

The other important thing, must be taken into account, is the output swing that is a bound of the P_{out} , thus also the efficiency. Namely, higher output signal means higher distortion. Unfortunately, these two things show opposite design directions.

In this paper an inverter has been chosen as basic stage because of its symmetry, amplification and linearity.

III. LINEARITY

First we make our calculations with long channel devices where V_{GS} has neglectable effect on the Early-voltage. In other words, we make the calculations with constant λ . After that we make some modifications when the $\lambda(V_{GS})$ is not constant. Namely, the channel is much shorter and we have to consider the third order term. According to definition [2] if y can be expressed by x the second order distortion is the following:

$$y - y_q = a_0 + a_1x + a_2x^2$$

$$HD_2 = \frac{1}{2} \frac{a_2}{a_1}$$

For linearity analysis we are going to determine the distortion in three possible cases.

- 1) Both the nMOS and the pMOS is in saturation.
- 2) One of them is in triode region.
- 3) One of them is in off state.

(Note: We will make the calculation in that case when the output drops. Because of symmetry the other direction gives the same result.)

A. Both transistors are in saturation

The calculation method of distortion is not difficult. First we set up a quiescent point and determine how the output depends on the input. Second, we go away from the quiescent point and do the same. Lastly we subtract the first one from the second.

$$di_{oq} = i_{SDP} - i_{DSN} =$$

$$\frac{K_P}{2} \left[-\frac{V_{DD}}{2} - V_{THP} \right]^2 [1 - \lambda V_{DD}] -$$

$$\frac{K_N}{2} \left[\frac{V_{DD}}{2} - V_{THN} \right]^2 [1 + \lambda V_{DD}]$$

$$di_o = \frac{K_P}{2} \left[-\frac{V_{DD}}{2} + dv_i - V_{THP} \right]^2 [1 - \lambda(V_{DD} + dv_o)] -$$

$$\frac{K_N}{2} \left[\frac{V_{DD}}{2} + dv_i - V_{THN} \right]^2 [1 - \lambda(V_{DD} - dv_o)]$$

$$\delta i = di_o - di_{oq} = C +$$

$$dv_i \left[\frac{K_P}{2} (-V_{DD} - 2V_{THP}) \left(1 - \lambda \left(\frac{V_{DD}}{2} + dv_o \right) \right) - \right.$$

$$\left. \frac{K_N}{2} (V_{DD} + 2V_{THN}) \left(1 + \lambda \left(\frac{V_{DD}}{2} - dv_o \right) \right) \right] +$$

$$dv_i^2 \left[\frac{K_P}{2} \left(1 - \lambda \left(\frac{V_{DD}}{2} + dv_o \right) \right) - \frac{K_N}{2} \left(1 + \lambda \left(\frac{V_{DD}}{2} - dv_o \right) \right) \right]$$

With proper transistor size we can ensure the output voltage is exactly $V_{DD}/2$ while the input voltage is also $V_{DD}/2$. At

this operation point there is no output current. Since $di_{oq} = 0$ we have to concentrate only to di_o . Let us define c_1 and c_2 as:

$$c_1 = \frac{K_P}{2} \left(1 - \lambda \left(\frac{V_{DD}}{2} + dv_o \right) \right)$$

$$c_2 = \frac{K_N}{2} \left(1 + \lambda \left(\frac{V_{DD}}{2} - dv_o \right) \right)$$

With the help of previously defined constants the second order distortion can be expressed as:

$$HD_2 = \frac{1}{2} \cdot \frac{c_1 - c_2}{c_1(-V_{DD} - 2V_{THP}) - c_2(V_{DD} + 2V_{THN})}$$

If we suppose that $-V_{THP} \simeq V_{THN}$, then

$$HD_2 \simeq \frac{1}{2} \cdot \frac{c_1 - c_2}{-V_{DD}(c_1 + c_2) - 2V_{TH}(c_1 + c_2)} = \frac{1}{2} \cdot \frac{c_1 - c_2}{c_1 + c_2} \cdot \frac{1}{-V_{DD} + 2V_{TH}}$$

If the threshold voltage of the transistors are approximately equal the $K_N \simeq K_P$ assumption can be used. (We can do this because the input and output are $V_{DD}/2$ therefore I_{SDP} must be equal to I_{SDN} . It can be only if the difference between K_P and K_N is not significant.)

With this assumption

$$\frac{c_1 - c_2}{c_1 + c_2} = \frac{-\lambda V_{DD}}{2(1 - \lambda dv_o)}$$

$$HD_2 = \frac{-\lambda V_{DD}}{4(1 - \lambda dv_o)} \cdot \frac{1}{-V_{DD} + 2V_{TH}}$$

$$= \frac{\lambda V_{DD}}{4(1 - \lambda dv_o)(V_{DD} - 2V_{TH})} \simeq \frac{1}{4 \left(\frac{1}{\lambda} - \frac{2V_{TH}}{\lambda V_{DD}} \right)}$$

$$HD_2 \simeq \frac{1}{\frac{4}{\lambda} \left(1 - \frac{2V_{TH}}{V_{DD}} \right)}$$

(Note: The dv_o was neglected.) It can be seen lower distortion means lower threshold, higher supply or longer channel.

B. One is in saturation the other is in triode region

Now we are going to do the same method but now the output voltage must be high enough to make one transistor be in triode region.

$$di_{oq} = i_{SDP} - i_{SDN} =$$

$$\frac{K_P}{2} \left[-\frac{V_{DD}}{2} - V_{THP} \right]^2 [1 - \lambda V_{DD}] -$$

$$\frac{K_N}{2} \left[\frac{V_{DD}}{2} - V_{THN} \right]^2 [1 + \lambda V_{DD}]$$

$$di_o = \frac{K_P}{2} \left[-\frac{V_{DD}}{2} + dv_i - V_{THP} \right]^2 [1 - \lambda (V_{DD} + dv_o)] -$$

$$K_N \left[\frac{V_{DD}}{4} + dv_i - V_{THN} + \frac{dv_o}{2} \right] \left[\frac{V_{DD}}{2} - dv_o \right]$$

(Note: As in the previous section the $\delta i = di_o - di_{oq} = di_o$.)

$$\delta i = \frac{K_P}{2} \left[-\frac{V_{DD}}{2} + dv_i - V_{THP} \right]^2 [1 - \lambda (V_{DD} + dv_o)] -$$

$$K_N \left[\frac{V_{DD}}{4} + dv_i - V_{THN} + \frac{dv_o}{2} \right] \left[\frac{V_{DD}}{2} - dv_o \right]$$

$$\delta i = C + f(dv_i, dv_i^2)$$

$$f(dv_i, dv_i^2) =$$

$$dv_i \left[-\frac{K_P}{2} (V_{DD} - 2V_{THP}) \left(1 - \lambda \left(\frac{V_{DD}}{2} + dv_o \right) \right) - \right.$$

$$K_N \left(\frac{V_{DD}}{2} - dv_o \right) \left. \right] +$$

$$dv_i^2 \left[\frac{K_P}{2} \left(1 - \lambda \left(\frac{V_{DD}}{2} + dv_o \right) \right) \right]$$

From this we can write:

$$HD_2 = \frac{1}{2} \cdot \frac{K_P [1 - \lambda (V_{DD}/2 + dv_o)]}{-K_P (V_{DD} + 2V_{THP}) (1 - \lambda (V_{DD}/2 + dv_o)) - K_N [V_{DD} - 2dv_o]}$$

$$\frac{1}{2} \cdot \frac{1}{- (V_{DD} + 2V_{THP}) - \frac{K_N (V_{DD} - 2dv_o)}{K_P (1 - \lambda (V_{DD}/2 + dv_o))}}$$

As we expected as dv_o tends to its maximum ($V_{DD}/2$) than the distortion increases. And similarly to the previous case higher V_{TH} means larger distortion. ($V_{THP} < 0$)

C. One is in off state the other is in saturation

In the previous section we have seen the transistor in triode region has no second order term just like in this region. Therefore the result from distortion's point of view would be the same.

IV. THEORETICAL MAXIMUM DRAIN EFFICIENCY

According to definition the drain efficiency $\eta = P_{out}/P_T$ and the power added efficiency $PAE = (P_{out} - P_{in})/P_T$ where P_T is the total average power injected into the circuit. Let us suppose the input voltage is sinusoid. The general case, when dv_o does not reach its maximum, depicted in Fig.2.a. Since the transistor behavior is nonlinear the average current can be less than $di_{o,max}/2$.

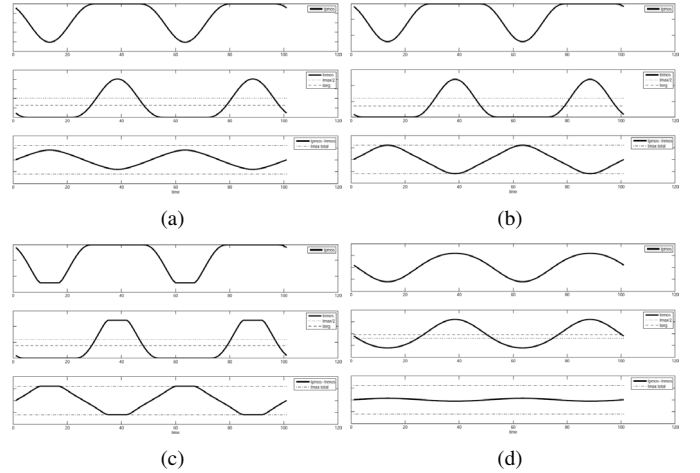


Fig. 1. Possible output regions: (a) general, (b) maximum, (c) above maximum, (d) wrong

Now the quiescent point is $V_{DD}/2$ and the output signal is a perfect sinusoid with maximum $V_{DD}/2$ amplitude. (The maximum output voltage is going to be calculated in the next section respect to the load.) Additionally, the output decoupling capacitor (C_{DC}) is large enough the phase shift between dv_o and di_o can be neglected ($X_{C_{DC}} \ll R_L$). The maximum input voltage, can be adjusted by feedback (we discuss later), is considered as $V_{DD}/2 - dv_i = V_{TH}$. In this case the output currents are as in Fig.2.b and the mean value of the total current can be written as:

$$\begin{aligned}
& \frac{1}{T} \int_0^T \left(\frac{\sqrt{d_{i_o,max}}}{2} \sin(t) \pm \frac{\sqrt{d_{i_o,max}}}{2} \right)^2 dt = \\
& \frac{1}{T} \frac{d_{i_o,max}}{4} \int_0^T (\sin^2(t) \pm 2\sin(t) + 1) dt = \\
& \frac{1}{T} \frac{d_{i_o,max}}{4} \int_0^T (\sin^2(t) + 1) dt = \\
& \frac{1}{T} \frac{d_{i_o,max}}{8} \int_0^T (1 - \cos(2t)) dt + \frac{1}{T} \frac{d_{i_o,max}}{4} \int_0^T 1 dt = \\
& \frac{1}{T} \frac{d_{i_o,max}}{8} [t]_0^T + \frac{1}{T} \frac{d_{i_o,max}}{4} [t]_0^T = \frac{3}{8} d_{i_o,max} = I_{avg,max}
\end{aligned}$$

(Note: We assume the transistors are ideal with infinity Early-voltage and the $d_{i_o,max}$ includes the constants of the MOS transistor ($\mu, C_{ox}, W/L$). The ' \pm ' means the average of pMOS and nMOS current are the same.) Since $P_T = V_{DD} \cdot I_{avg,max}$ and $d_{i_o,max} \cdot R_L = dv_{o,max} = V_{DD}/2$ we can write the following.

$$\begin{aligned}
\eta_{max} &= \frac{P_{out,max}}{P_T} = \frac{d_{i_o,max}^2 \cdot R_L}{2 \cdot V_{DD} \cdot I_{avg,max}} = \\
&= \frac{8 \cdot d_{i_o,max}^2 \cdot R_L}{6 \cdot V_{DD} \cdot d_{i_o,max}} = \frac{4 \cdot d_{i_o,max} \cdot R_L}{3 \cdot V_{DD}} = 66.6\%
\end{aligned}$$

Important to note the input voltage can be chosen wrongly respect to the parameters (V_{DD}, V_{TH}) of the circuit. That is the average total current is larger than the half of the maximum (Fig.2.d), so the maximum drain efficiency can not reach even the 50%.

V. MAXIMUM OUTPUT POWER AND SUPPLY VOLTAGE

The main cause of the efficiency degradation is the high quiescent DC current. Namely, if $V_{DD}/2 \gg V_{TH}$ a significant DC current flows through the transistors that degrades the efficiency. We can degrade the DC current with with an optimum threshold ($V_{DD}/2 \simeq V_{TH}$).

Unfortunately the maximum threshold value of the technology cannot be modified and if the required output voltage larger then this value the efficiency decays. Although there are lots of solutions to increase the threshold voltage each of them have have a drawback. One way to do that is in the next figures. We are going to calculate what is the maximum dv_o in both case.

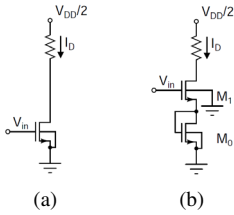


Fig. 2. V_{TH} modification

In case a)

$$\begin{aligned}
dv_o &= R_L K_N \left(\frac{V_{DD}}{4} + dv_i - V_{TH_N} + \frac{dv_o}{2} \right) \left(\frac{V_{DD}}{2} - dv_o \right) \\
\frac{\partial dv_o}{\partial dv_i} &= R_L K_N \left(\frac{V_{DD}}{2} - dv_o - dv_i \frac{\partial dv_o}{\partial dv_i} + V_{TH_N} \frac{\partial dv_o}{\partial dv_i} \right) \\
\frac{\partial dv_o}{\partial dv_i} &= \frac{R_L K_N (V_{DD}/2 - dv_o)}{1 + R_L K_N (dv_i - V_{TH_N} + dv_o)}
\end{aligned}$$

with the same method the case b) gives the following solution.

$$\begin{aligned}
V_{TH1} &= V_{TH1} + V_{GS0} = V_{TH1} + V_{TH0} \\
\frac{\partial dv_o}{\partial dv_i} &= \frac{R_L K_N (V_{DD}/2 - dv_o - V_{TH0})}{1 + R_L K_N (dv_i - V_{TH_N} + dv_o - V_{TH0})}
\end{aligned}$$

It can be seen that in this case higher threshold means lower output voltage.

(Note: In Fig.3.b V_{TH1} is increased by body effect but the alteration is not significant.)

VI. NONLINEAR NEGATIVE FEEDBACK

In this section we are going to improve the efficiency in other way. As we wrote in the first section the distortion have significant effect on efficiency. With a simple negative feedback (FB) we are able to make more linear circuits but with the cost of decline of amplification. Our target is not only to decrease the distortion but also to eliminate the higher order harmonics. Therefore we need a FB (non-linear) characteristic similar to the basic stage characteristic. Thus we have chosen the circuit depicted in figure below [3] due to its advantages respect to others.

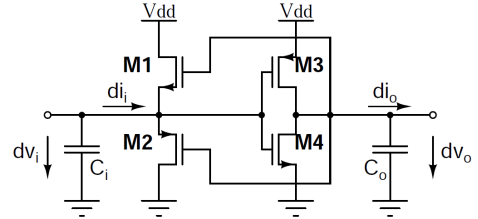


Fig. 3. Nonlinear feedback

The FB has three region of operation.

- 1) The dv_o is not large enough to open the FB transistors. In this region there is no FB but in this region the distortion of the inverter is not significant.
- 2) One of the FB transistors is in triode region. This is more or less equivalent to a linear FB.
- 3) One of the FB transistors is in saturation. This is the most important case, therefore we will deal with this point in more details.

We are going to discuss only the third point because this is the most critical. (We are going to make the calculation in the case when pMOS is active and nMOS is in triode region. Because of the symmetry the other way gives the same result.)

A. Second order term for long channel transistors

In ideal case, when the Early-voltage is constant (does not depend on V_{GS}) only the second order term must be reduced.

$$\begin{aligned}
d_{i_{FB}} &= \frac{K_P}{2} [-dv_o - dv_i - V_{TH_P}]^2 \left[1 - \lambda \left(\frac{V_{DD}}{2} + dv_i \right) \right] \\
g(dv_o, dv_o^2) &= C + \\
&dv_o \left[K_P (-dv_i - V_{TH_P}) \left(1 - \lambda \left(\frac{V_{DD}}{2} + dv_i \right) \right) \right] + \\
&dv_o^2 \left[\frac{K_P}{2} \left(1 - \lambda \left(\frac{V_{DD}}{2} + dv_i \right) \right) \right]
\end{aligned}$$

(Note: If the signal is sinusoid the amplitude of the current is constant times the voltage amplitude. That is: $di_i = \cos(\omega t)$ means $dvi = \sin(\omega t)/(C_i \cdot \omega)$)

The basic stage behavior in the third case can be written as:

$$\begin{aligned}
di_o &= i_{SDP} - i_{DSN} = \\
&\frac{K_P}{2} \left[-\frac{V_{DD}}{2} + dv_i - V_{THP} \right]^2 \left[1 - \lambda \left(\frac{V_{DD}}{2} + dv_o \right) \right] - \\
&K_N \left[-\frac{V_{DD}}{4} + dv_i - \frac{dv_o}{2} - V_{THN} \right] \left[\frac{V_{DD}}{2} - dv_o \right] \\
di_o &= C + f(dv_i, dv_i^2) \\
f(dv_i, dv_i^2) &= \\
&dv_i \left[\frac{K_P}{2} (-V_{DD} - 2V_{THP}) \left(1 - \lambda \left(\frac{V_{DD}}{2} + dv_o \right) \right) - \right. \\
&\quad \left. K_N \left(\frac{V_{DD}}{2} - dv_o \right) \right] + \\
&dv_i^2 \left[\frac{K_P}{2} \left(1 - \lambda \left(\frac{V_{DD}}{2} + dv_o \right) \right) \right]
\end{aligned}$$

Now let us suppose that $dv_o = A \cdot dv_i$ where $dv_i > 0$, $dv_o > 0$ and $A < 0$. With this assumption we can write the following equation.

$$C_{FF} + f(dv_i, dv_i^2) = -A(C_{FB} + g(dv_o, dv_o^2))$$

$$\begin{aligned}
C_{FF} + dv_i \left[\frac{K_{PFF}}{2} (-V_{DD} - 2V_{THP}) \left(1 - \lambda \left(\frac{V_{DD}}{2} + dv_o \right) \right) - \right. \\
\left. K_{NFF} \left(\frac{V_{DD}}{2} - dv_o \right) \right] + dv_i^2 \left[\frac{K_{PFF}}{2} \left(1 - \lambda \left(\frac{V_{DD}}{2} + dv_o \right) \right) \right] = \\
-A \left[C_{FB} + dv_o \left[K_{PFB} (-dv_i - V_{THP}) \left(1 - \lambda \left(\frac{V_{DD}}{2} + dv_i \right) \right) \right] + \right. \\
\left. dv_o^2 \left[\frac{K_{PFB}}{2} \left(1 - \lambda \left(\frac{V_{DD}}{2} + dv_i \right) \right) \right] \right]
\end{aligned}$$

Now we concentrate only to the second order terms and use the assumption $dv_o = -A \cdot dv_i$ (i.e. the characteristic is linear). If the circuit is linear then the second order terms must also fulfill this assumption. So we can write the following equation.

$$\begin{aligned}
dv_i^2 \left[\frac{K_{PFF}}{2} \left(1 - \lambda \left(\frac{V_{DD}}{2} + dv_o \right) \right) \right] = \\
-A dv_o^2 \left[\frac{K_{PFB}}{2} \left(1 - \lambda \left(\frac{V_{DD}}{2} + dv_i \right) \right) \right] \\
K_{PFF} \left(1 - \lambda \left(\frac{V_{DD}}{2} + dv_o \right) \right) = -A^3 K_{PFB} \left(1 - \lambda \left(\frac{V_{DD}}{2} + dv_i \right) \right) \\
\left(\frac{1}{\lambda} - \frac{V_{DD}}{2} \right) \frac{1}{dv_o} = \frac{K_{PFF} - A^2 K_{PFB}}{K_{PFF} + A^3 K_{PFB}}
\end{aligned}$$

At the beginning of the design we determined the dv_i , the dv_o and the K_{PFF} . Now we would like to know what is the connection between them and the K_{PFB} .

$$\begin{aligned}
K_{PFB} &= \left[\frac{\frac{1}{\lambda} - \frac{V_{DD}}{2}}{\frac{1}{\lambda} - \frac{V_{DD}}{2}} \right] \frac{\frac{1}{dv_o} - 1}{\frac{1}{dv_i} - 1} \cdot \frac{K_{PFF}}{A^2} \\
K_{PFB} &= K_{PFF} \cdot \frac{dv_i^3}{dv_o^3} \cdot \frac{1/\lambda - V_{DD}/2 - dv_o}{1/\lambda - V_{DD}/2 - dv_i}
\end{aligned}$$

(Note: According to our assumption, that is nMOS of the inverter is in triode region, $dv_o \neq 0$)

B. Third order term for short channel transistors

Nowadays the transistor channel length is reduced to as low size as makes the Early-voltage be a function of V_{GS} . To handle this situation is a little bit difficult because first we need to make some simulation (or measurement) to determine this characteristic of $\lambda(V_{GS})$. Unfortunately, this result can be handled only numerically, therefore we need some simplification to make the result be applicable. The simplification method is to split the curve to linear parts (figure below) where each linear curve can be characterized with its slope and its x axis

intercept point. ($f(x) = s(x - i)$ where $s (= \partial\lambda/\partial V_{GS})$ is the slope and i the x axis intercept point.) After linearization of Early-voltage we can write the following.

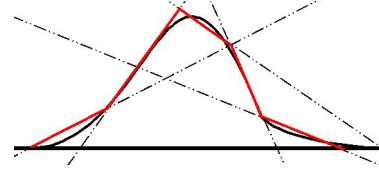


Fig. 4. Splitting to linear parts

$$\begin{aligned}
\lambda(V_{GS}) &= \lambda_1(V_{GS}) = s_1(V_{GS} - i_1), \quad 0 < V_{GS} < V_{GS1} \\
&\lambda_2(V_{GS}) = s_2(V_{GS} - i_2), \quad V_{GS1} < V_{GS} < V_{GS2} \\
&\dots \\
&\lambda_n(V_{GS}) = s_n(V_{GS} - i_n), \quad V_{GS_{n-1}} < V_{GS} < V_{GS_n}
\end{aligned}$$

With this method the f and g functions defined above modify as

$$\begin{aligned}
f(dv_i, dv_i^2, dv_i^3) &= \\
dv_i \left[\frac{K_P}{2} (-V_{DD} - 2V_{THP}) \left(1 - s_{ff} \left(-\frac{V_{DD}}{2} + dv_i - i_{ff} \right) \left(\frac{V_{DD}}{2} + dv_o \right) \right) - \right. \\
&\quad \left. K_N \left(\frac{V_{DD}}{2} - dv_o \right) \right] + \\
dv_i^2 \left[\frac{K_P}{2} \left(1 - s_{ff} \left(-\frac{V_{DD}}{2} + dv_i - i_{ff} \right) \left(\frac{V_{DD}}{2} + dv_o \right) \right) \right] \\
g(dv_o, dv_o^2, dv_o^3) &= \\
dv_o \left[K_P (-dv_i - V_{THP}) \left(1 - s_{fb} (-dv_o - dv_i - i_{fb}) \left(\frac{V_{DD}}{2} + dv_i \right) \right) \right] + \\
dv_o^2 \left[\frac{K_P}{2} \left(1 - s_{fb} (-dv_o - dv_i - i_{fb}) \left(\frac{V_{DD}}{2} + dv_i \right) \right) \right]
\end{aligned}$$

We use the same consideration as above.

$$C_{FF} + f(dv_i, dv_i^2, dv_i^3) = -A(C_{FB} + g(dv_o, dv_o^2, dv_o^3))$$

The connection between the third order terms is the following.

$$\begin{aligned}
dv_i^3 \left[\frac{K_{PFF}}{2} s_{ff} \left(\frac{V_{DD}}{2} + dv_o \right) \right] &= -A dv_o^3 \left[\frac{K_{PFB}}{2} s_{fb} \left(\frac{V_{DD}}{2} + dv_i \right) \right] \\
K_{PFF} s_{ff} \left(\frac{V_{DD}}{2} + dv_o \right) &= A^4 K_{PFB} s_{fb} \left(\frac{V_{DD}}{2} + dv_i \right) \\
K_{PFF} \frac{dv_i^4}{dv_o^4} \frac{s_{ff}}{s_{fb}} \frac{V_{DD}/2 + dv_o}{V_{DD}/2 + dv_i} &= K_{PFB}
\end{aligned}$$

Where

$$\begin{aligned}
s_{ff} &= \frac{\partial \lambda_{FF}}{\partial V_{GS}} \Big|_{V_{GS}=V_{DD}/2 \pm dv_{i_{max}}} \\
s_{fb} &= \frac{\partial \lambda_{FB}}{\partial V_{GS}} \Big|_{V_{GS}=V_{DD}/2 \pm dv_{o_{max}}}
\end{aligned}$$

For larger PAE we have to use minimum size FB transistors. Therefore the K_{NFB} is given (and depends on the applied technology). In this case for a predefined input and output maximum the K_{PFB} can be expressed as

$$\begin{aligned}
A^4 &= \frac{K_{PFF}}{K_{PFB}} \cdot \frac{s_{PFF}}{s_{PFB}} \cdot \frac{V_{DD}/2 + dv_o}{V_{DD}/2 + dv_i} = \frac{K_{NFF}}{K_{NFB}} \cdot \frac{s_{NFF}}{s_{NFB}} \cdot \frac{V_{DD}/2 + dv_o}{V_{DD}/2 + dv_i} \\
K_{PFB} &= \frac{K_{NFB}}{K_{NFF}} \cdot K_{PFF} \cdot \frac{s_{NFB}}{s_{NFF}} \cdot \frac{s_{PFF}}{s_{PFB}}
\end{aligned}$$

VII. HIGHEST LINEARITY

The comparison between third and second solution leads us to the next expression where the circuit reaches its highest linearity.

$$K_{P_{FB2}} = K_{P_{FB3}} \cdot \frac{s_{N_{FB}}}{s_{N_{FF}}} \cdot \frac{s_{P_{FF}}}{s_{P_{FB}}} = K_{P_{FB3}} \cdot S$$

The $K_{P_{FB2}}$ is the second and $K_{P_{FB3}}$ is the third order solution. From this connection it can be considered that with proper input and output voltages, where $S = 1$, both the second and the third order components can be eliminated. However from efficiency's point of view this is probably not optimal.

VIII. STABILITY

As we wrote in the introduction this topology is useful at low frequencies however due to its simplicity, this topology can be attractive at higher frequencies. Therefore, the designer has to know where are the poles and zeros of the system. The system contains two main poles. The dominant pole is at the output where the C_L with the parasitic capacitances of the basic stage and the FB and the large output resistance of the inverter set relatively low $1/(C_L + C_{par}) \cdot (r_{DSp} \times r_{DSn})$ value. The second pole (first non-dominant pole) is at the input due to the large input capacity of the basic state. Since the circuit has 180° phase shift the FB should shift additional 180° for an unstable state. With one pole it can not be reached.

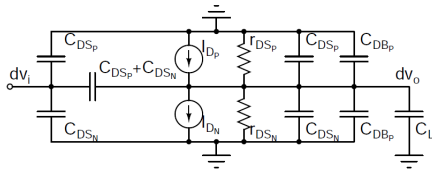


Fig. 5. Small signal equivalent

IX. IMPLEMENTATION

As a physical implementation this circuit is the part of transceiver which is designed with 0.13u HCMOS technology. In transceiver due to parasitic coupling the PA has significant effect to the phase noise of the VCO. A common method is to drive the PA at lower frequency than the VCO oscillates. Generally the divider circuit is series of flip-flops with a square-wave output. This raises the problem of efficiency. Namely a square-wave signal contains very high level harmonics are not necessary and filtered out by the environment very rapidly. In the next section a possible way will be presented to generate a sinusoid from square-wave.

A. Sine generator

Because of the periodic behavior of the signal we can make our calculation with Fourier-series in place of Fourier-transform. The series of square-wave is the following.

$$\sum_{k=0}^{\infty} \frac{\sin((2k+1)\omega t)}{2k+1}$$

Theoretically to get a sinusoid from a square-wave we do not have other than eliminate the parts with $k > 0$. The periodic behavior of the signal let us avoid large filters and make the transformation with the help of the next topology.

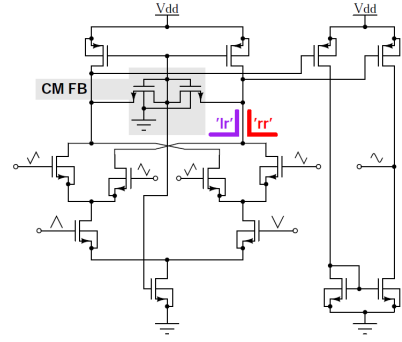


Fig. 6. Sine generator

The inputs of the circuit are shifted triangle-waves where the phase difference is $\pi/2$. These phases can be generated easily after division while the triangle form is created by a simple passive integrator.

As it can be seen the topology is very similar to a Gilbert-cell ("Inverted Gilbert-cell"). Let's focus on the right output branch. Basically the topology is a multiplier therefore we have to consider both the product and the sum of two triangle-wave with $\pi/2$ phase difference.

1) *Product*: It can be calculated the result of this product is the following where the accuracy depends on the accuracy of the triangle-wave.

$$\sum_{k=0}^{\infty} (-1)^k \frac{\sin((2k+1)\omega t)}{(2k+1)^2} \cdot \sum_{k=0}^{\infty} \frac{\cos((2k+1)\omega t)}{(2k+1)^2} \approx \frac{7}{18} \sin(2\omega t)$$

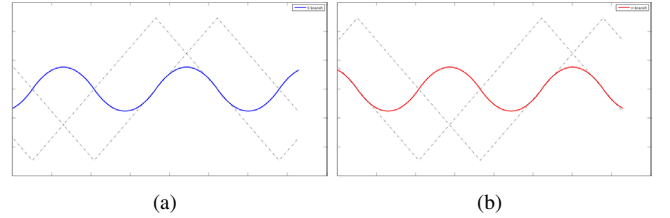


Fig. 7. Products in the right output branch

From the pictures can be seen the product of branch 'lr' has π phase shift respect to the product of branch 'rr'. So sum of these two branches does not contain second order harmonics.

2) *Sum*: Now we calculate the sum of the branches what is the main point of this circuit. The sum of the 'lr' branch is

$$\sum_{k=0}^{\infty} (-1)^k \frac{\sin((2k+1)\omega t)}{(2k+1)^2} + \sum_{k=0}^{\infty} (-1)^k \frac{\sin((2k+1)\omega t + (2k+1)\pi/2)}{(2k+1)^2} = \sqrt{2} \cdot \sum_{k=0}^{\infty} (-1)^k \frac{\sin((2k+1)\omega t + (2k+1)\pi/4)}{(2k+1)^2}$$

while of the 'rr' branch is

$$\sum_{k=0}^{\infty} (-1)^k \frac{\sin((2k+1)\omega t)}{(2k+1)^2} + \sum_{k=0}^{\infty} (-1)^k \frac{\sin((2k+1)\omega t - (2k+1)\pi/2)}{(2k+1)^2} = \sqrt{2} \cdot \sum_{k=0}^{\infty} (-1)^k \frac{\sin((2k+1)\omega t - (2k+1)\pi/4)}{(2k+1)^2}$$

Now we add these to part to get the final result of the right output branches.

$$\begin{aligned} & \sqrt{2} \cdot \sum_{k=0}^{\infty} (-1)^k \frac{\sin((2k+1)\omega t - (2k+1)\pi/4)}{(2k+1)^2} + \\ & \sqrt{2} \cdot \sum_{k=0}^{\infty} (-1)^k \frac{\sin((2k+1)\omega t + (2k+1)\pi/4)}{(2k+1)^2} = \\ & 2 \cdot \sum_{k=0}^{\infty} (-1)^k \frac{\sin((2k+1)\omega t + (2k+1)\pi)}{(2k+1)^2} \end{aligned}$$

It can be seen the sign of every second harmonic become positive again. So the final result is the following.

$$\sum_{k=0}^{\infty} \frac{\sin((2k+1)\omega t)}{2k+1} \Rightarrow \sum_{k=0}^{\infty} \frac{\sin((2k+1)\omega t)}{(2k+1)^2}$$

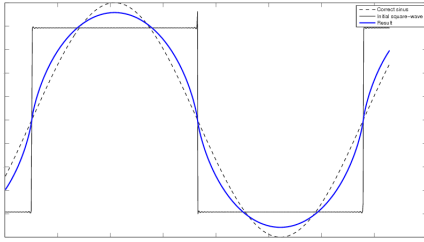


Fig. 8. The output of the sine generator

3) *Common mode feedback*: The behavior of this circuit has very significant effect to the behavior of the all circuit. Therefore a CM feedback is inevitable to reduce the bias changing caused by different corners.

B. The architecture

The final architecture of the project contains three main blocks. The PA, the sine generator and a Low Dropout regulator (LDO) which set the supply voltage of the PA as low as possible. The stability of the regulated voltage can be increased in two ways. With larger op.amp. open loop gain or larger output capacitance. Larger gain means larger consumption therefore we increased the capacitor size (90pF). The feedback was realized with diode connected MOS transistors because of their large resistance.

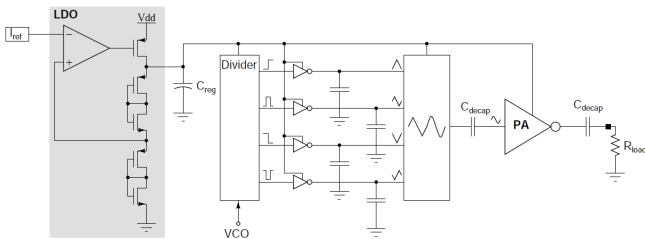


Fig. 9. Architecture

Between the sine generator and the PA there is a decoupling capacitor. This is because the bias current of the PA. The quite significant parasitics of this type of capacitor has two effects. On the one hand it decreases the PA input level but on the other hand it has a low pass filter effect on the harmonics. Before the PAD the large (50pF) DC decoupling does not let the DC flow across the load resistance.

The size of the total circuit is approximately $0.075mm^2$ and the main area (95%) is capacitance but the passives can remain onchip.

X. CONCLUSION FOR ITK

Due to the different corner requirements the design method was not easy and therefore very time consuming. I am at the very end of the layout designing but unfortunately my time has expired so I need to continue the work at home.

APPENDIX A SIMULATION RESULTS

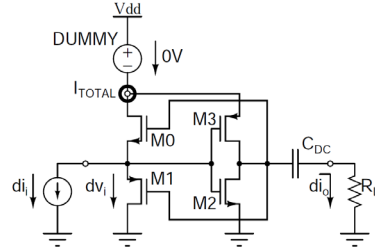


Fig. 10. Test schematic

(Note: C_{DC} must be large enough for that the $X_{C_{DC}}$ can be neglected respect to the R_L , the input current is a sinus and M2, M3 are low-leakage transistors for higher threshold voltage.)

For third order elimination we need to know how the Early-voltage changes respect to V_{GS} . In our calculations we have used this characteristics with 16 points resolution.

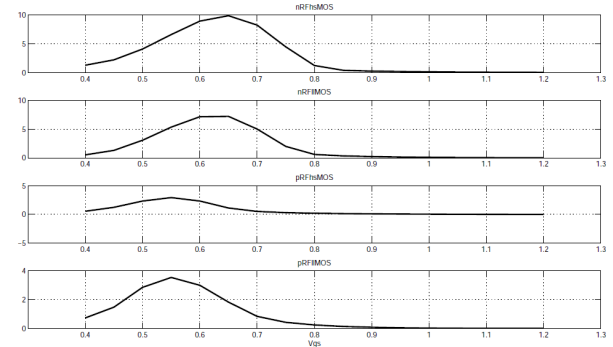


Fig. 11. Early-voltages respect to V_{GS}

REFERENCES

- [1] Patric Reynaert, Michiel Steyaert, RF Power Amplifiers for Mobile Communications, 2006
- [2] Scott D. Willingham, Integrated Video-Frequency Continuous-Time Filters, 1995
- [3] A. Rodríguez Vázquez, R. Domínguez-Castro, F. Medeiro and M. Delgado-Restituto, High Resolution CMOS Current Comparators and Piecewise-Linear Current Mode Circuits

Sensing in the terahertz frequency domain

Domonkos Gergelyi
(Supervisor: Dr. Péter Földesy)
gerdo@digitus.itk.ppke.hu

Abstract— There is an increasing demand for smart imaging devices working in the 0.3-3THz frequency domain. Although even the implementation of an efficient terahertz sensor is still a challenging task. A possible imaging method is proposed in this paper, what is based on compressive sensing. A digital micromirror device was used by the sampling for generating the aperture mask patterns. The measurement results of the test configuration with our basic sensor prototype are also provided.

Index Terms— DLP, compressed sensing, vision system on-chip, testbench

I. INTRODUCTION

In DLP (digital light processing) projectors, the image is created by microscopically small mirrors laid out in a matrix on a semiconductor chip, known as a Digital Micromirror Device (DMD) [2]. These micromirrors are flipped with a high frame rate to produce graytone and color projections with the usage of pulse width modulation and rotating color filters.

Using an off-the-shelf projector does not solve the high speed image, possible binary, projecting problem, which is required for vision system test and application development. Due to the fact, that those projectors accept grayscale/color information and modulate themselves the DMD array. On the other hand, development kits are available for DMD operation, with which the user gains full access to the projected view [3].

In this paper we highlight some of the possible applications and algorithm developments, where a DMD array is used to modulate the incoming light, and when the DLP projector is combined in an optical testbench with vision system on-chips:

- Enhancement of visual resolution through compressed sensing (CS).
- Scene projection.

The paper briefly describes the DLP technology (Section 2), a resolution enhancement method (Section 3), and a scene projection example (Section 4).

II. DLP TECHNOLOGY

In the DMD arrays, a mirror represents one or more pixels in the projected image. These mirrors can be repositioned rapidly to reflect light either through the lens or on to a dark area. Rapidly toggling the mirror between these two orientations (essentially on and off) produces grayscales, controlled by the ratio of on time to off time. The toggling rate of the DMD array is above 10,000 flips per second, depending on the array resolution and embedding system.

As the mirror size is in the range of 10-13 μm , there is no fundamental difficulty to project UV, near or short wave IR (NIR, SWIR) scenes.

In the development kit that is used in the experiments (DLP® Discovery™ 4000 Starter Kit Board .7 XGA with a LED OM optical head module) the highest black and white projection rate is above 22,000 frame per second [3].

III. RESOLUTION ENHANCEMENT OF VISION SOCS

The today emerging, high speed sensor processor arrays, like the EYE-Ris [1], the XENON family [4] or the VISCUBE [5] due to sensory technology difficulties or need for higher fill factor, they may require the vertical integration of sensory layer on top of the chip. This makes the per-unit area cost high. Therefore their resolution is limited. In some situations it will be useful to capture images at much lower speed, but in higher resolution. In this section we propose a possible (theoretical) technique, which helps to exploit adaptively the power of these cameras. We investigate in the MDM array framework, that it is possible to convert their high time resolution into spatial resolution by using compressed sensing [6].

We also use this framework to identify architecture and parameters for SWIR and terahertz imagers currently under development. The constraints were indicated by two dominant physical factors: the characteristics of the terahertz source, used for illumination and the properties of the sensor. Due to technological and economical considerations of production, the sensor is a small array of special detectors, what utilize thermopile technology. This fact makes the use of a more complex measurement configuration reasonable, until it has significant advantages over scanning pixel by pixel.

The purest CS methods are relatively sensible for noise, but there exist many solutions of the theoretical problem, whose model intrinsically handle it (by the addition of an explicit noise term.) However, due to the weak illumination sources, the ubiquitous terahertz noise and the limitations of the detector, the sensor SNR expected to be extremely low. Under such circumstances the inner structure of the image become distorted, unrecognizable, hence the restoration is impossible based on critically sampled data.

Without loss of generality, we work with only a part of the MDM array, and we suppose that the processing is done partially on array processors and their companion general purpose processor. With this setup the functionality of the method can be proved, reaching about four times bigger resolution as the sensor's native pixel number without losing the video real-time speed. The carefully chosen predefined mirror patterns guarantee that no pixel will be absent in the

integrals, moreover due to overlapping regions a significant gain can be reached by the weak signals.

In a broader view, the application example would be capable to find out the practical limits of the CS, by increase the resolution to the upper limit of the enhancement. This limit is theoretically the mirror arrays native resolution. On the other hand, the limits are not coming only from the array resolution, but from the reconstruction computation need as well. Other difficulties of the process are the proper design of the optics and the correction of the differences, caused by the displacement of the pixels relative to the DMD. Another critical point is to ensure an SNR low enough to span a dynamic range of 117-125dB at the sensor side to provide suitable data for 8 bit imaging when targeting the highest resolution. The third challenging task is to efficiently perform the linear optimization by the processor arrays.

It was examined to utilize - among others - the tools of the so called compressed sensing technique, which makes possible to restore images from under sampled data. From our point of view, not the possibility of under sampling was the most appealing property, but creating detailed pictures from multiple measurements on a sophisticated way, on which the poor SNR can be improved.

A. Brief summary of CS

The random sampling of a discrete signal can be written up as a multiplication:

$$\Phi x = y \quad (1)$$

where Φ is an M by N random matrix containing only 0 and 1. 'x' is the signal vector and 'y' is the measurement. Now, if one has the measurement data, y and the so called measurement matrix, Φ , then the signal can be reconstructed with a simple inverse multiplication:

$$x = \Phi^{-1}y \quad (2)$$

(Assuming that $M = N$ and Φ is invertible.) In the case of undersampling ($M < N$), the linear system above has infinite solutions. Multiplying with the pseudo-inverse is the solution of the system that minimizes the l_2 norm. If such a basis exists, in which the representation of the observed signal is sparse

$$\Psi a = x, \quad (3)$$

we can get a better result. Here Ψ is the matrix of the special basis and a is the sparse representation of the signal x . Exploiting this extra information one can minimize according to the l_0 norm, that is finding an a vector with the most zero components. Substituting (3) into (1), the problem can be reformulated as:

$$a^* = \arg \min_a \|a\|_0 \text{ s. t. } \Phi \Psi a = y. \quad (4)$$

Of course, obtaining a this way is computationally expensive, because it is a (NP complex) combinatorial problem. However there are several techniques, which offer more efficient, but still good approximations for that.

As a first approach l_0 norm can be replaced with the l_1 norm, hence the problem becomes a convex optimization. It can be solved by linear programming. However l_0 is not a vector norm in the usual mathematical sense and its equivalence with l_1 is proved only for practically useless cases. The so called basis pursuit algorithm (BP) is the most common tool for reconstruction according to the l_1 norm [6].

The second possibility is to convert the objective function into a continuous, differentiable one and do the minimization considering the l_0 norm through that. The sensitivity and the accuracy of the goal function are contradictory to its global convergence. Therefore successive approximation and parameterization is used. For instance, local minima can be avoided by scaling the smoothness of the conversion function. Hence the name of the studied algorithm: smoothed l_0 (sl_0) [7].

A third, similar solution is to reverse the problem and by fixing some of the parameters (for instance the sparsity of the vector), divide the solution space of the combinatorial problem into parts. After that, by carefully choosing and testing these partitions, the combinatorial search can be done on a reduced subset [8]. At this point one has the choice to solve this subproblem by brute force or with one of the methods mentioned above.

Those areas, where CS can provide us advance in overall efficiency, are limited considering our project. However, the computations of combinatorial problems - even in relatively small size - are impracticable. Hence it is worth to concern an architectural solution of the subproblem of CS.

The sl_0 algorithm is said to be noise tolerant and it is obvious that for a particular set of Gaussian noise, whose deviation falls into a narrow range, it performs better than the other algorithms. Despite of that, the simulations showed, it cannot function at lower sensor SNRs than 31-35dB, due to the intrinsic properties of CS.

B. Image reconstruction

Fig. 1. shows a possible imaging setup. If one projects an image to a single sensor using random patterns, the process can be described as in (1). This way the picture can be reconstructed from sufficient number of measurements (M). Using CS this number can be less than required by the native resolution of the DMD, what creates the masking patterns of the view. The most important thing according to single pixel imaging, that summed signals are detected, which fact improves the signal to noise ratio.

The image made by the computationally expensive CS has some advantageous properties. If an inner representation exists, that is sparse enough, then the acquisition time can be shortened significantly. To have some notion about the achievable relation between M and N, one can take (5) as a ground.

$$M = O(K * \log \frac{N}{K}) \quad (5)$$

Here K is the number of non zero components in the vector a .

The method filters efficiently the additive Gaussian noise as well, naturally, only under a given average energy. When the

gathered information is archived in the sparse representation, it needs much less storage. Another very promising property, that the signal in this form can be invariant to many transformations. This helps analysis and feature extraction; in fact, it can make any computation more effective, what involves the signal.

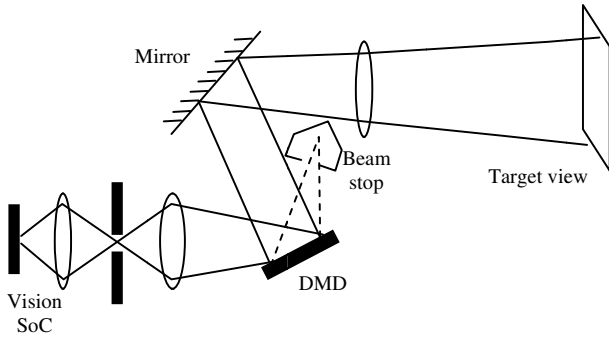


Figure 1. Schematic of the compressed sensing resolution enhancement setup.

C. The used method

The computational need increasing quadratically with the image size and the amount of data to be stored does so as well. Thus the algorithm can be computed practically only for pictures of resolution 16x16, 32x32 and 64x64. A table of comparison can be seen on Fig. 2.

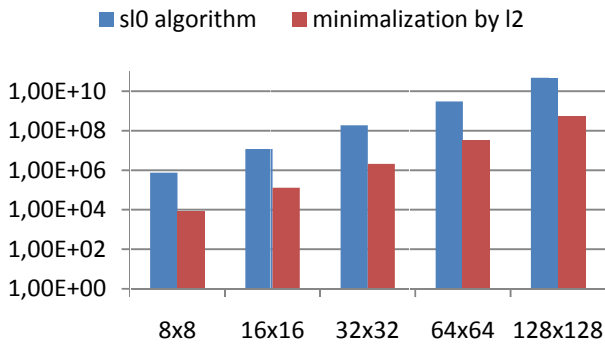


Figure 2. The number of the needed multiplications and additions in the two main method. The sI0 algorithm is compared with pure multiplication by the inverse of the measurement matrix (Φ) and the matrix of the basis (Ψ).

We decided to use the size 8x8 and 16x16 pixel as the basis of the data processing. Dividing the scene into small pieces, reduces the complexity tremendously, however it needs an array of detectors instead of a single one. Therefore the ratio of the real pixels and reconstructable pixels is 1/64 and 1/256, respectively. A necessary condition of a usable projection is the high fill factor of the detector. In this case each sensing element function as a single pixel camera, summing up the signal, what is reflected from a given part of the DMD. To help this phenomenon, the sensor can be situated a bit off the focus.

To examine the characteristics of the different methods, Matlab simulations were done. We concerned Gaussian noise with different deviations (σ) on the image and on the detector.

Independently from the used algorithm, due to the summation, the error caused by the former factor was relatively low.

To perform the simple l_2 minimalization, 16 bit multipliers and adders are adequate. Only some 20 bit wide accumulators have to be applied to produce pictures equal to those, which were created by double precision floating point representation. (It is true only for this 16 by 16 pixel data size.)

In the case of the sI0, the widest needed representation is 22-24 bit, considering the typical iteration number (10-15). In this case the decrease of the SNR on the resulting image is lower than 0.5dB, which is highly acceptable. (Since it is kind of successive approximation the gain of the method is limited by the used bitwidth, further steps have no effect on the quality.)

At noise levels resulting about 40 dB input image SNR ($\sigma=0.01$ relative to the maximal signal value) the outcome is still good: 17-20 dB. Since in this situation the advance of sI0 is about 3 dB, it is worth to cope with the high computational burden. However to achieve this result, the second noise term should be kept low. (This σ_2 must be smaller than 0.001.)

The XENON V3 [4] embodies our ideal test chip concerning high speed sensing and efficient processing. When the first specimen arrives the theory will be proved on that architecture. The reconstruction requires the 16 bit accuracy as mentioned above, however that device supports natively only the 8 bit arithmetic. These features are emulated by the hardware, which increase the computation time of the addition and the multiplication operation with a factor of 2 and 4, respectively. Despite of that, it is still capable to compute about 1-2 frame per second with sI0 or about 90-200 frame per second performing simple l_2 minimalization. Even these values can be easily surpassed when the reconstruction is done offline. In that case the bottleneck of the system is the DMD with its maximal 22000 frame per second refreshing rate.

D. Planned application field

As it was mentioned above, our main goal with this DMD setup was to build a test environment in the visible range, what can provide experimental data for creating our SWIR, THz imager under development. Measurements with a test sensor showed us that in the THz range the sensors SNR will be extremely low. That's why we had to examine the oversampling cases as well. Already from 10-30% additional measurement improvement can be observed. By evaluating the factor of necessary extra samples, one should take into account, that the advantage of sensing on a 16 times greater area reduces the need for oversampling with a significant constant factor.

IV. SCENE PROJECTION

The proposed setup can function as an optical, high speed test environment as well. Namely, with the help of the DLP projector, many type of a real-world environment could be simulated, which is indispensable for testing high speed sensor related phenomena. The schematic of the optical setup can be seen in Fig. 3.

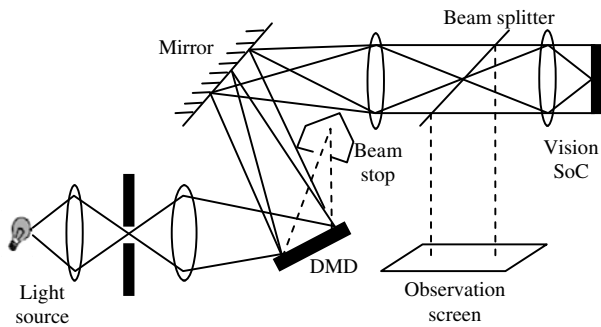


Figure 3. Schematic of the scene projection setup.

The synthetic or recorded high speed video flow can be projected real-time to the visual processor. The reason to provide optical input instead of electronic is multifold. It is worth to note, that the real-time speed can be several thousand frames per second, depending on the application. The optical input could provide orders of magnitude higher data throughput as the electronic in visual processors. In a mixed-signal implementation the limited data retention could seriously affect the operation when image sequence is processed and former results are required. The direct optical input – as the actual application possesses – during algorithm development provides a realistic situation from timing point of view. Hence, there is no need for taking into account the data retention and degradation issues that could otherwise mislead the development. The other straightforward advantage is the evaluation of the optical input as part of the application. The generated or recorded representation downloaded electronically is not necessarily the same to what the optical input provides – e.g. linearity, SNR. Considering a low-light condition or exotic sensor design (e.g. short wave IR), the application itself could be the sensory input preprocessing combined e.g. with high speed space-time signature analysis. Finally, in several cases the high speed field test is not trivial and does not accessible during application development (e.g. UAV collision detection and navigation, when the main task is to detect a fast approaching, distant object from as far as possible [5]). If the above mentioned conditions arise together, the high speed video flow projecting in visual or near/short wave IR region could be an attractive solution.

ACKNOWLEDGMENT

The work is supported by the Hungarian Scientific Research Fund - National Office for Research and Technology OTKA-NTKH CNK-77564 project.

REFERENCES

- [1] A. Rodríguez-Vázquez, R. Domínguez-Castro, F. Jiménez-Garrido, S. Morillas, A. García, C. Utrera, M. Dolores Pardo, J. Listan, and R. Romay, "A CMOS Vision System On-Chip with Multi-Core, Cellular Sensory-Processing Front-End", in Cellular Nanoscale Sensory Wave Computing, edited by C. Baatar, W. Porod and T. Roska, ISBN: 978-1-4419-1010-3, 2009
- [2] <http://www.ti.com>
- [3] <http://www.vialux.de>, DLP® Discovery™ Starter Kits

- [4] P. Földesy, Á. Zarándy, Cs. Rekeczky, and T. Roska „Configurable 3D integrated focal-plane sensor-processor array architecture”, *Int. J. Circuit Theory and Applications (CTA)*, pp: 573-588, 2008.
- [5] Földesy, Peter; Carmona-Galan, Ricardo; Zarandy, Akos; Rekeczky, Csaba; Rodriguez-Vazquez, Angel; Roska, Tamas, “3D multi-layer vision architecture for surveillance and reconnaissance applications”, *proceedings of European Conference on Circuit Theory and Design. ECCTD 2009*, pp.: 185 - 188, 23-27 August 2009, Antalya, Turkey.
- [6] D. Donoho, “Compressed sensing”, *IEEE Trans. on Information Theory*, Vol. 52, Issue 4, pp.: 1289-1306, April 2006.
- [7] G. Hosein Mohimani, M. Babaie-Zadeh, C. Jutten, “A fast approach for overcomplete sparse decomposition based on smoothed l_0 norm”, *IEEE Transactions on Signal Processing*, Vol. 57, No. 1, Jan. 2009, pp. 289-301.
- [8] L. Mancera, J. Portilla, “ L_0 -norm-based Sparse Representation through Alternate Projections”, *13th International Conference on Image Processing (ICIP)*, October 2006.

Energy balancing in wireless sensorial network using discrete energies

Balázs Karlócai

(Supervisor: Dr. János Levendovszky)
karlocai.balazs@itk.ppke.hu

Abstract—In this paper we have introduced 3 similar algorithm, having low complexity. Although the methods are suboptimal, but better performed compared to the mostly-used current Leach protocol. The basis are the same for all the methods: the physical implementation of the wireless sensor nodes are not capable to transmit signal with any gain in the range, but discrete values depending on the implementation.

The new idea is based on this fact. In the beginning the transmission gain is set to the available minimum, and later on, we increase it smoothly until the transmission reaches the desired probability [1].

The optimal transmission energies are derived which guarantee that the packets are received by the Base Station (BS) with a given probability subject to achieving the longest possible lifespan [2]. The algorithm is based on a single iteration loop, which can be with a single Dijkstra or Bellmann-Ford algorithm. The new results have been tested by extensive simulations, which demonstrated that the lifespan of WSN could significantly be increased by the new protocols.

I. INTRODUCTION

Due to the recent advances in electronics and wireless communication, the development of low-cost, low-power, multifunctional sensors have received increasing attention [3], [4], [5]. These sensors are compact in size and besides sensing they also have some limited signal processing and communication capabilities. However, these limitations in size and energy make WSNs different from other wireless and ad-hoc networks [6], [7].

As a result, new protocols must be developed with special focus on energy balancing in order to increase the lifetime of the network which is crucial in case applications, where recharging of the nodes is out of reach (e.g. military field observations, living habitat monitoring ... etc., for more details see [8]).

Former researches were spotted to find a good path and gain in the transmission in continuous scale [9], [10], [11]. It is not realistic. The real options for the gain settings are discrete values (Fig. 1). The difference is huge: the problem can be bigger complexity, and the nodes are unable to set the proper value, but the available one.

Having this ideas a new algorithm has been planned, turning the mentioned disadvantages to advantages. In the routing algorithm we are taking only the possible values, so we have an opportunity to decrease the complexity, and in the other hand the node will surely able to set the selected value.

We hope, if the network is distributed normally, and there are no densely located areas, the imagined algorithm could

perform well.

II. MODEL

The problem, and the new idea will be described as follows. First of all, we have to characterize the problem itself. Our goal is to get an optimal route in a graph having the nodes. The vertices of the graph will be denoted as v . The route, R , will be described having a V vector, which contains the list of the v -s, and the belonging energies, E , in the path. The distances are stored as well.

$$R(V, E, d_{\in V}) \quad (1)$$

Our task is to find the optimal R_{opt} .

In our model, we choose an environment, where the success rate of the transmission can be described as a function (Ψ), with the following parameters: distance (d) and the gain (g). Later on we will have an opportunity to choose any propagation model for the Ψ function. For example in our simulation, we used the widely applied Rayleigh Fading model [12].

$$P_{u,v} = \Psi(d_{u,v}, g) \quad (2)$$

The base criteria for the problem is to hold the transmission rate above a specified parameter (ϵ). Therefore we can describe it with a product:

$$\prod_{u,v} \Psi(d_{u,v}, g) \geq 1 - \epsilon \quad (3)$$

- Tiny devices
- Constrained processor
 - 4Mhz, 8-bit operations
- Little memory
 - 128KB Flash, 4KB EEPROM and 4KB RAM
- Low bit rate communication
 - 40Kbits/second
- Short transmission range
 - ~30 meters (100 feet)
- Low energy
 - Running on batteries

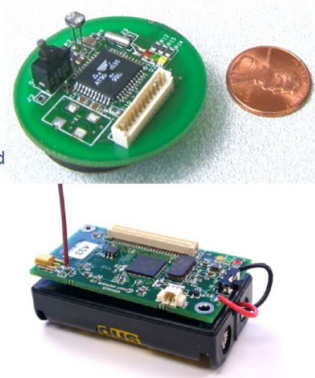


Fig. 1. mica node having 16 discrete gain level

To keep the equations in a closed form, we put the condition into a logarithmic scale. Therefore the product became a sum.

$$\sum_{(u,v) \in R} -\lg(\Psi(d_{u,v}, g)) \leq -\lg(1 - \epsilon) \quad (4)$$

In the current model, we only use discrete gains (G_1, G_2, \dots, G_N) , so the problems goal-function is to minimize the system-wide energy usage.

$$\min(g), g \in \{G_1, G_2, \dots, G_N\} \quad (5)$$

Based on the new idea, we separate the task into iteration steps. In the beginning, all the energies are set to the minimum. We find an optimal route as follows:

$$g_0 \rightarrow R_0 : \min \sum_{(u,v) \in R} -\lg(\Psi(d_{u,v}, g)) \quad (6)$$

This can be done with a Dijkstra or Bellmann-Ford algorithm in polynomial time. The second step of the iteration is to check if the results are eligible for the condition, and enhance the parameters.

$$\sum_{(u,v) \in R} -\lg(\Psi(d_{u,v}, g)) \leq -\lg(1 - \epsilon) \quad (7)$$

If it fulfills

$$g_1 := g_0 - \Delta \quad (8)$$

If not

$$g_1 := g_0 + \Delta \quad (9)$$

This method increase the energy level, until the criteria is not fulfilled. Therefore we get a fast, and cost-efficient algorithm.

A. Enhanced iteration algorithm

Although the algorithm described above is a fine, and fast algorithm, and in some circumstances it brings really good results, it can not take care, if the nodes are not equally distributed.

On the other hand, it ignores if the nodes close to the sink, became bottle-neck-node, i.e. nodes which are part of many routes, so the load is much higher than the others. We modify the method a little, to consider the energy levels as well. The idea is, to increase the highest charged node's energy level, until the criteria is fulfilled, or the energy level differ to much from the others.

Let's defines an n integer, to limit the energy level of a node, from the others. The base energy level can be increased only if all the nodes has been increased individually. Note that, the $n = 0$ is the same algorithm, described before.

The description of the method can be observed as a state-diagram in fig. 2.

The method will be referred as "method G" in Simulation section.

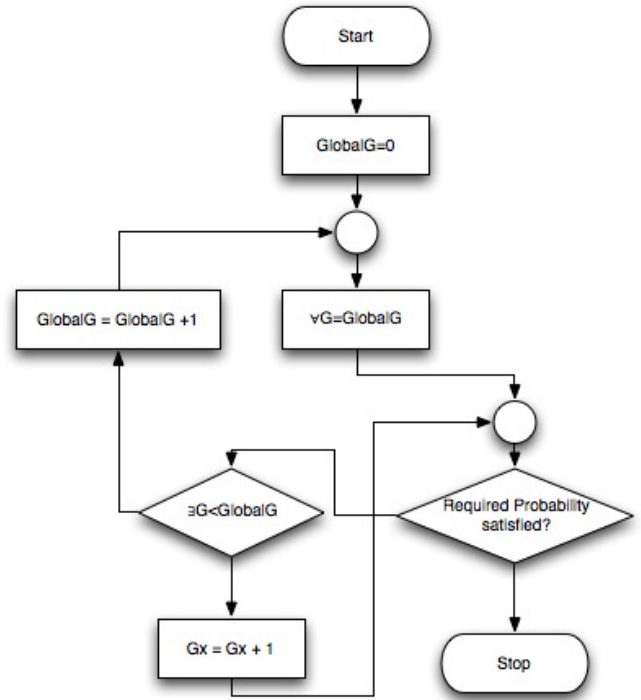


Fig. 2. Enhanced iteration algorithm status diagram, during the packet transmission

B. Enhanced iteration algorithm – Low energy node exclusion

The second modification is an energy level consideration. A mean of the method is to exclude the nodes from the transmission, if the charge is low. This nodes are activated only if they are to send packets, not during the transmission. With this method, we can keep the network alive much longer.

Using this method we can keep the weak nodes safe. 3

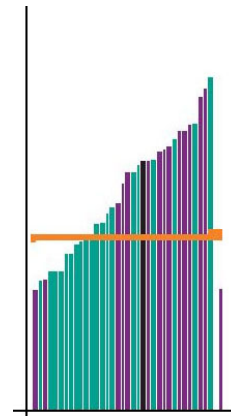


Fig. 3. The method puts a threshold, and denies the low energy nodes to be part of the transmission

The method will be referred as "method min-G" in Simulation section.

III. SIMULATION

In the simulation, we compared the well-known Leach protocol with the described ones. As I mentioned before, I used the Rayleigh fading model, with the following parameters.

- $g_0=5$;
- $\theta=0.1$;
- $E_{noise}=0.1$;
- $\alpha=2$;

The network has been prepared as follows: I have put the base station (sink) to the middle of an area of 100x100. The nodes has been spread randomly. All the energy are 20.000, and the required probability (P_{req}) 0.9.

A. Running the simulation

The simulation has been completed as follows: I've generated 100 random different network. All the networks has been copied as many times, as many different algorithm I have tested (method G, method min-G, leach). The networks were running simultaneous, and the stopping criteria was the first node exhaustion. I've took care to be sure, the same nodes were generated the packets, so the environment were the same.

B. Simulation environment

Due to that, the simulation was not evaluated in the aspect of running time, but the number of transmitted packets, the environment did not influenced the results. Simulation program was programmed in Java.

C. Results

In the following figure, the transmitted packets can be seen, until it reached the criteria.

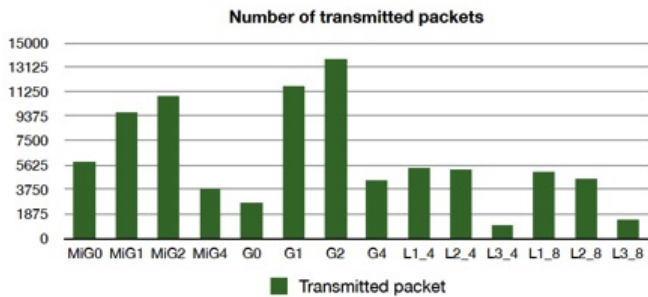


Fig. 4. transmitted packets (x1000) until the exhaustion, with the different algorithms

Can be seen, that the fix-G algorithms are much better. Although if we choose a too big number, the performance decreases.

In this figure, the MinG shows its majority, to keep the energy-load well distributed, that's why the leftover energy is the less with this method. But compared to figure 4., it shows, the main goal is not achieved the best way through this method.

Let's discover that, not the network having the less energy at the end-time performs the best. Having a look at G2, it can

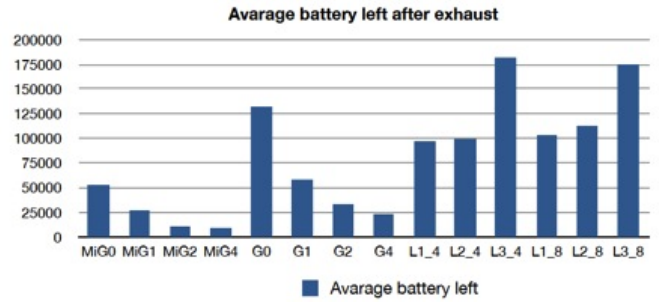


Fig. 5. the energy in the network after the exhaustion

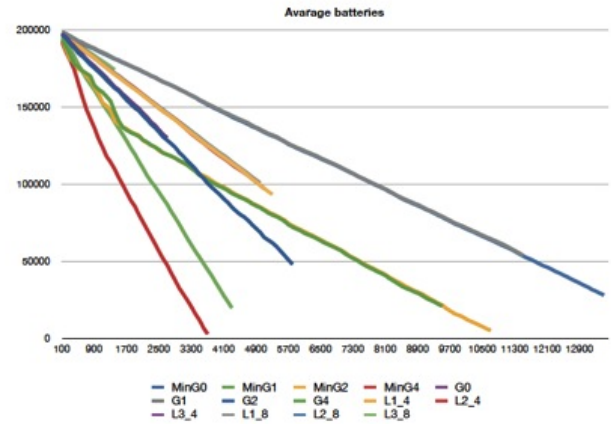


Fig. 6. mean energy values in every network

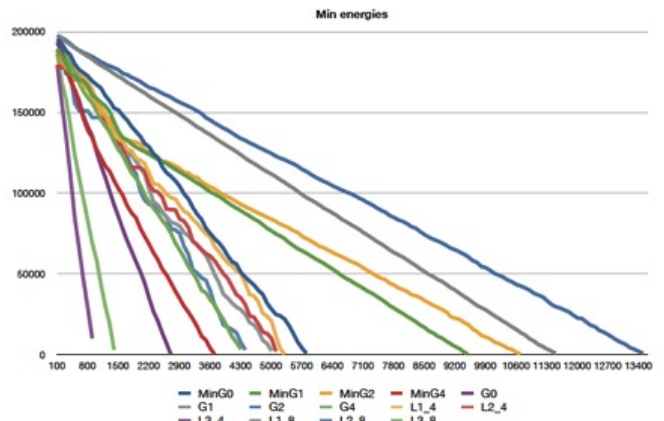


Fig. 7. the lowest energy level

be see, that the energy left is much more that MinG4, although the overall performance is much better.

The history of the most, and the less energy node can be seen in 7, and in 8. The MinG algorithm observed the lowest energy level, and tried to avoid them from the communication, so the gradient getting better after a while. On the other hand we can see if the highest energy node is loaded, the network still have some resource.

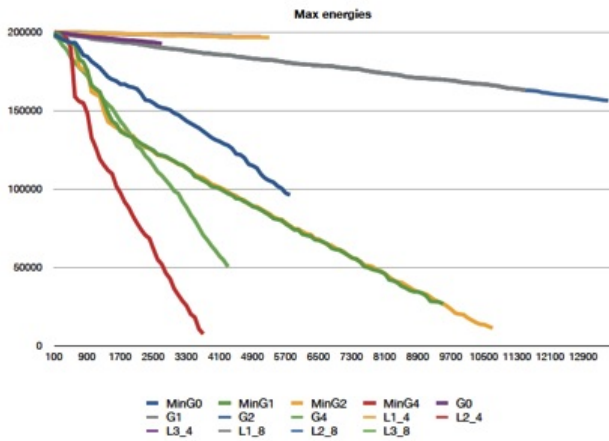


Fig. 8. the highest energy level

IV. CONCLUSION

In this paper, novel energy balancing packet forwarding methods have been developed to maximize the lifespan of WSNs and to ensure reliable packet transfer at the same time. We have optimized the transmission energies of the nodes depending on the source of the packet in order to minimize the energy consumption of the bottleneck node with discrete energies, subject to satisfying a given the reliability constraint. The algorithm has been developed to use the discrete energy values as an iteration base. The algorithm runs in P time complexity. The given method is able to find a good route and the energy optimization strategy is quite good as well. The underlying protocol optimization was reduced to a Dijkstra or Bellman-Ford algorithm, which can be solved fast and effective. The performance of the protocol has been compared with a well known route find algorithm – the Leach algorithm. The results showed us that the new method has a good efficiency. The very same network exhausted 2-3 times later with the new algorithm than with the former Leach. We have made a long test with 100 random generated network, and we saw that the new algorithm perform in average 1.8 times better than the Leach.

At the same simulation we saw that some of the nodes still kept some energy, so there are also some opportunities to continue the research in this direction.

REFERENCES

- [1] V. Li and J. Silvester, "Performance analysis of networks with unreliable components," *Communications, IEEE Transactions on [legacy, pre-1988]*, vol. 32, no. 10, pp. 1105–1110, 1984.
- [2] T. Ozgur and I. Tan, "Power efficient data gathering and aggregation in wireless sensor networks," *SIGMOD Record*, vol. 32, no. 4, pp. 66–71, 2003.
- [3] J. Leventovszky, A. Olah, A. Bojarszky, and B. Karlocai, "Energy balancing by combinatorial optimization for wireless sensor networks," in *IWDN07 Conference pp. 1–6.*, September 2007.
- [4] C. Kumar and S. Kumar, "Sensor networks: evolution, opportunities, and challenges," *Proceedings of the IEEE*, vol. 91, no. 8, pp. 1247–1256, 2003.

- [5] D. Puccinelli and M. Haenggi, "Wireless sensor networks: applications and challenges of ubiquitous sensing," *IEEE Circuits and Systems Magazine*, vol. 5, no. 3, pp. 19–31, 2005.
- [6] A. Goldsmith and S. Wicker, "Design challenges for energy-constrained ad hoc wireless networks," *IEEE wireless communications*, vol. 9, no. 4, pp. 8–27, 2002.
- [7] R. YanYo and R. Estrin, "Geographical and energy aware routing: A recursive data dissemination protocol for wireless sensor networks," *UCLA Computer Science Department Technical Report UCLAICSD-TR-OI-0023*, 2001.
- [8] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, "Wireless sensor networks for habitat monitoring," in *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*. ACM, 2002, pp. 88–97.
- [9] W. Heinzelman, A. Chandrakasan, H. Balakrishnan, and C. MIT, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Transactions on wireless communications*, vol. 1, no. 4, pp. 660–670, 2002.
- [10] —, "Energy-efficient communication protocol for wireless microsensor networks," in *System Sciences, 2000. Proceedings of the 33rd Annual Hawaii International Conference on*, 2000, p. 10.
- [11] W. Heinzelman, A. Sinha, A. Wang, A. Chandrakasan, and C. MIT, "Energy-scalable algorithms and protocols for wireless microsensor networks," in *2000 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2000. ICASSP'00. Proceedings*, vol. 6, 2000.
- [12] M. Haenggi, "On routing in random rayleigh fading networks," *IEEE Transactions on Wireless Communications*, vol. 4, no. 4, pp. 1553–1562, 2005.

Cooperative Communication Algorithms in Wireless Systems

Gergely Treplán

(Supervisor: Dr. János Levendovszky)

trege@digitus.itk.ppke.hu

Abstract—In this paper, a reliable cooperative multipath routing algorithm is proposed for data forwarding in wireless sensor networks (WSNs). In this algorithm, data packets are forwarded towards the base station (BS) through a number of paths, using a set of relay nodes. In addition, the Rayleigh fading model is used to calculate the evaluation metric of links. Here, the quality of reliability is guaranteed by selecting optimal relay set with which the probability of correct packet reception at the BS will exceed a predefined threshold. Therefore, the proposed scheme ensures reliable packet transmission to the BS . Furthermore, in the proposed algorithm, energy efficiency is achieved by energy balancing (i.e. minimizing the energy consumption of the bottleneck node of the routing path) at the same time. This work also demonstrates that the proposed algorithm outperforms existing algorithms in extending longevity of the network, with respect to the quality of reliability. Given this, the obtained results make possible reliable path selection with minimum energy consumption in real time.

Index Terms—wireless sensor networks, reliability, cooperative routing, Rayleigh fading model, energy balancing

I. INTRODUCTION

Due to the recent advances in electronics and wireless communication, the development of low-cost, low-energy, multifunctional sensors have received increasing attention [3]. These sensors are compact in size and besides sensing they also have some limited signal processing and communication capabilities. However, these limitations in size and energy make the wireless sensor networks (WSNs) different from other wireless and ad-hoc networks [2]. As a result, new data packet transmission methods must be developed with special focus on energy effectiveness in order to increase the lifetime of the network which is crucial in case applications, where recharging of the nodes is out of reach (e.g. military field observations, living habitat monitoring etc., for more details see [8]).

Although a number of methods has been developed for energy aware data packet transmission in WSNs, such as destination-sequenced distance-vector (DSDV) routing, dynamic source routing (DSR), and ad hoc on-demand distance vector (AODV) routing, much of the research works is based on idealized assumptions about the wireless channel characteristics [7]. That is wireless communication can be perfect in term of packet loss within a circular radio range. However, several recent studies have convinced researchers that there is a need to replace this idealistic channel model with a more realistic one [10].

Against this background, using the Rayleigh fading model

[5] for wireless communication, this paper addresses reliable packet transmission in WSN when packets are to be received on the base station (BS) with a given reliability in terms of keeping the transmission error probability under a given threshold. In realistic communication channel models, the success of every individual packet transmission depends on the distance and the power of transmission, the probability of correct reception will diminish exponentially with respect to the number hops, in the case of multi-hop packet transfers. In this paper, a cooperative multipath approach is proposed for data packet routing which achieves optimal energy balancing (i.e. it minimizes the energy consumption of the bottleneck node of the network), with respect to the constraint of guaranteeing reliable packet transfer to the BS . In particular, in order to maximize the probability of successful delivery (i.e. reliability), a multipath routing technique is used. That is, each sensor node multicasts data to a set of relay nodes, which then independently forward each copy of the packet to the BS . The advantage of multipath routing is that, the reliability of the network does not depend on single node failures, which makes the network more robust. Given this, the main concern is to derive the appropriate transmission energies and the appropriate number for relay nodes needed to achieve a given reliability and to maximize the lifespan at the same time. To achieve this, first the energy output vector is optimized, if the set of cooperative nodes is given in order to keep the optimal energy balancing subject to the required reliability parameter. Second, an algorithm is devised which chooses the optimal cooperation set in polynomial time with respect to the bottleneck node. Finally the achieved lifespan of the proposed algorithm is compared to the longevity of traditional protocols by extensive simulations.

II. RELATED WORK

To date, cooperative routing techniques in WSNs can be classified into two groups, namely: (i) flat routing; and (ii) hierarchical routing. In the former, each node can send information to any other nodes within its communication range. On the other hand, hierarchical routing forms a hierarchical structure, so that each node can send data to those who are in a higher position in the hierarchy. Given this, related work can be discussed as follows.

From the side of reliable flat routing in WSNs, many research works have been published recently, such as directed diffusion (DD), rumor routing, and SPIN [6]. In these approaches,

one must choose routing paths such that the occurrences of packet loss on those paths are minimized. In these methods the possible forwarding nodes are carefully evaluated and the node of a higher probability of delivery is then selected as a forwarding node. However, the applied evaluation metrics vary in different approaches. For instance, in GeRaF [11] the geographic distance and a loss-aware metric in ETX [4] was used. However, these methods use a simplified wireless communication model, which, in many cases, is not sufficient to model the probability of data loss of the network. More recently, Zamalloa *et al.* proposed a position-based routing method using metrics similar to ETX [10]. Furthermore, in [9], the authors have proposed a flat routing algorithm, called BERA, that aims to maintain energy balancing, while the quality of reliability is satisfied. Their work can be seen as the closest to this work from the topic of reliable cooperative routing, since they also used a generic lossy link model instead of idealized simplified channel models. The aforementioned algorithms, however, do not exploit the advantage of multipath routing, and thus, may fail in environments with highly lossy radio links, or node failures.

On the other hand, a number of proposed methods in the topic of hierarchical cooperative routing in WSNs has also been proposed, such as the low energy adaptive clustering hierarchy (LEACH) [2].

In general, these methods make good attempts to try to balance the energy consumption by electing the clusterheads (CHs), each of which is responsible for relaying the data from a subset of nodes back to the *BS* in an intelligent way. These methods, however, do not take data loss into account, and thus, are not suitable for reliable routing.

III. THE MODEL

In this model, a WSN of N nodes with a single *BS* is considered. To forward data towards the *BS*, each node uses a set of relay nodes for relaying data. Here, the focus is on the case when a single node wants to deliver its packets. Hereafter, for the sake of simplicity, that node is referred to as the source node, and is denoted it with S . Given this, in this model, the source node broadcasts its packets to several relay nodes (i.e. the source has to send that packet *once* to multiple relay nodes at the same time), which then forward the copies of the packet *directly* to the *BS*. This model can be regarded as an extension of the hierarchical cooperative routing protocols. In particular, the relay nodes can be seen as nodes with higher positions in the hierarchy, and thus, the others have to forward data through them. On the other hand, in the proposed approach, the set of relay nodes are not fixed, that is, each transmitting node can use a different optimal set of relay nodes, in order to forward data to the *BS*. The main idea of this protocol is that, instead of using unicast transmission, where data sending needs large transmission power in order reach the reliability quality of service (QoS), it is more efficient for the node to multicast data with smaller power consumption, while the reliability QoS is still maintained, since there is a higher chance that at least one copy of the packet will arrive to the *BS*.

According to the Rayleigh fading model, the energy needed for transmitting the packet to distance d with the probability of correct reception P_r is given as

$$P^{(r)} = \exp \left\{ \frac{-d^\alpha \Theta \sigma_Z^2}{g} \right\} \quad (1)$$

where Θ is the modulation constant, σ_Z^2 denotes the energy of noise, and α is the propagation coefficient (its value is typically between 2 and 4). One must note that equation 1 connects the reliability of packet transfer P_r over distance d with the required energy g . For the sake of notational simplicity this relationship will be denoted by $P_r = \Psi(g)$. Furthermore, when need there is a transmission by a single hop packet transfer between two nodes i and j in the chain, then the corresponding reliability is $P_{ij}^{(r)} = \Psi(G_{i,j})$, where $G_{i,j}$ denotes the transmission energy on node i .

In this model, the source node transmits the data packet with certain G_s energy. According to equation 1, if the relay node R_i is in the receive mode, it can receive the packet successfully with the following probability:

$$P_{S,R_i} = \exp \left\{ \frac{-d_{S,R_i}^\alpha \Theta \sigma_Z^2}{G_S} \right\} \quad (2)$$

where d_{S,R_i} is the distance between the source node S and the relay node R_i . Each relay node then forwards the received packet towards the *BS* with G_{R_i} energy. The probability that the transmission of relay node R_i is successful can be calculated as the following:

$$P_{R_i,BS} = \exp \left\{ \frac{-d_{R_i,BS}^\alpha \Theta \sigma_Z^2}{G_{R_i}} \right\} \quad (3)$$

where $d_{R_i,BS}$ is the distance between the relay node R_i and the *BS*. In the model *BS* can receive messages from all of the sender nodes with a certain probability. Hence, the probability that the packet arrives successfully to the *BS* is:

$$P_{\text{success}} = 1 - (1 - P_{S,BS}) \prod_{i=1}^K (1 - P_{S,R_i} P_{R_i,BS}) \quad (4)$$

where K is the number of relay nodes used in this data delivery.

First, suppose that the set of relay nodes is given a priori for source node S . This set is denoted with $\mathbf{R} = \{R_1, R_2, \dots, R_K\}$. The energy required by a packet transfer is described by the set of the transmission energies with $\mathfrak{S}_{\mathbf{R}} = \{G_S, G_{R_1}, G_{R_2}, \dots, G_{R_K}\}$. In addition, let denote the energy level of each node v (both relay and source nodes) before the transfer with c_v . Given this, the objective is to find the energies $\mathfrak{S}_{\mathbf{R}}^{\text{opt}} = \{G_S^{\text{opt}}, G_{R_1}^{\text{opt}}, G_{R_2}^{\text{opt}}, \dots, G_{R_K}^{\text{opt}}\}$, that achieve optimal energy balancing; that is, it *maximizes the residual energy of the bottleneck sensor node in the transfer of the packet toward the BS*. The formulation of the problem can be described as follows:

$$\mathfrak{S}_{\mathbf{R}}^{\text{opt}} : \max_{\mathfrak{S}_{\mathbf{R}}} \left\{ \min_{v \in \mathbf{R} \cup \{S\}} (c_v - G_v) \right\} \quad (5)$$

That is, it is necessary to determine the optimal energy consumption values that maximizes the residual energy level of the bottleneck node (i.e. the node that has the lowest energy level after data transmission). However, it also has to be guaranteed that the packets arrive at the BS with a given reliability $(1 - \varepsilon)$; that is:

$$P_{\text{success}} \geq (1 - \varepsilon) \quad (6)$$

where P_{success} is defined in equation 4.

Now, since the set of relay nodes is typically not given for source node S , the optimal set of relay nodes has to be determined as well. Given this, the second objective is to determine the optimal set of relay nodes as well. That is, the goal can be formulated as follows:

$$\mathbf{R}^{\text{opt}} : \max_{\mathfrak{S}_{\mathbf{R}}^{\text{opt}}} \left\{ \min_{v \in \mathbf{R} \cup \{S\}} (c_v - G_v^{\text{opt}}) \right\} \quad (7)$$

In so doing, the case of fixed relay set will be studied, then an algorithm that determines the optimal relay set will be proposed in the subsequent sections. In particular, the analysis of the case of fixed relay set with fixed source transmission energy will be described.

IV. ROUTING WITH FIXED RELAY SET AND GIVEN SOURCE TRANSMISSION ENERGY

In this section, it is assumed that both \mathbf{R} and G_S are already given. Thus, the goal is to find the optimal energies $\mathfrak{S}_{\mathbf{R}}^{\text{opt}} = \{G_S, G_{R_1}^{\text{opt}}, G_{R_2}^{\text{opt}}, \dots, G_{R_K}^{\text{opt}}\}$ which maximize the residual energy of the bottleneck relay sensor node subject to fulfilling the reliability criterion. Given this, one has the following:

Theorem 1: *Assuming that G_S is already given, under the reliability parameter $(1 - \varepsilon)$, the value of the residual energy level of the bottleneck relay node reaches the maximum when all the residual energy levels are the same.*

V. ROUTING WITH FIXED RELAY SET

This section extends the aforementioned optimization problem as follows. Here, the source node S is allowed to modify its transmission energy. Thus, beside finding the optimal value for each G_{R_i} , the optimal value of G_S has to be determined as well. According to theorem 1, the maximal residual energy level of the bottleneck relay node is achieved when the residual energies are equal at the relay nodes. This energy level depends on the value of G_S . Thus, hereafter this energy level is referred to as $\Phi(G_S)$. Given this, one can state the following:

Lemma 2: *The function $\Phi(G_S)$ is strictly monotonously increasing.*

In the general case, the source node is also taken into account. Thus, the residual energy at the source node is $c_S - G_S$. Since $\Phi(G_S)$ is increasing, and $c_S - G_S$ is strictly decreasing, as G_S is increased, it can be proven that the maximum value of the general bottleneck node's residual energy (including the source and all the relay nodes) is achieved only if $c_S - G_S = \Phi(G_S)$. As the result, one can state the following:

Theorem 2: *Under the reliability parameter $(1 - \varepsilon)$, the maximal value of the residual energy level of the general*

bottleneck node is achieved when all the residual energy levels at the source node and the relay nodes are equal to each other. Therefore, fast and computationally efficient algorithms can be used, such as the well known Newton-Raphson method, to determine the optimal value of A . Note that since A must be non-negative, otherwise there is no solution for the reliability routing problem.

VI. SELECTING THE OPTIMAL RELAY SET

So far in this paper it was assumed that the set of relay nodes is already given. In this section, the selection of the optimal set of relay nodes participating in the cooperation is investigated in more detail. Let \mathbf{R}^{opt} denote the optimal set of relay nodes, as defined in equation 7.

Using theorem 1, one can assume that for each node v of the optimal set, including the source node as well, the optimal residual energy level is $c_v - G_v = A$.

Against this background, an algorithm is proposed which determines the optimal set of relay nodes. In so doing, first a number of notation is introduced as follows. Given a set of relay nodes \mathbf{R} , let $A_{\mathbf{R}}$ denote the optimal residual energy level of nodes within \mathbf{R} , including source node S . That is, $A_{\mathbf{R}} = c_v - G_v^{\text{opt}}$ for $\forall v \in \mathbf{R} \cup \{S\}$, which can be calculated by using theorem 2. Now, consider the following algorithm:

- 1) Step 1: Let $\mathbf{R}_1 = \{S\}$. Given this, let $A_{\mathbf{R}_1}$ denote the optimal residual energy of source node S , when none of the relay nodes is used for data forwarding. Let $k = 1$. GOTO step 2.
- 2) Step 2: For each value of k , if $\exists c_v > A_{\mathbf{R}_k}$, then GOTO step 3, otherwise GOTO step 4.
- 3) Step 3: Let i denote the node that satisfies the following: $i = \arg \max_v \{c_v | c_v > A_{\mathbf{R}_k}\}$; that is, i has the highest energy level among nodes which have more energy than $A_{\mathbf{R}_k}$. Thus, $\mathbf{R}_{k+1} = \mathbf{R}_k \cup \{i\}$, and $k = k + 1$. GOTO step 2.
- 4) Step 4: If $A_{\mathbf{R}} > 0$ then it is the optimal solution, otherwise there is no solution for the problem.

Theorem 3: *The aforementioned algorithm converges to the optimal set of relay nodes, which is the optimal in terms of maximizing equation 7, with respect to the constraint given in equation 6.*

VII. PERFORMANCE EVALUATION

Within the previous sections, it has been proved that the proposed algorithm is optimal in the sense of balancing the energy consumption in the network. However, it is not clear that this approach is whether efficient, compared to other existing algorithms, such as LEACH, or BERA. Given this, this section demonstrates that by using multipath cooperative routing, one can achieve a better longevity of the network, compared to that of networks using LEACH for data forwarding. In so doing, first the parameter settings of the simulation environment will be described. Following this, since LEACH is originally not suitable for reliable routing, it will be discussed how to modify it such that reliability can be still maintained, in order to make the comparison between LEACH and the proposed approach

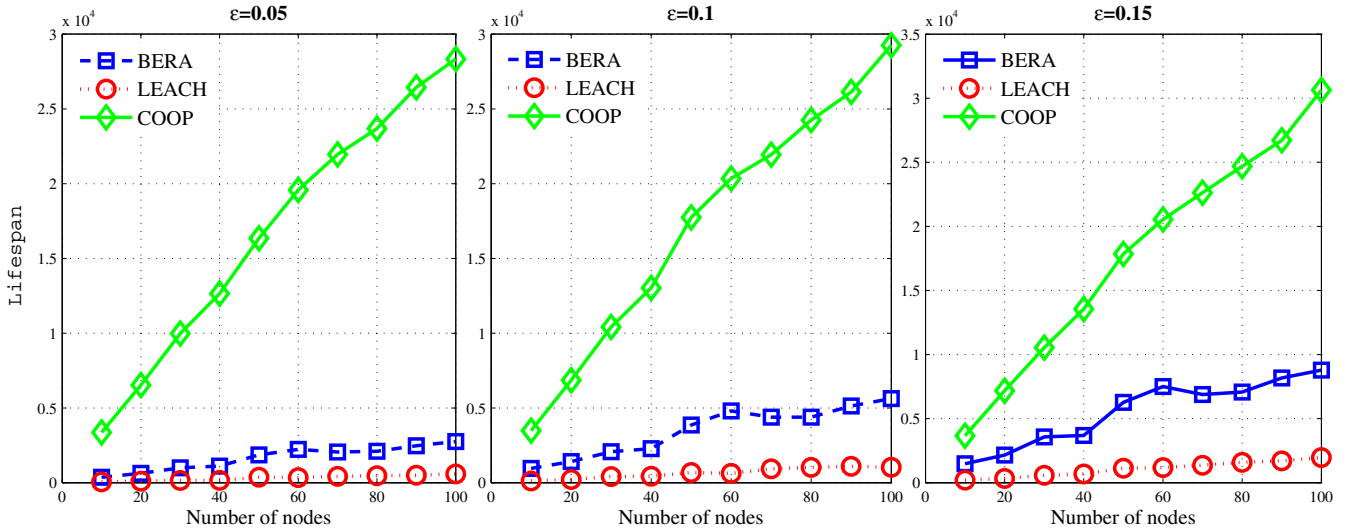


Fig. 1. Performance evaluation of BERA, LEACH, and the cooperative multipath routing algorithm. The simulations were run with $\epsilon = 0.05, 0.10, 0.15$.

fair. On the other hand, since BERA also takes Rayleigh fading into account, it is suitable for reliable data forwarding. Thus, there is no need to modify BERA in order to compare it with the proposed approach. Then the analysis the numerical results will be taken place as well.

In the simulations, values are assigned to the parameters based on the widely used RF module of the CC2420 (these values can be found in [1]). Here, $\alpha = 2$, and the sensor nodes are deployed in a $100m \times 100m$ field and placed randomly with uniform distribution. In the simulations, the number of nodes are varied from 10 to 100.

Now, since LEACH is not designed for reliable routing, it does not take into account the lossy radio links. In order to overcome this shortcomings, and make it comparable with the proposed algorithm, LEACH has to be modified as follows. First, it must be guaranteed that each packet is delivered to the *BS* with at least $(1 - \epsilon)$ success probability. Since LEACH uses cluster heads (*CH*) to relay data from nodes to the *BS*, similarly to the proposed algorithm, to deliver a packet, two hops are needed. Thus, one must to maintain a $\sqrt{1 - \epsilon}$ delivery success probability for each hop (i.e. then the total delivery success probability is $(1 - \epsilon)$). In so doing, the transmission energy of each node has to be set, including the *CH*s, by using equation 1.

Given all this, the numerical result of the simulations is depicted in Figure 1. From this figure, one can see that the proposed cooperative multipath routing algorithm outperforms the other two methods. In particular, it shows an improvement of 200%, compared to the BERA. Furthermore, networks using the algorithm for data forwarding can extend their longevity 14 times, compared to the life span of networks using LEACH.

VIII. CONCLUSION

This paper has focused on the problem of reliable data forwarding in the wireless sensor networks. More precisely,

the challenge here was to provide a routing algorithm that maintains each packet's probability of successful delivery above a certain threshold $(1 - \epsilon)$. In addition, it has also been aimed to maintain energy balancing in the network, that is, the focus was on maximizing the residual energy level of the bottleneck node. Given this, a cooperative multipath routing algorithm has been proposed, that fulfills both objectives. In particular, it has been proved that the proposed algorithm is optimal, in both energy balancing, and determining the optimal relay set. Finally, it has been demonstrated that the proposed algorithm outperforms other existing methods.

REFERENCES

- [1] Chipcon, smartrf cc2420, 2.4ghz ieee 802.15.4/zigbee-ready rf transceiver.
- [2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *Computer Networks*, 38:393–422, 2002.
- [3] C.-Y. Chong and S. P. Kumar. Sensor networks: Evolution, opportunities and challenges. *Proceedings of IEEE*, 91(8):1247–1256, 2003.
- [4] D. De Couto, D. Aguayo, J. Bicket, and R. Morris. A high-throughput path metric for multi-hop wireless routing. *Wireless Networks*, 11(4):419–434, 2005.
- [5] Martin Haenggi. Analysis and design of diversity schemes for ad hoc wireless networks. *IEEE Journal on Selected Areas in Communications*, 23(1):19–27, 2005.
- [6] C. Intanagonwiwat, R. Govindan, and D. Estrin. Directed diffusion for wireless sensor networking. *IEEE/ACM Transactions on Networking*, 11(1):2–16, 2003.
- [7] C. E. Perkins and E. M. Royer. Ad hoc on-demand distance vector routing. In *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, pages 90–100, 1999.
- [8] A. Rogers, D. D. Corkill, and N. R. Jennings. Agent technologies for sensor networks. *IEEE Intelligent Systems*, 24(2):13–17, 2009.
- [9] G. Treplan, L. Tran-Thanh, A. Olah, and J. Leventovszky. Reliable and energy aware routing protocols for wireless sensor networks. In *Proceedings of the 17th international conference on Software, Telecommunications and Computer Networks*, pages 171–175, 2009.
- [10] M. Z. Zamalloa and B. Krishnamachari. An analysis of unreliability and asymmetry in low-power wireless links. *ACM Transactions on Sensor Networks (TOSN)*, 3(2):1–34, 2007.
- [11] M. Zorzi and R. R. Rao. Geographic random forwarding (geraf) for ad hoc and sensor networks: Multihop performance. *IEEE Transactions on Mobile Computing*, pages 3948–3952, 2003.

Object Outline and Surface-Trace Detection Using Infrared Proximity Array

Ákos Tar

(Supervisors: Tamás Roska, György Cserey and Gábor Szederkényi)
tarak@itk.ppke.hu

Abstract—In this paper we demonstrate a new way of detecting outlines and creating surface-traces of various object with a novel low cost paired infra LED and Photo Diode based distance measurement array. We also demonstrate the advantage of the sensor array in detecting the angle of the reflected light and in increasing the resolution. Experiments demonstrate successful and promising results of detected object outlines and surfaces. These sensors give the best results in those environments where prior information of the object distance is given (e.g production lines) or only relative distance information is needed, or they can be used as supplementary sensors with slow, expensive but precise distance measurement devices.

I. INTRODUCTION

Robots must sense their environment in order for safe navigation generally in a non-contact way, whatever the main task is: only obstacle avoidance, object picking - placing, or in a more complex case simultaneous localization and mapping. In general, when a robot is placed into an unknown environment, the only information it can rely on is the sensor based data, so some of the biggest challenges are object detection, classification and localization. The more precise and fast we can get the sensorial information about its vicinity the much faster and more reliable we can react. More of these tasks can be characterized as distance measurement problems and they more or less rely on different kinds of distance measurement sensors.

The robot must know how far an object is and how it looks like and what its orientation is. Camera systems are already used for creating 3D images of the environment, but the reason why mobile robots hardly use the data provided by cameras for low level obstacle avoidance is the required of high computation power. More often, 2D laser scanners are used with a tilt mechanism to create the 3D scan of the environment. They are very accurate but their size and price are backdraws. Traditional distance measurement sensors also could be used for creating 3D images of an object [1],[2].

Ultrasonic (US) and offset based infrared (PSD) sensors are widely used in order to determine an object's distance. The US sensor measures the time of flight (ToF) of the ultrasound signal emitted and reflected to the receiver. In this case, a typical single data acquisition time is 3ms if an object is placed 50cm away. The real back draw of these kind of sensors is the poor angular resolution. The detected object could be anywhere along the perimeter of the US beam because of the wide (typically 35 degree) angular sensitivity of the receiver. Because of the relatively big size ($d = 16mm$, the emitter

and the receiver has the same diameter) dense array cannot be achieved.

Infrared technology uses much narrower beam both in the case of measuring amplitude response and the offset of the reflected light. The most common offset type infrared sensors are the *SharpGP** series. They are very compact, low cost, and only 44mm wide and 13mm high. The output of the sensor is analog, and available in various measurement ranges, the minimal sensing distance is 4cm (*GP2D120* : 4cm – 30cm) the maximum is 5.5m (*GP2Y0A700K* : 1m – 5.5m). Unfortunately sometimes this minimal sensing range and the sensor physical size is just too much, and the sensor has a maximum readout speed of 26Hz (38ms) and the output characteristic is also nonlinear. Researchers already proved it to be useful for object detection [3] and for creating surface-traces of various objects [1], and for localization purposes [4], [5], but because of the sensor speed, real time operation can hardly be achieved.

In this article we would like to show a new infra LED and Photo Diode based distance measurement array and its potential usage for tracing object outlines and surfaces. We also demonstrate the advantage of using a sensor array in the detection of the angle of the reflected light and in increasing the resolution.

Using the emitted light of an infra LED and measuring the amount of light reflected from an object with a Photo Diode in order to measure distance is a well known method, although its applications are mainly restricted for object avoidance and on-off type object detection or, typically, guidance for docking [6], usually utilizing only one infra LED and Photo Diode in pairs. The potential usage for recreating the outline of object or the surface has not been studied yet. The main reasons are its nonlinear characteristic and that the reflected amount of light highly depends on the reflective properties of the object. Our solution may not be as accurate as for example laser scanners or camera systems are, but normally low resolution data is enough for object detection, avoidance and classification tasks. Also in those environments where the operation speed crucial and computation power has to be small we have to make a compromise between resolution and speed.

The reason why it is still worth to focus on the infra LED and Photo Diode based distance measure method is the inherent high resolution both in the time and range and its small dimensions. The readout speed could be in the MHz range, and the output is analog so the distance resolution

will depend on the applied readout circuit and the used Analog Digital Converter (ADC). In [7] they used the infra sensor for submicron level positioning and in [8] for distance measurement on a mobile robot in a 10cm-110cm range.

There are some outstanding articles that utilize infra sensors for distance estimation ([8]) and for localization purposes ([9]). The key is if a prior assumption about the distance of the object is given (based on a US, PSD sensor) then the infra sensors can be used responsibly, or another good method is to try to find the maximum energy of the reflective light [10]. The problem is still that the object distance cannot be measured accurately without knowing its reflective properties, more precisely the sensor gives exactly the same result if the sensed object is close or it is white.

It can also be depicted that in most of the articles normally only a few infra sensors are used on the robots (1 or 2 on each side), each sensor is independent, and the infra LED control is an on-off type. It is also hard to find articles where the sensors are used in arrays [11].

The remainder of this paper is organized as follows. Section II presents the Sensor model and the method to increase the in-depth resolution. Then, in Section III experimental results demonstrates the capabilities of the infrared array. Finally, concluding remarks follow in Section IV.

II. SENSOR MODEL

First we would like to give a general description of the used sensor model and introduce a method how the precision of the distance measurement can be increased by using dynamic illumination infra LED control. In this current research an 8x1 sensor array is used. It suggests that the native resolution is 8 pixels. The sensor array resolution can be extended if during the measurement process not just the appropriate Photo Diode is used but the neighbors as well. The first pixel in our measurement is created by measuring with the first Photo Diode during the first infra LED emitting, the second pixel is measured with also the first Photo Diode but using the second infra LED illumination, and so on, this method results 15 pixels in the array. The array layout and the described pixel measurement method also will help to determinate the angle of incidence.

A. General description of the used model

A pioneering work *G.Benet et al* [8] introduced how the inverse square law can be used to determine an object's distance instead using the Phong illumination model. The used equation is simple (1) and follows the inverse square law, there are only two major parameters that have to be obtained in order to accurately measure an object's distance. The first one is α_i , that is the reflective properties of the sensed object at the viewing area and θ is the angle of incidence.

$$y(x, \theta) = \frac{\alpha_i \cdot \alpha_0 \cdot \cos(\theta)}{x^2} - \beta \quad (1)$$

Where $y(x, \theta)$ is the sensor output, x is the distance of the object, α_0 is constant (containing the radiant intensity of the

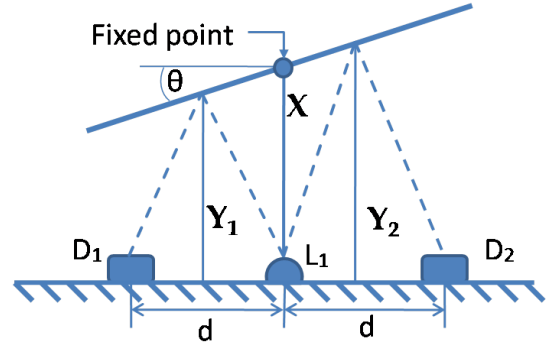


Fig. 1. This figure shows part of the sensor array, two Photo Diodes (D_1, D_2) and between them an infra LED (L_1). The infra LED illuminates the target surface and that reflects the light to Photo Diodes with the angle of incidence θ . The distance based on the first sensor reading is Y_1 , and Y_2 for the second sensor. The d parameter indicates the distance between the infra LED and the Photo Diode.

used infra LED, spectral sensitivity of the Photo Diode, and amplification), and β is the ambient light and the offset voltage of the amplifier. Because the used Photo Diodes do not have daylight filter attached, we use a single measurement without infra light emission to obtain the ambient light and the offset voltage of the amplifier and the measured value can be used as β .

The α_i parameter is usually obtained by using other distance measurement sensors [12], typically US or priori known while an approximation has to be given for θ .

We will demonstrate an iterative solution to accurately calculate the θ parameter. In Fig. 1, we can see how the θ value can be expressed using Eq. (2).

$$\tan(\theta) = \frac{Y_2 - Y_1}{d} \quad (2)$$

Where Y_1 and Y_2 are the distances of the object and d is the distance between the Photo Diode and the infra LED. The iteration process is simple, the first step is to calculate the Y_1' and Y_2' distance (which are approximations for Y_1 and Y_2) using equation (1) with the assumption that $\cos(\theta)$ is equal to 1. Then with equation (2) we can calculate a θ' that can be used to obtain a more precise Y_1' and Y_2' . The iteration process can be continued until the error between two calculated θ s is higher than a given threshold based on the required distance measurement accuracy. Fig. 2, shows the error in degree between the calculated θ' and the real θ at a different number of iterations, in every iteration step the error decreases about a quarter. It can be clearly seen that with only two iteration the highest deviation is 0.3 degree meaning only about $\pm 0.006mm$ uncertainty in the measurement when the angle of incidence is around 45 degree. This iterative process can be done without requiring new sensorial data so it can be implemented to be very fast even in a microcontroller. With this method the iteration number can be dynamically varied based on the requested precision or on the current value of the angle of incidence.

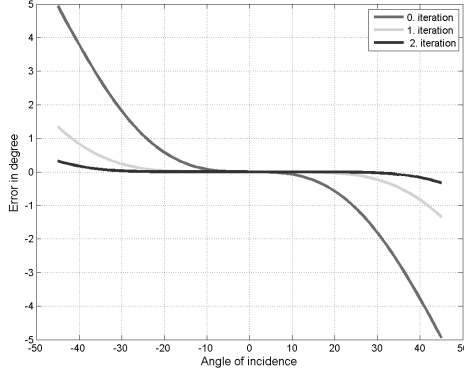


Fig. 2. This figure shows the error level between the real angle of incidence (θ) and the approximation (θ') using Eq. 2, with different number of iteration used. It can be seen that the error level after the 2nd iteration process is smaller than 0.3 degree at the angle of incidence 45 degree. Meaning an approximately $\pm 0.006mm$ uncertainty in the measurement.

III. EXPERIMENTAL RESULTS

In the first experiment we tried to simulate the movement of the production line or any mobile robot (including sensor guided wheelchairs), where the motion in x, y, z direction can be measured based on sensorial data (e.g: odometry) and we are expecting straight line movements. The sensor array was $65mm$ far from the ground and looking down to the floor. As the robot was moving, different kinds of measurements were taken with the sensor array.

The experimental setup is as follows: we placed the sensor array on the z axis (aligned with the x axis and facing down from around 30 cm high to the x, y plane, where the object was placed) of an x, y, z table that is capable of moving with a $10\mu m$ precision. The sensor array was moved only in the y direction and no movement was made in the x and z axis. We tried to model an 8×8 sensor array so the incrementation step in the y direction is set to $8mm$. Approximation for the θ value was only calculated in the x direction based on the measured pixels values in the array.

A. Edge reconstruction

As the first measured object we placed a red wooden block cube ($W = 3.2, H = 3.2, L = 3.2$ cm) on a flat homogeneous surface under the x, y, z table in the x, y plane. After the scanning process we tried to outline and recreate the surface of the detected object. Even though the used infra LEDs are highly directional, it still causes a problem that the emitted light also reflects from the sides of the objects. Thus, the edges on the measured images are rather blurred (especially in the scanning direction) as can be seen in Fig. 3(a). It is because as the sensor array is moving closer to the object, more light reflects from the side of the object to the Photo Diode causing false distance measurement. In order to solve this problem we applied a second order sinusoidal data fit both in the x and y direction of the original image using Eq. (3):

$$y = a_1 * \sin(b_1 * x + c_1) + a_2 * \sin(b_2 * x + c_2) \quad (3)$$

where $a_1, b_1, c_1, a_2, b_2, c_2$ are the model parameters.

Second order sinusoidal approximation was chosen because in case of a straight slope it still could follow the gradient with relatively small error but still can be calculated very fast. The fitted image highly follows the rule how the reflection from the object sides are smoothening the edges. It is because when there is a high-pitched change in the image, both the fitting method and the measurement process will result a sinusoidal like gradient as can be seen in Fig. 3(a,b). Thus, after normalization, the fitted image could be used as a weight function with the original image to decrease the effect of the smoothing. With this method, the gradients at the edges are lowered as can be seen in Fig. 3(c). As in this case the measured object surface was flat, we could use the Prewitt edge detection method to determine the edges of the sides. The output of the filter is marked with white dotted lines in Fig. 3(d,f). In case of the original image, the edge detection method only found two edges while on the final image the real edges are successfully marked. It has to be mentioned that the edge detection outlined a 4×4 pixel array where each pixel size is $8mm$ (both in the x, y direction) suggesting that the scanned object dimensions are $W = 3.2, L = 3.2cm$.

IV. CONCLUSION

In this paper a novel infrared LED and Photo Diode based distance measurement array is presented. The two main advantages of the system are the inherit fast readout speed and resolution. Also the array structure helps to improve the resolution and also helps in the calculation of the angle of incidence. We examined the sensor array capabilities for outline and surface-trace detection of various objects. It is proved to be useful but with limitations. One problem is the deflection that smoothing the edges. And also the reflected amount of light highly depends on the brightness of the object, but this could be improved by using a supplementary distance measurement sensor (e.g US). We think that this sensor array could be useful in many applications for example in production lines for object classification or orientation detection, or in robotic for navigation (landmark detection), obstacle avoidance and detection and for SLAM in home and industrial robotic. In the future we would like to extend the sensor array size to an 8×8 array, and test the capabilities for SLAM.

ACKNOWLEDGMENT

The Office of Naval Research (ONR) and the Operational Program for Economic Competitiveness (GVOP KMA) which supports the multidisciplinary doctoral school at the Faculty of Information Technology of the Pazmany Peter Catholic University is gratefully acknowledged. The authors are also to grateful to Professor Tamas Roska, and the members of the Robotics lab for the discussions and their suggestions.

REFERENCES

- [1] Y. Omura, A. Goto, and N. Shidara, "Surface-Trace Feasibility for IR-Based Position-Sensing Devices," *IEEE SENSORS JOURNAL*, vol. 9, no. 10, 2009.

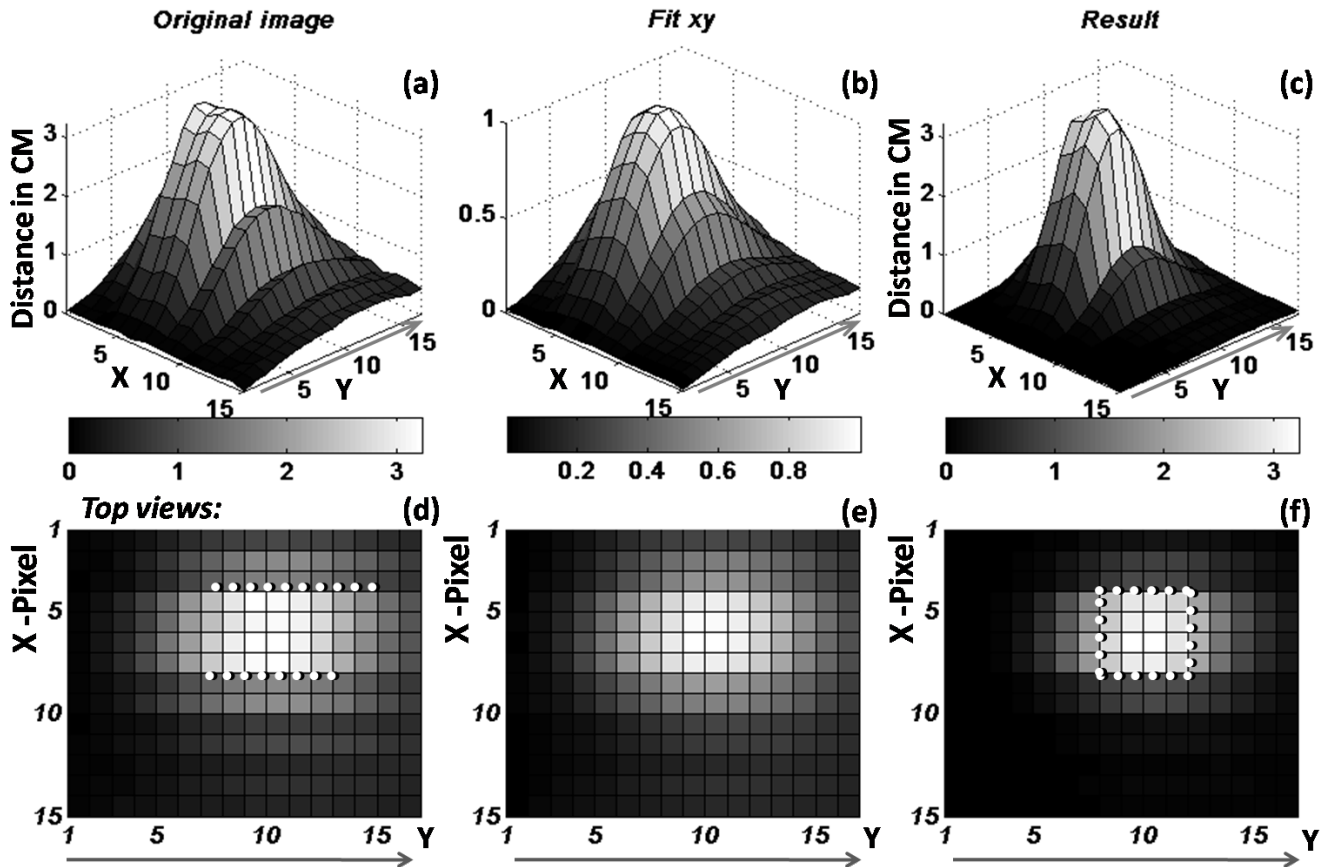


Fig. 3. Measurement result of a red wooden brick cube with dimensions of 3.2cm on each side. The sensor was mounted on the z axis of a x, y, z table, aligned with the x axis and looking down to the x, y plane where the object was placed. The table was moved with 8mm incremental steps. (a) shows the original image created from the sensor array data output after back light and offset cancellation, and angle of incidence approximation. On each image the scanning direction is marked with an arrow. The x axis indicates the number of the pixels, and the z axis shows the distance in cm. On (b) a second order sinusoidal fit in the x, y direction of the original image can be seen. It will be used to correct the false distance measurement near the edges caused by the reflection from the sides. (c) shows the result of the data process. As can be seen, the gradients at the edges are much steeper. (d), (e), (f) are the same as the above images but from the top view. As the top of the object was flat we could use the Prewitt edge detection method to outline the object edges, the result is marked with dotted white line on (d),(f). As it can be seen in the original image (d) this method only found two edges, but on the result image (f) the real edges of the object are outlined. It has to be mentioned that the edge detection outlined a 4x4 pixel array where each pixel size is 8mm (both in the x, y direction) suggesting that the scanned object dimensions are $W = 3.2, L = 3.2$ cm.

- [2] M. Baba, K. Ohtani, and S. Komatsu, "3D shape recognition system by ultrasonic sensor array and genetic algorithms," in *Instrumentation and Measurement Technology Conference, 2004. IMTC 04. Proceedings of the 21st IEEE*, vol. 3, 2004.
- [3] H. Park, S. Lee, and W. Chung, "Obstacle Detection and Feature Extraction using 2.5 D Range Sensor System," in *SICE-ICASE, 2006. International Joint Conference*, 2006, pp. 2000–2004.
- [4] H. Park, S. Baek, and S. Lee, "IR sensor array for a mobile robot," in *2005 IEEE/ASME International Conference on Advanced Intelligent Mechatronics. Proceedings*, 2005, pp. 928–933.
- [5] S. Lee and W. Chung, "Rotating IR Sensor System for 2.5 D Sensing," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006, pp. 814–819.
- [6] P. Vaz, R. Ferreira, V. Grossmann, M. Ribeiro, I. Norte, and A. Pais, "Docking of a mobile platform based on infrared sensors," in *IEEE International Symposium on Industrial Electronics, Guimaraes, Portugal*. Citeseer, 1997.
- [7] Y. Shan, J. Speich, and K. Leang, "Low-Cost IR Reflective Sensors for Submicrolelevel Position Measurement and Control," *IEEE/ASME Transaction on Mechatronics*, vol. 13, no. 6, 2008.
- [8] G. Benet, F. Blanes, J. Simo, and P. Perez, "Using infrared sensors for distance measurement in mobile robots," *Robotics and autonomous systems*, vol. 40, no. 4, pp. 255–266, 2002.
- [9] D. Navarro, G. Benet, and F. Blanes, "Line-based incremental map building using infrared sensor ring," in *IEEE International Conference on Emerging Technologies and Factory Automation, 2008. ETFA 2008*, 2008, pp. 833–838.
- [10] M. Garcia and A. Solanas, "Estimation of distance to planar surfaces and type of material with infrared sensors," in *Proceedings of the 17th International Conference on Pattern Recognition*, vol. 1, 2004, pp. 745–748.
- [11] A. Tar, M. Koller, and G. Cserey, "3D geometry reconstruction using Large Infrared Proximity Array for robotic applications," in *IEEE International Conference on Mechatronics, ICM 2009.*, 2009, pp. 1 – 6.
- [12] A. Flynn, "Combining sonar and infrared sensors for mobile robot navigation," *The International Journal of Robotics Research*, vol. 7, no. 6, p. 5, 1988.

Control Authority of a Five Link Planar Biped Underactuated by One

József Veres

(Supervisors: Dr. James Schmiedeler, Dr. György Cserey and Dr. Gábor Szederkényi)
verjo@digitus.itk.ppke.hu

Abstract—Biped robots that are designed for walking form a special subclass of the robotics. Since these are common in the sense that all are mechanical devices having lower number of actuators than degrees of freedom. This naturally leads to their well known characteristic of being unstable. The goal of stabilization or so called balancing is an on-going research problem of the field. In this paper an ERNIE [1] like five link planar biped system is presented which is underactuated by one degree of freedom (DOF). This underactuation means we have no direct control over that unactuated joint. The only way to gain control is to exploit the influence of the other actuated joints. This kind of indirect control requires a quantitative description of the coupling between the actuated and the unactuated joints. In this work for that purpose the Intrinsic Vector-Valued Symmetric Bilinear Form [2] is used as a novel approach. That formulization was previously proven to be effective for a simple mechanical control system [3]. In the following sections one can find the detailed steps of the above mentioned method. After an introduction the second section will introduce the used mechanical system by showing the details of the used model. The third section will cover the control authority formulization. And at the end of the paper the result of the approach will be showed.

I. INTRODUCTION

Biped robots belong to the legged locomotion field of the robotics. These two-legged mechanisms are designed for locomotion. By their nature all these mechanical devices are underactuated namely having lower number of actuators than degrees of freedom, which leads to their well known characteristic of being unstable. The task of stabilization or so called balancing is an on-going research problem of the field. There are two major types of how this postural stability is tried to be maintained.

he first one is the Zero Moment Point (ZMP) criteria [4]. This has become very popular in the last decades due to its relative easy applicability even for high DOF. This is what are used in numerous state-of-the-art humanoid robots [5],[6]. The main disadvantages of this approach are the significant limitation in the robot motion capabilities and its low efficiency [7].

The second one is the Limit Cycle analysis [8]. This approach shows superior characteristics in both efficiency and agility by releasing the constraints even more. Namely one step of a gait has to be stable as a whole but not locally stable at every instant in time. This usually means that after a computational extensive trial and error or more sophisticated optimization algorithms we come up with a gait which is said to be stable.

But both approaches suffer from the lack of good disturbance rejection. For example in the case of the ZMP criteria as the zero moment point gets out of the feet's support polygon or in the other case as we deviate from the predefined stable gait's trajectory by an external disturbance we are facing the problem of postural stability again. Unfortunately we can not ignore the existence of the external disturbances since these robots have to work in a the real world not in a completely isolated environment.

This paper attempts to show a novel approach for this problem by introducing control authority measure of the system over the underactuated DOF. For this purpose the Intrinsic Vector-Valued Symmetric Bilinear Form [2] is used, which was previously proven to be effective for a simple mechanical control system [3]. The idea is to come up with a quantitative description of how the actuated joints influence the unactuated one which will give us the control authority measure of the system over the unactuated DOF. Since the underactuation means we have no direct control, the only way to act on it is to use the natural coupling between the actuated and unactuated DOF. This kind of coupling is intuitive if we think about how the robot configuration can effect the unactuated joint by the gravity. But not only position dependent coupling exist but velocity dependent too. The formulization of the total coupling for a given biped robot is not a trivial problem that is why this approach is novel. By using this technique one can write it up in a coordinate invariant manner for any system where the Lagrangian dynamics can be formulated.

In this paper an ERNIE [1] like five link planar biped robot is presented on which the above mentioned approach will be demonstrated. It is underactuated by one and since it is planar it moves in the sagittal plane. Despite the fact that it is planar and having relatively low DOF this biped type is an excellent testbed for locomotion research and are widely used in the robotic community [9],[10],[11],[1]. In the second section the chosen mechanical system will be described. This will contain the model itself, the parameters and its lagrangian dynamics will also be formulated. In the third section the control authority formulization for this previously described mechanical system will be presented in step by step. In the forth section experimental results will be showed that was symbolically computed in Wolfram Research's Mathematica and numerically in MATLAB. Here an effective way of real-time computation is also suggested. And at the last section the whole work will be summarized.

II. DESCRIPTION OF THE MECHANICAL SYSTEM

In order to model the walking of a planar biped robot a step must be divided into different phases. First we can categorize the stance phase as single or double support depending on whether one or two leg is touching the ground. In this case I assume that the double support phase is just instantaneous and therefore it is neglectable, which is a commonly accepted simplification [9],[1]. So now we only have a swing phase and an impact phase. Since during a normal periodic walking the swing phase dominates in most of the time now I will focus on modeling of that portion of a step. This part is when one leg is in contact with the supporting surface and the other leg swings in the air where that contact is considered to be only a point contact. This can be modeled as an open chain multi-link manipulator described by differential equations. Now I am going to present this model for a five link planar biped.

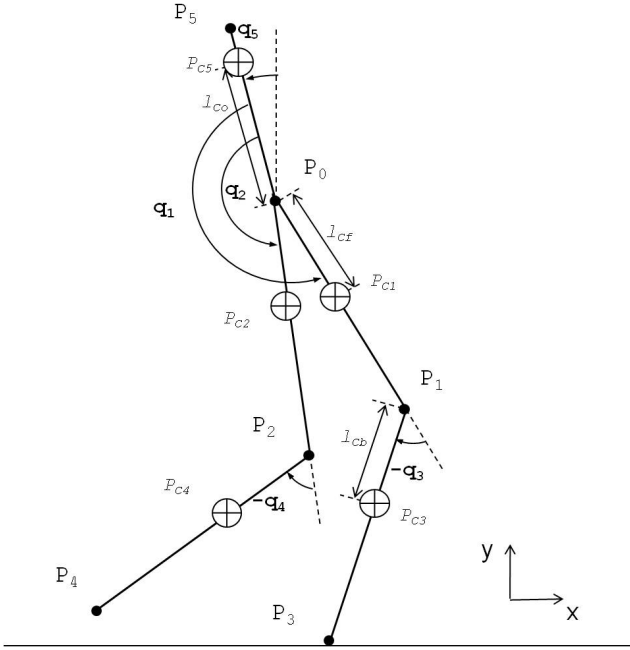


Fig. 1. This figure shows a five link planar biped model parameters and joint variables. In each of those links the "⊕" symbols represent the COM of a given link. Both the joints' and the link's COMs' points cartesian coordinates are depicted as P_1, P_2, P_3, P_4, P_5 and $P_{C1}, P_{C2}, P_{C3}, P_{C4}, P_{C5}$ respectively. The q_1, q_2, q_3 and q_4 are the actuated and q_5 is the unactuated coordinate all measured in radian.

The Fig. 1 illustrates the joint variables and other model parameters. As we can see it has a pair of femur and tibia plus a torso link. In each of those links the "⊕" symbols represent the center of mass (COM) of a given link. Both the joints' and the link's COMs' points cartesian coordinates are depicted as P_1, P_2, P_3, P_4, P_5 and $P_{C1}, P_{C2}, P_{C3}, P_{C4}, P_{C5}$ respectively. The q_1, q_2, q_3 and q_4 are the actuated and q_5 is the unactuated coordinate all measured in radian. The reason for this measurement convention is the following. In order to keep these planar robots in the sagittal plane one of the most commonly used technique is to attach a boom to its torso via an unactuated revolute joint coaxial with the hips. The revolute

connection allows for the body to pitch relative to the boom. So this is just a practical issue but the system behaves like it was introduced at the beginning of this section. The Table I shows the model's geometric and inertial parameters that are used. All inertias are given about the COM of a given link.

TABLE I
MODEL'S GEOMETRIC AND INERTIAL PARAMETERS

Model Parameters	Symbol	Units	Link	Value
Mass	m_o	kg	torso	13.6
	m_f		femur	1.5
	m_b		tibia	1.0
Length	l_o	m	torso	0.28
	l_f		femur	0.36
	l_b		tibia	0.36
Mass center	l_{co}	m	torso	0.14
	l_{cf}		femur	0.13
	l_{cb}		tibia	0.12
Inertia	J_o	$\text{kg}\cdot\text{m}^2$	torso	0.09
	J_f		femur	0.02
	J_b		tibia	0.02

Now let's come up with the equation of motion which I am going to calculate by the help of the Lagrange's equation. In order to do that I have to write up the kinetic and potential energies of the system. First I am starting with the positions where I assume that every P position is written in vector format $[P_x, P_y]^T$. Then the 5+5 cartesian coordinates are:

$$\begin{aligned}
 P_3 &= \begin{bmatrix} 0 \\ 0 \end{bmatrix} & P_{C3} &= P_3 + (l_b - l_{cb}) \begin{bmatrix} \sin(\hat{q}_3) \\ -\cos(\hat{q}_3) \end{bmatrix} \\
 P_1 &= P_3 + l_b \begin{bmatrix} \sin(\hat{q}_3) \\ -\cos(\hat{q}_3) \end{bmatrix} & P_{C1} &= P_1 + (l_f - l_{cf}) \begin{bmatrix} \sin(\hat{q}_1) \\ -\cos(\hat{q}_1) \end{bmatrix} \\
 P_0 &= P_1 + l_f \begin{bmatrix} \sin(\hat{q}_3) \\ -\cos(\hat{q}_3) \end{bmatrix} \\
 P_2 &= P_0 + l_f \begin{bmatrix} -\sin(\hat{q}_2) \\ \cos(\hat{q}_2) \end{bmatrix} & P_{C2} &= P_0 + l_{cf} \begin{bmatrix} -\sin(\hat{q}_2) \\ \cos(\hat{q}_2) \end{bmatrix} \\
 P_4 &= P_2 + l_b \begin{bmatrix} -\sin(\hat{q}_4) \\ \cos(\hat{q}_4) \end{bmatrix} & P_{C4} &= P_2 + l_{cb} \begin{bmatrix} -\sin(\hat{q}_4) \\ \cos(\hat{q}_4) \end{bmatrix} \\
 P_5 &= P_0 + l_o \begin{bmatrix} -\sin(\hat{q}_5) \\ \cos(\hat{q}_5) \end{bmatrix} & P_{C5} &= P_0 + l_{co} \begin{bmatrix} -\sin(\hat{q}_5) \\ \cos(\hat{q}_5) \end{bmatrix}
 \end{aligned}$$

Where $\hat{q}_1 = q_5 + q_1$, $\hat{q}_2 = q_5 + q_2$, $\hat{q}_3 = q_5 + q_1 + q_3$, $\hat{q}_4 = q_5 + q_2 + q_4$ and $\hat{q}_5 = q_5$. With the help of these position vectors now we can formulate the kinetic (K) and potential (U) energies, where the y as superscript denotes the y cartesian coordinate of a P point and the dot over a symbol represents the time derivative of that symbol.

$$\begin{aligned}
 U &= g(m_f(P_{C1}^y + P_{C2}^y) + m_b(P_{C3}^y + P_{C4}^y) + m_o P_{C5}^y) \\
 K &= \frac{1}{2} (m_f(\|\dot{P}_{C1}\|^2 + \|\dot{P}_{C2}\|^2) + \\
 &\quad + m_b(\|\dot{P}_{C1}\|^2 + \|\dot{P}_{C2}\|^2) + m_o P_{C5}^y) + \\
 &\quad \frac{1}{2} (J_f(\dot{q}_1^2 + \dot{q}_2^2) + J_b(\dot{q}_3^2 + \dot{q}_4^2) + J_o \dot{q}_5^2) \quad (1)
 \end{aligned}$$

Now we can come up with the equation of motion of the system by writing up the Lagrange's equation for $L = K - U$.

$$Le = \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{Q}} \right) - \frac{\partial L}{\partial Q} \quad (2)$$

Where $Q = [q_1, q_2, q_3, q_4, q_5]^T$. By using the Lagrange's equation (Le) one can calculate the system's mass matrix (M), centrifugal and Coriolis matrix (C) and the gravity term (G).

$$M = \frac{\partial Le}{\partial \ddot{Q}} \quad C = \frac{1}{2} \frac{\partial Le}{\partial \dot{Q}} \quad G = \frac{\partial U}{\partial g} g \quad (3)$$

Where g denotes the gravity constant. So finally by using those terms the state-space equation could be formulated in the following way. (Equation 4)

$$M(Q)\ddot{Q} + C(Q, \dot{Q})\dot{Q} + G(Q) = Bu \quad (4)$$

Where B is a 5×5 identity matrix, $u = [\tau_1, \tau_2, \tau_3, \tau_4, 0]^T$ and τ_n are the joint torques. Now the system is described and we are ready to formulate the control authority of this actually presented mechanical system.

III. CONTROL AUTHORITY FORMULATION

The formulation that is based on the Intrinsic Vector-Valued Symmetric Bilinear Form [2] is mathematically complex so it is presented here in a simplified form. The derivation and proof could be found in the referenced work [2]. The concept is to decompose velocity of the system into component velocities in controlled and uncontrolled directions. By calculating that decomposed uncontrolled velocity's rate of change one will be able to determine how the controlled velocities influence the uncontrolled velocity.

$$v = \omega_1 Y_1 + \omega_2 Y_2 + \omega_3 Y_3 + \omega_4 Y_4 + s Y_\perp \quad (5)$$

Where $v = [\dot{q}_1, \dot{q}_2, \dot{q}_3, \dot{q}_4]^T$ is the vector of system velocities, ω_i is the controlled component velocity in direction Y_i , and s is the uncontrolled velocity in direction Y_\perp . Let's write up Y_i as the following.

$$Y_1 = \check{M} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad Y_2 = \check{M} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad Y_3 = \check{M} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad Y_4 = \check{M} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

Where \check{M} is the inverse of the mass matrix. The uncontrolled direction Y_\perp must be M -orthogonal to all Y_i where M is the mass matrix. Then the rate of change of the uncontrolled velocity s is the following.

$$\frac{ds}{dt} = -\omega_a \omega_p B_1^{ap}(Y_a, Y_p) - s \omega_a B_2^a(Y_a) - s \omega_p B_3^p(Y_p) - s^2 B_4(Y_\perp) - Grav \quad (6)$$

Where the B_i 's are scalar coefficients which depend on the velocity direction, $Grav$ is the contribution from gravity and the repeated a and p indices imply summation. The B_1, B_2 and B_3 coefficients are the so called coupling coefficients since those indicates how the controlled velocities are coupled

with the uncontrolled velocity. This formulation is useful since one could easily see how the choice of the velocities that are directly controllable could effect indirectly the unactuated joint. And also have a measure for the relation of the terms including the contribution of the gravity.

Now let's see how these coefficients and gravity terms could be calculated in coordinates.

$$B_1^{ap}(Y_a, Y_p) = \left(\frac{\partial Y_p^k}{\partial Q^i} Y_a^i + \Gamma_{ij}^k Y_a^i Y_p^j \right) M_{kl} Y_\perp^l \quad (7)$$

$$B_2^a(Y_a) = \left(\frac{\partial Y_\perp^k}{\partial Q^i} Y_a^i + \Gamma_{ij}^k Y_a^i Y_\perp^j \right) M_{kl} Y_\perp^l \quad (8)$$

$$B_3^p(Y_p) = \left(\frac{\partial Y_p^k}{\partial Q^i} Y_a^i + \Gamma_{ij}^k Y_a^i Y_p^j \right) M_{kl} Y_\perp^l \quad (9)$$

$$B_4(Y_\perp) = \left(\frac{\partial Y_\perp^k}{\partial Q^i} Y_\perp^i + \Gamma_{ij}^k Y_\perp^i Y_\perp^j \right) M_{kl} Y_\perp^l \quad (10)$$

$$Grav = \frac{\partial U}{\partial Q_i} Y_\perp^i \quad (11)$$

Where over the repeated indices Einstein summation convention is used and Γ_{ij}^k Christoffel symbols are given in the following way. (Equation 12)

$$\Gamma_{ij}^k = \frac{1}{2} M_{kl}^{-1} \left(\frac{\partial M_{il}}{\partial Q_j} + \frac{\partial M_{jl}}{\partial Q_i} + \frac{\partial M_{ij}}{\partial Q_l} \right) \quad (12)$$

Where repeated indices imply summation as before. With this last step we become able to calculate the control authority formula in every time instance during a gait. The next section will present experimental results of this approach.

IV. EXPERIMENTAL RESULTS

In the previous section the in question mechanical system's control authority has been formulated. If we computed everything in a symbolic way the task to get the results would become very computational expensive problem. I found that solving it in a mixed symbolic and numeric method gives an optimal solution. So I used the Wolfram Research's Mathematica to compute all the partial derivatives then I transferred those to MATLAB to have the numerical iterative calculations in real-time.

I also found that some of the B_i 's coefficients are always zero for this particular system namely the B_1, B_2 and the B_4 . That the B_1 coefficients is zero that means the rate of change of s can not be changed arbitrary by the ω_i 's simply. So B_3 is the only non-zero coupling coefficients which is multiplied by both the the ω_i 's and the s . The other remaining term is the $Grav$ of course. With this approach I analyzed more than 10 different gaits of ERNIE. By comparing these terms I found that nearly in all cases the $Grav$ term dominates, but the other four B_3^p scaled terms were neither neglectable. Figure 2 shows these terms for the 10 different gaits consecutively. The $Grav$ term looks like a sawtooth signal that highlights the gait boundaries. We can find regions within a gait where the gravity term could be over dominated. With these priori information one can get an idea even in real-time for a given time instance how could the unactuated joint be indirectly controlled.

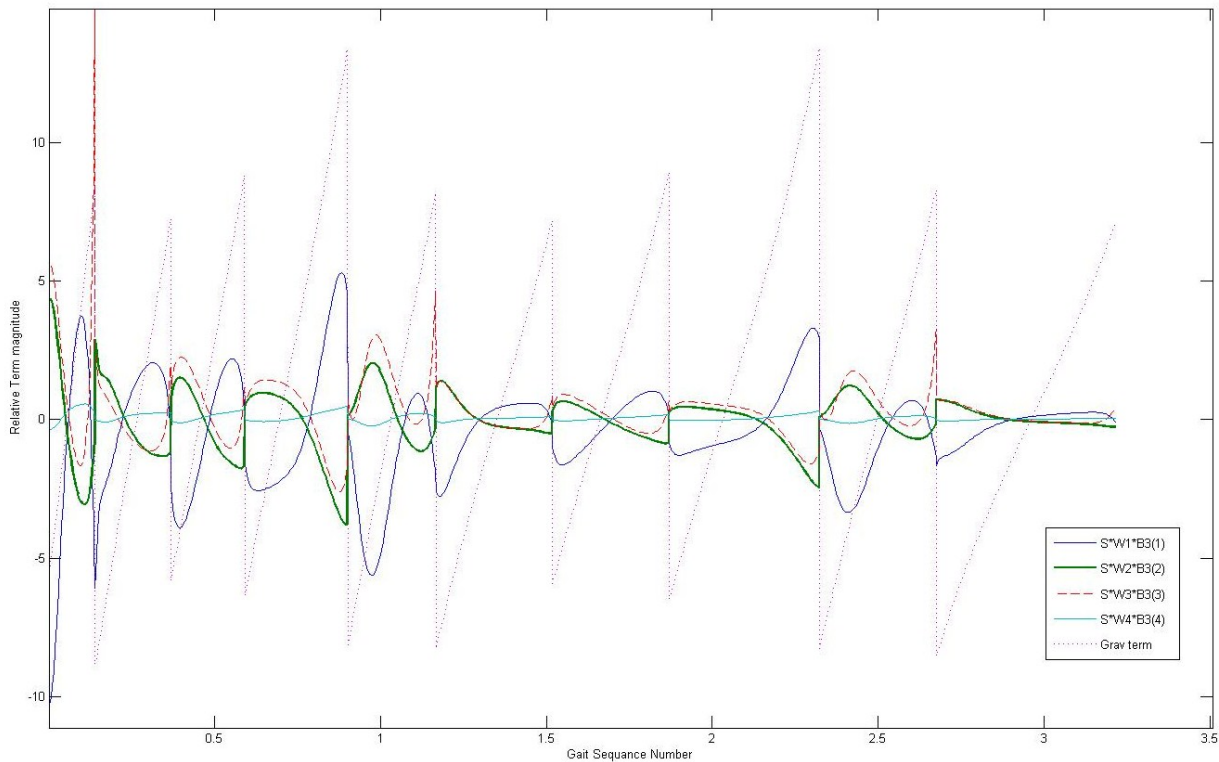


Fig. 2. This figure shows the unactuated velocity's (s) rate of change's five terms during 10 different gaits that was recorded during an experiment on planar biped robot ERNIE [1]. The *Grav* term looks like a sawtooth signal that highlights the gait boundaries. We can find regions within a gait where the gravity term could be over dominated. With these priori information one can get an idea even in real-time for a given time instance how could the unactuated joint be indirectly controlled.

V. CONCLUSION

In this work the control authority of a five link planar biped robot underactuated by one was presented. The robot's model and all modelling parameters was also given. With the help of the Intrinsic Vector-Valued Symmetric Bilinear Form the control authority was formulated in detailed steps. And finally it was examined on different experimental gaits.

ACKNOWLEDGEMENTS

The Hungarian Scientific Research Fund (OTKA) which supports the multidisciplinary doctoral school at the Faculty of Information Technology of the Pázmány Péter Catholic University is gratefully acknowledged. And also the University of Notre Dame du Lac which was supporting this work. The author is also grateful to James Schmiedeler, Tamás Roska, György Cserey, Gábor Szederkényi, David Post, Travis Brown and the members of the Robotics lab for the discussions and their suggestions.

REFERENCES

[1] T. Yang, E.R. Westervelt, J.P. Schmiedeler, and R.A. Bockbrader, "Design and control of a planar bipedal robot ERNIE with parallel knee compliance," *Auton. Robots*, vol. 25, 2008, o. 317-330.
 [2] J. Nightingale, R. Hind, and B. Goodwine, "Intrinsic vector-valued symmetric form for simple mechanical control systems in the nonzero velocity setting," *Robotics and Automation*, 2008. ICRA 2008. IEEE International Conference on, 2008, o. 2435-2440.

[3] J. Nightingale, R. Hind, and B. Goodwine, "A Stopping Algorithm for Mechanical Systems," *Algorithmic Foundation of Robotics VIII*, 2009, o. 167-180.
 [4] M. Vukobratovic and B. Borovac, "Zero-moment point-thirty five years of its life," *International Journal of Humanoid Robotics*, vol. 1, 2004, o. 157173.
 [5] K. Hirai, M. Hirose, Y. Haikawa, and T. Takenaka, "The development of Honda humanoid robot," *IEEE International Conference on Robotics and Automation*, 1998, o. 13211326.
 [6] S. Kajita, T. Nagasaki, K. Kaneko, and H. Hirukawa, "ZMP-Based Biped Running Control," *Robotics & Automation Magazine, IEEE*, vol. 14, 2007, o. 63-72.
 [7] A.D. Kuo, "Choosing your steps carefully," *IEEE Robotics & Automation Magazine*, vol. 14, 2007, o. 1829.
 [8] D.G. Hobbelen and M. Wisse, "Limit cycle walking," *Humanoid Robots, Human-like Machines*, 2007.
 [9] C. Chevallereau and P. Sardain, "Design and Actuation Optimization of a 4 axes Biped Robot for Walking and Running", *Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 3365-3370, San Francisco, CA-USA, April 2000.
 [10] J.W. Grizzle, J. Hurst, B. Morris, H. Park, és K. Sreenath, "MABEL, a new robotic bipedal walker and runner," *Proceedings of the 2009 conference on American Control Conference*, St. Louis, Missouri, USA: IEEE Press, 2009, o. 2030-2036.
 [11] B.T. Knox, "Design of a biped robot capable of dynamic maneuvers," *Ohio State University*, 2008.

