DOI:10.15774/PPKE.ITK.2024.006

Determination of geometrical properties of dynamic scenes by using motion features and motion statistics in 2D imaging sensors

László Havasi

A thesis submitted for the degree of
Doctor of Philosophy

Scientific adviser:
Tamás Szirányi, D. Sc.

Péter Pázmány Catholic University
Faculty of Information Technology

Budapest, 2006

DOI:10.15774/PPKE.ITK.2024.006

Acknowledgements

I would like to say thanks

To *my family*.

To *prof. Tamás Szirányi* for his patience, help and support during my study and work.

To *prof. Tamás Roska* for his support during my study and work.

To my friend and colleague *Zoltán Szlávik*.

To my colleagues, with whom I discussed lot of questions:

Csaba Benedek, István Kóbor and *Levente Kovács*.

To *prof. Michael Rudzsky* and *Miklós Rásonyi*, who provided me with fruitful discussion, ideas and help.

To the MTA-SzTAKI for the support of my Ph.D. studies.

Finally, thanks National R&D Program of Hungary, TeleSense project, Grant 035/02/2001; NoE MUSCLE project of the European Union, Economic Competitiveness Operative Programme of Hungary, Video Indexing project, Grant AKF388.

DOI:10.15774/PPKE.ITK.2024.006

Contents

DOI:10.15774/PPKE.ITK.2024.006

DOI:10.15774/PPKE.ITK.2024.006

List of figures

DOI:10.15774/PPKE.ITK.2024.006

DOI:10.15774/PPKE.ITK.2024.006

DOI:10.15774/PPKE.ITK.2024.006

DOI:10.15774/PPKE.ITK.2024.006

List of tables

DOI:10.15774/PPKE.ITK.2024.006

1.Introduction

The process of extracting and tracking of human figures in image-sequences is a key issue for video surveillance and video-indexing applications. The need for automated person identification systems strongly motivates this interest. The process can be broken down into the following steps: detection [51], tracking, classification [52] and identification [53][82] of human movement or gait. There are several approaches for each of these sub-problems. A useful and popular approach is based on silhouette analysis [55] with spatio-temporal representation, where the goal is to achieve an invariant representation of the detected object. In [82] symmetries of the silhouette are utilized as a gait parameter for person-identification. Other methods focus on the legs [56] and periodicity of human movements [51][57]. We present a simple motion pattern generation and extraction method, which extracts and tracks the symmetries of objects using the images of exactly two legs walking. This task is a binary classification problem: the periodicity of human walking, together with the characteristic human shape of the target, provides key differences which enable us to distinguish pedestrians from the motion patterns of other objects. Our approach uses the motion information contained in video sequences, so that the extracted motion patterns consist information about the spatio-temporal changes of a moving object.

Reflections and cast shadow in surveillance videos usually cause problems in image analysis [29]. This is because it appears in the foreground mask extracted by using an adaptive background model e.g. [30]. In turn, the inaccurate mask reduces the performance of the further image-processing steps. Consequently, techniques for the avoidance of such disturbances constitute an active current research area [29][31]. Construction of an accurate geometric model of the camera-mirror scene forms the basis for our ultimate goal, namely the integration of the model and statistics into a foreground-extraction method which is more reliable than previous approaches. We present a method which integrates the estimated geometric model and the extracted statistics to enable removal of the pixels related to reflection and cast shadow. Our goal is not to present an all-in-one algorithm for shadow detection, but the main idea

DOI:10.15774/PPKE.ITK.2024.006

is to clamp features into a probabilistic framework. During evaluation outdoor sequences will be used, which is impaired by strong shadow and showing pedestrians.

In recent years there has been a dramatic increase in the number of video surveillance systems in use; and these have in turn generated a large quantity of archived video recordings, which are usually stored without any image-processing. In most cases for such recordings one does not know the relative and global geometrical properties of the surveillance cameras. Despite this, there is a striking lack of publications concerning the extraction of geometric characteristics from images contained in video recordings. We may note that this task is much simplified in the case where some known test object is used during system calibration. In this dissertation however a statistical framework is introduced which allows us, without such calibration to derive the horizontal vanishing line (VL).

In videos captured by analog surveillance cameras the contrast and focus are often badly adjusted, and thus precise measurements are not possible in individual frames. This consideration led to our concept of summarizing the information from a sequence of a number of frames (as many as possible) in order to achieve higher accuracy in the averaged retrieved information. The only information that is used is the change-mask of moving objects, or more generally the change-detection (binarized intensity-change) mask. This is the basic information that can be extracted from a video sequence without making any *a priori* assumptions about scene content. Both the empirical and the theoretical results confirm that the method is robust and is fairly insensitive to inaccuracy of the motion-mask. Furthermore, the method has no requirement for any time-consuming preprocessing steps (e.g. object detection, tracking or motion analysis). Another advantage of the method is that it is capable of working on low frame-rate videos, since the relevant parameter for the statistical information extraction is not the refresh rate itself, but rather the total frame-count of the processed sequence.

We introduce a novel exploitation of the so-called co-motion statistics of a video sequence, and demonstrate the method's robustness for correspondence detection. In [7], co-motion statistics were used for image-registration (homography estimation) and the method was tested in images of outdoor scenes. In contrast to [7] the statistics have been investigated in a model-based framework in this work. We

DOI:10.15774/PPKE.ITK.2024.006

have introduced a theoretical investigation of the method's robustness. Happily, the analysis supports our empirical confidence in this statistical method.

DOI:10.15774/PPKE.ITK.2024.006

2.Examined geometrical models

The camera's sensor array reflects a 2D image about the 3D real word. This transformation is a projection onto the camera plane. Despite the information loss after the 3D to 2D transformation the 2D image still contains useful description about the scene geometry.

This chapter describes the investigated geometrical models and their properties with the basic computational methods. We will show the most important properties of models with the explanation of usual computation techniques.

DOI:10.15774/PPKE.ITK.2024.006

2.1. Plane homography

Registration between partially overlapping wide baseline views of the same scene is an important task in a number of applications involving multi-camera systems, such as stereovision, three-dimensional reconstruction, or object tracking/observation in surveillance systems [7][59].

Registration between non-overlapping views is still a challenge and it is only solved in special cases [60][61][62]. It will be shown experimentally that, in case of linear motion, our feature detection method provide usable information for registration of non-overlapping views. The cameras in this test are pointed in opposite directions and they are mounted on different sides of the same wall.

2.1.1. Overlapping views

The problem can be summarized as follows: given a set of points x_i in a view and a corresponding set of points x'_i in another view, we need to compute the projective transformation (2D homography) that takes each element x_i to x'_i (vectors are in homogenous form). The problem is to compute a 3x3 matrix, H (point map), such that:

$$x' = Hx \tag{2.1}$$

This computation can be accomplished in several ways; details can be found in [24]. To solve the problem, we need at least four point-correspondences.

2.1.2. Non-overlapping views

In our surveillance system, there is a non-overlapping camera configuration where the persons walk from the view of “entrance” camera to the view of “main hall” camera. The motion from one view to the other is rectilinear in this configuration and the following computation is utilizing this property.

The computation differs from the overlapping case because corresponding points could not be detected. An alternative way to determine the matrix H is the use of line correspondences instead of point correspondences [24]. This approach needs the

DOI:10.15774/PPKE.ITK.2024.006

assumption that the motion is along a straight line and these line fragments may be detected in both views. Equation (2.2) formulates the problem to compute matrix H:

$$l_i' = H^{-T} l_i \tag{2.2}$$

Where H^{-T} (inverse-transpose) is the line map corresponding to the point map H and l is a three elements vector representation of a line in 2D defined by the join of two points. The points lie on the line $l=[A\ B\ C]$ when satisfy the equation:

$$Ax + By + C = 0 \tag{2.3}$$

The line sets built from two successive walk-steps (a walk cycle) which define two points on the ground-plane, thus the parameters of the line across these points can be calculated directly. The matrix equation with a minimal solution requires four corresponding lines in general position.

2.2. Vanishing point in a single view

Sets of parallel lines in 3D space are projected into a 2D image obtained with a pin-hole camera to a set of concurrent lines. The meeting point of these lines in the image plane, is called a *vanishing point*, and may eventually belong to the line at infinity of the image plane in the case of 3D lines parallel to the image plane.

The determination of the position of the vanishing point [23] (or focus of expansion, FOE [24], or mirror pole [25]) in case of a skew-symmetric fundamental matrix is a task that has rarely been the object of investigation, especially for cases where the input is a noisy outdoor video sequence which contains a planar reflective surface, or of shadows cast on the ground-plane. The importance of this task lies in the fact that knowledge of the position of the vanishing point (henceforward: VP) enables the geometrical modeling of secondary images visible in a planar reflective surface. These situations occur frequently in surveillance videos, and they inevitably cause problems in further image-processing steps and reduce the processing system's performance. Most previous publications which have focused on the use of a mirror to accomplish the 3-D reconstruction task have done so only for an indoor scene [23][25][26]; moreover, most of these works have relied on hand-selected point correspondences.

A principal theoretical foundation in handling this aspect of the topic is the mirror-stereo theorem. This posits that the view of a scene containing a mirror taken with a

projective camera is equivalent to a combination of two views from two projective cameras; and hence that traditional processing methods for two-view stereo images can be applied [26]. Here the second camera is termed a virtual camera. Thus the determination of the model is equivalent to defining the geometrical connection (transformation) between the two views; which in turn is equivalent to VP estimation in a camera-mirror case. Shafer in ref. [72] points out that an object and its cast shadow share a similar geometrical relationship to that found in the camera-mirror case. The approach we introduce therefore applies to the cast-shadow case as well; and we describe practical results using input from a real-life video test-sequence which demonstrate the applicability of our method in this situation.

Since the determination of the model is equivalent to defining the transformation between the original and the virtual views in camera-mirror case, this point-to-line transformation may be determined by using corresponding point pairs in the two views.

2.2.1. Mirror pole in camera-mirror scenes

This section introduces the mathematical description of the geometric model related to a camera-mirror scene. The properties will however be derived using an algebraic approach, rather than a geometric one [26]. The notations and interpretations that we use are based on the published book of Hartley and Zisserman [24].

Figure 2.1 shows in diagrammatic form a common case of a reflective surface (e.g. a mirror), denoted by Ω, which lies in the (x-y) plane (right-handed system). *C* denotes the camera center, and the image plane is denoted by Π (3-D points are mapped to this plane via central projection). The uppercase bold letters (e.g. $\mathbf{X}$) denote 3-D point coordinates (in vector form), the elements of which will be denoted by $\mathbf{X} = [x_1, x_2, x_3]^T$. The lowercase bold letters are the 2-D points (in vector form) on the camera plane. Note that these coordinates are not homogenous. The homogenous vectors are designated by an overbar, e.g. $\tilde{\mathbf{X}}$, which corresponds to $\mathbf{X}$ according to the transformation $\mathbf{X} \to \tilde{\mathbf{X}} = [x_1, x_2, x_3, 1]^T$, while the reverse transformation is given by $\tilde{\mathbf{X}} \to \mathbf{X} = [\tilde{x}_1 / \tilde{x}_4, \tilde{x}_2 / \tilde{x}_4, \tilde{x}_3 / \tilde{x}_4]^T$.

DOI:10.15774/PPKE.ITK.2024.006

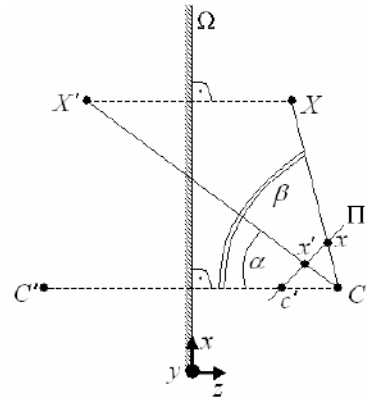


Figure 2.1: Simple model for reflective surface: $\mathbf{C}=[c_1,c_2,c_3]^T$ is the camera center and Π is the plane for central projection (image plane). An arbitrary 3-D point $\mathbf{X}$ has a virtual point pair because of the mirror plane (Ω), and termed by $\mathbf{X}'$, likewise $\mathbf{C}'$. Consistently, the 2-D points in the image are $\mathbf{x}$, $\mathbf{x}'$ and $\mathbf{c}'$.

Without loss of generality, in the following relationships we assume that the original points lie on the positive side of the z axis, thus the third coordinate is always positive for all original points (e.g. $c_3>0$). In the diagram the two angles α and β are included angles of 3-D vectors, defined by $\alpha=\overrightarrow{\mathbf{CC}'}\sphericalangle\overrightarrow{\mathbf{CX}'}$ and $\beta=\overrightarrow{\mathbf{CC}'}\sphericalangle\overrightarrow{\mathbf{CX}}$.

In our model the camera is a general projective camera [24]. The matrix P denotes the camera which maps world points $\mathbf{x}$ to image points $\mathbf{x}$ according to:

$$\tilde{\mathbf{x}}=P\tilde{\mathbf{X}} \tag{2.4}$$

Note that the camera center is the 1-dimensional right null-space $\mathbf{C}$ of P:

$$P\tilde{\mathbf{C}}=0 \tag{2.5}$$

Furthermore, we introduce the notation that the columns of P are $\mathbf{p}_i$:

$$P=\begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix}=\begin{bmatrix} \mathbf{p}_1^T \\ \mathbf{p}_2^T \\ \mathbf{p}_3^T \end{bmatrix} \tag{2.6}$$

The effect of mirror Ω may be described by the following coordinate transformation:

$$\mathbf{X}'=M\mathbf{X} \tag{2.7}$$

where M is a 3x3 matrix or a 4x4 matrix in case of homogenous coordinates:

DOI:10.15774/PPKE.ITK.2024.006

$$M=\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & -1\end{bmatrix} \text{ and } \tilde{M}=\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & -1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix} \tag{2.8}$$

This transformation is the reflection in the (x-y) plane which operates only on the z coordinate. Thus, the image point of the virtual point $\mathbf{X}'$ generated by the reflection is

$$\tilde{\mathbf{x}}'=P\tilde{\mathbf{X}}'=P\tilde{M}\tilde{\mathbf{X}} \tag{2.9}$$

We start the algebraic derivation of the model similar fashion to the work of Xu and Zhang [32]. The ray back-projected from $\mathbf{x}$ by P is obtained by solving (2.4). The solution is given as a 3-D line in parametric form (the ray is parametrized by the scalar λ):

$$\mathbf{X}(\lambda)=P^{+}\mathbf{x}+\lambda\mathbf{C} \tag{2.10}$$

where P^{+} is the pseudo inverse of P (i.e. $PP^{+}=I$). The epipolar line is the line joining the projections of two reflected points: $\mathbf{C}'$ and $\mathbf{X}'$. These projected points are expressed by using (2.7) and the equation of the epipolar line is determined by the cross product:

$$\mathbf{l}=\left(P\tilde{M}\tilde{\mathbf{C}}\right)\times\left(P\tilde{M}P^{+}\tilde{\mathbf{x}}\right)=F\tilde{\mathbf{x}} \tag{2.11}$$

where F is the fundamental matrix. This formula defines a point-line map, thus the F may be expressed by

$$F=\left(P\tilde{M}\tilde{\mathbf{C}}\right)\times\left(P\tilde{M}P^{+}\right)=\left[\tilde{\mathbf{c}}'\right]_{\times}P\tilde{M}P^{+} \tag{2.12}$$

where the notion $\left[\mathbf{a}\right]_{\times}$ in general form is defined by [24] as follows:

$$\left[\mathbf{a}\right]_{\times}=\begin{bmatrix}0 & -a_3 & a_2\\ a_3 & 0 & -a_1\\ -a_2 & a_1 & 0\end{bmatrix} \tag{2.13}$$

Thus, the cross product is related to skew-symmetric matrices according to the equivalence [24]:

$$\mathbf{a}\times\mathbf{b}=\left[\mathbf{a}\right]_{\times}\mathbf{b} \tag{2.14}$$

DOI:10.15774/PPKE.ITK.2024.006

Model properties

In the next subsections two main properties of camera-mirror scene geometry will be discussed. These properties play an important role in the subsequent sections: model parameter estimation and foreground classification.

Property 1. *The fundamental matrix corresponding to the original image and the virtual image in a camera-mirror scene is of the form* $F=[\mathbf{c}']_\times$, *where* $\mathbf{c}'$ *is the vanishing point (VP). Consequently, F has 2 degrees of freedom and is identified with the VP.*

Proof. The key step is the expression of $\mathbf{x}'$ in terms of $\mathbf{x}$ based on (2.4) and (2.9) thus:

$$\tilde{\mathbf{x}}' = P\tilde{M}\tilde{\mathbf{X}} = P\tilde{\mathbf{X}} - 2\mathbf{p}_3 x_3 \tag{2.15}$$

By substituting this formula into (2.12) and utilizing the definition of the camera center (2.5), F may be written as

$$F=\left(P\tilde{M}\tilde{\mathbf{C}}\right)\times\left(P\tilde{M}P^{+}\right)=\left(P\tilde{\mathbf{C}}-2\mathbf{p}_3c_3\right)\times\left(PP^{+}-2\mathbf{p}_3\mathbf{p}_3^{+T}\right)=-2c_3\mathbf{p}_3\times\left(I-2\mathbf{p}_3\mathbf{p}_3^{+T}\right) \tag{2.16}$$

The next step employs the following relationship:

$$-2c_3\mathbf{p}_3\times\left(2\mathbf{p}_3\mathbf{p}_3^{+T}\right)=-4c_3\mathbf{p}_3\times\mathbf{p}_3\mathbf{p}_3^{+T}=-4c_3\mathbf{0}\mathbf{p}_3^{+T}=\mathbf{0} \tag{2.17}$$

Substituting this into (2.16) the final formula for F emerges as:

$$F=-2c_3[\mathbf{p}_3]_\times=[\mathbf{c}']_\times \tag{2.18}$$

From this formula we see that F is skew-symmetric and is formed from the VP. □

Note that the layout of F is similar to (2.13):

$$F=\begin{bmatrix}0 & -1 & c_2'\\ 1 & 0 & -c_1'\\ -c_2' & c_1' & 0\end{bmatrix} \tag{2.19}$$

In case of skew symmetric matrix F the fundamental constraint may be transformed into the collinearity constraint: namely that the points $\mathbf{x}$, $\mathbf{x}'$ and $\mathbf{c}'$ lie on a common line. This follows directly from the rewritten form of the fundamental constraint:

$$\tilde{\mathbf{x}}_1^T F\tilde{\mathbf{x}}_2=\tilde{\mathbf{x}}_1^T\left(\tilde{\mathbf{c}}'\times\tilde{\mathbf{x}}_2\right)=\left\langle\tilde{\mathbf{x}}_1,\tilde{\mathbf{c}}'\times\tilde{\mathbf{x}}_2\right\rangle=0 \tag{2.20}$$

where $\tilde{\mathbf{x}}_1$ and $\tilde{\mathbf{x}}_2$ are an arbitrary corresponding point pair ($\times$ and $\langle .,. \rangle$ denote the cross product and the dot product, respectively.). In this formula the homogenous

DOI:10.15774/PPKE.ITK.2024.006

forms of the vectors are used. Because of the fact that the cross product of the vectors expresses the equation of a straight line through these two points, and furthermore that the dot product is a simple substitution into this line equation, the whole expression becomes zero when the third point lies on the line defined by the two points.

In summary, the first property states that the geometric model of a camera-mirror scene can be defined with a 2-D point, namely with the vanishing point which is the parameter of the model.

Property 2. *From a given corresponding point pair, the nearest point to the VP is the reflection of the second point.*

Proof. This statement may seem obvious, but nevertheless we shall give a short proof. The statement is equivalent to

$$\|\mathbf{c}' - \mathbf{x}\| > \|\mathbf{c}' - \mathbf{x}'\| \tag{2.21}$$

where $\|\mathbf{x}\|$ denotes the Euclidian length of vector $\mathbf{x}$. The simplest form can be derived based on the angles indicated in Figure 2.1: $\alpha < \beta$. It can be justified by using some elementary linear algebra. □

The importance of this property lies in the fact that knowledge of the position of the VP makes it possible to decide whether a given point is truly the reflection of another point, i.e. whether they form a corresponding point pair.

2.2.2. Light direction in case of cast shadow

For most outdoor situations, the direction of daylight shadows is controlled by the position of sun. Because the rays of sunlight are essentially parallel, they converge in an infinite vanishing point.

DOI:10.15774/PPKE.ITK.2024.006

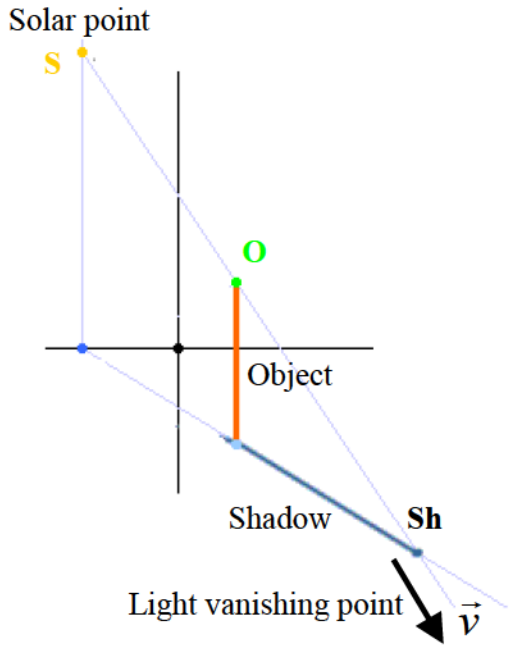


Figure 2.2: Geometry of shadow: the three points; object (original), shadow and vanishing point are collinear. Because in outdoor cases the distance of the light source from the object casting the shadow is near infinity, the knowledge of a common direction ($\vec{v}$) is sufficient instead of position of vanishing point.

Because of the far vanishing point the geometrical model may be simplified; the knowledge of a direction (2D vector) is enough for the estimation of shadow region. Details about the geometry can be found in [72]. In our previous work we have described a method to compute the vanishing point in case of camera-mirror setting using motion statistics [1].

The importance of knowledge of the light vanishing point lies in the fact that it simply enables the integration of geometrical constraint into the shadow detection process. Obviously, the shadow point must lie on the line going through the original point with direction $\vec{v}$.

Note that, $\vec{v}$ has unit length: $\|\vec{v}\| = 1$. Unfortunately, this geometrical model is not enough for the exact determination of the corresponding shadow point, because it is a point-to-line transformation instead of a point-to-point analogy. Accordingly, we will use this extra knowledge together with other features (colour, motion etc.) to achieve a better classification results.

DOI:10.15774/PPKE.ITK.2024.006

2.3. Vanishing line parametrization

Parallel planes in a 3-dimensional space intersect a plane at infinity in a common line, and the image of this line is the horizontal vanishing line, or horizon. Geometrically the vanishing line is constructed, by intersecting the image with a plane parallel to the scene plane through the camera caneter. The vanishing line (VL) depends only on the orientation of the camera. The following three examples demonstrate the usefulness of horizon [24]:

- The plane's orientation relative to the camera may be determined from its vanishing line.
- The plane may be metrically rectified given only its vanishing line.
- The angle between two scene planes can be determined from their vanishing lines.

Thus, the vanishing line is useful for camera orientation and extrinsic parameter determination [45]. A common way to determine the vanishing line of a scene plane is first to determine the vanishing points for two sets of lines parallel to the plane, and then to construct the line through the two vanishing line.

For still images [46], it can be successfully determined only when there are detectable parallel lines; and in image-sequences, only when certain assumptions are satisfied which enable us to detect and track known objects [45]. In summary, most of the published still-image based methods are unsuitable for processing the images of a typical surveillance scene. Furthermore, in typical surveillance scenes of public places the assumptions on which the video-based methods are posited are not satisfied.

In summary, the determination of the vanishing line is possible with knowledge of at least two vanishing points (these lie in the VL); thus three corresponding line segments (e.g. derived from the height of a given person in the image), or else known parallel lines in the same plane, are necessary.

DOI:10.15774/PPKE.ITK.2024.006

3.Extraction of motion characteristics

The use of still images for the extraction of correspondences is limited. This is because the additional information about the scene provided by image sequences is lost. Hereby the implementation of classification and feature extraction tasks are more complicated. Then again the scene dynamics provide useful information without using appearance based image analysis or matching.

In this chapter we show that, the human motion and the co-motion statistics can be extracted robustly from video sequences. This section presents a walk detection method using non-linear classification and a model based approach for the processing of motion statistics. These methods extract the basic spatial (2D) information for the introduced geometrical model computation.

DOI:10.15774/PPKE.ITK.2024.006

3.1. Detection of gait characteristics in videos

The main aim of the section is to present a method for the detection of human walking in videos and for the extraction of gait features. Our feature extraction method, which is based on method proposed in [2], utilizes extended third-level symmetries of the edge map to detect and track structural changes of moving objects in video sequences. In [2] we introduced a novel walk detection algorithm in stable experimental conditions that is able to detect pedestrians by recognizing their characteristic symmetry patterns, using Kernel Fisher Discriminant Analysis (KFDA technique).
This method is based on detecting the moving leg pairs and developing the symmetry based approach in [2] for more robust cases: independence of noise and the varying frame rate [58]. We apply an invariant and effective data representation in the Eigenwalk space, based on spline interpolation and a dimension-reduction technique. Here we present a more established pattern classification method based on the continuous interpolation of the symmetry patterns. A more robust classification is carried out via Support Vector Machine (SVM) with Gaussian kernel function.
For testing the efficiency and the robustness of the above feature extraction method it has been applied for camera registration. The features we used (concurrent walk-steps, and leading-leg identity) seem to be beneficial to provide data (matching points) for the estimation of transformation between two different camera views of the same scene. The accuracy of the registration results proves the usefulness of detected features.

The basis of our algorithms is the ability to detect human movements. The main steps of the algorithm are

- Background subtraction, change-detection
- Edgemap detection and symmetry computation (first level)
- Extension of symmetry computation up to three levels (L3S)
- Temporal tracking using reconstructed masks

Samples of the image processing steps are shown in Figure 3.1, which illustrates the results of the algorithmic procedures up to the stage of symmetry pattern extraction from the reconstructed masks.

DOI:10.15774/PPKE.ITK.2024.006

The assumptions we use are

- The camera is in arbitrary static position.
- The image motion can be from more than one person.
- The image capture rate is at least 10fps.
- The height of each "target" person is at least 100 pixels.
- Leg-opening is visible in most cases.

3.1.1. Walk detection in EigenWalk space

Symmetry pattern extraction

Symmetry is a basic geometric attribute, and most objects have a characteristic symmetry-map. This unique and invariant property leads to the applicability of symmetries in our approach for image-processing.

The medial axis is formally defined as the closure of the locus of centers of maximal spheres that are at least tangent to the surface at two places. The symmetry set is a related representation, which makes explicit more of the symmetries of the shape by removing the maximality condition [80]. The Generalized Symmetry Operator [90] applies distance weight function (affected by spacing), a phase weight function (affected by edge direction), and a logarithmic mapping of points' intensity. A large variety of numerical techniques have been developed to extract medial axis from a 2D shape [80]; thinning methods, which iteratively peel off the surface in the discrete domain while maintaining object topology. The skeleton is a linear pattern representation that is generally recognized as a good shape descriptor. An effective implemented skeleton algorithm by using binary pyramid is called multiscale skeleton [78]. These techniques require a fully segmented, and connected object. The availability of efficient distance transform algorithms has led to ridge-following algorithms that view the medial axis as ridges of the distance map. Another class of methods casts the surface as the level set of an embedded object and finds the weak solutions of a PDE which models the wave propagation process whose singularities yield the medial axis.

DOI:10.15774/PPKE.ITK.2024.006

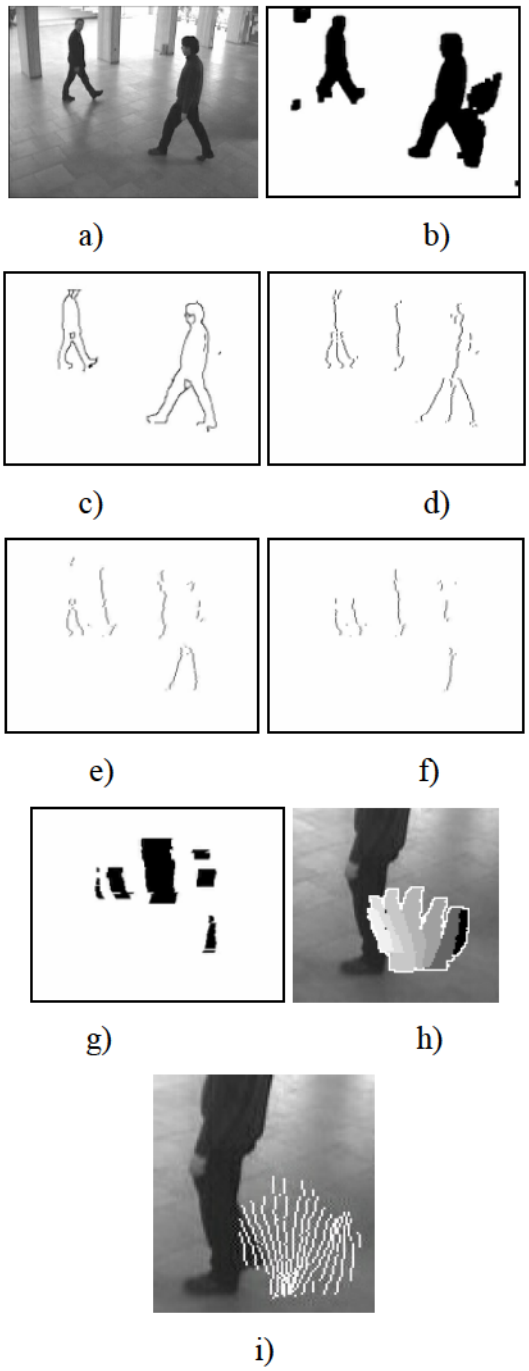


Figure 3.1: Overview of feature extraction steps: a) Image from input sequence. b) Result of change-detection. c) Filtered Canny edge map. d) First level symmetries. e) Second level symmetries. f) Third-level symmetries (L3S). g) Reconstructed masks from symmetries. h) Tracking, showing coherent masks in the sequence (of 7 frames). i) Symmetry pattern (of 25 frames).

DOI:10.15774/PPKE.ITK.2024.006

Background subtraction

An elementary method to reduce the computation cost of methods using motion information derived from static-position cameras is background subtraction (or change-detection), that is, remove all but what are important artifacts, see Figure 3.1(b). Implementing a more sophisticated method for background subtraction is a challenging task [31] [30]. We have therefore implemented a mixed solution; the algorithm can be selected from one of two methods: either a simple running as default (as used in Figure 3.1(b)); or a Gaussian-mixture model [30] in case of noisy outdoor scenes. In our trials, most of the problems were caused by shadows. For shadow removal we used the method of [31], which is a modification of the well known SAKBOT algorithm [63].

Horizontal symmetry extraction

Our symmetry detection method [2] is based on the use of morphological operators to simulate spreading waves from the edges. In that pedestrian detection approach, only horizontal morphological operators are used to extract the symmetries.

The mathematical definition of this kind of symmetry extraction is similar to skeleton algorithm [64] with distance definition of taking into account only the x (horizontal) coordinates instead of Euclidean distance. Thus, it is possible to recover the original edge map given its symmetry map and the distance of each symmetry point to its edge point [64].

As illustrated in Figure 3.1, the symmetry concept can be extended by iterative operations. The symmetry of the Level 1 symmetry map (Figure 3.1(d)) is the Level 2 symmetry (Figure 3.1(e)); and the symmetry of the Level 2 map is the Level 3 symmetry (L3S), as shown in Figure 3.1(f).

The symmetry axis describes well global and local structural properties of an object (even non-rigid). Higher order symmetries are used to describe local structure reflecting the overall complexity of an object. L3S is a representative feature of objects having two coherent objects with two parallel edges. Typically, humans' legs are such objects. Thus, L3S can be used to indicate them.

Our symmetry-extraction method is less sensitive to edge fragmentation than the original "skeleton" method is. Nevertheless, the L3Ss contain an accumulation of fragments from the preceding symmetry levels. To reduce this error we use vertical

DOI:10.15774/PPKE.ITK.2024.006

morphological closing operators at each level of processing. In addition, it is an important factor when the objects are small and near to one another on the image. The vertically-oriented kernels help to avoid possible confusion with nearby neighbouring symmetries. In contrast to the horizontal method the circle-based spreading produces several small and overhanging axes which lead to unusable higher level symmetries as illustrated in Figure 3.2.

Figure 3.2: The Level 1 and Level 2 symmetry maps derived using 2D wave spreading (not optimal circle).

Figure 3.3 shows the steps of the algorithm where *Input* means the binary input image (Canny edge map), *Symmetry* means a binary image with the detected symmetries, *SizeY(I)* and *SizeX(I)* mean the size of the given image *I* and *I(x,y)* means the pixel of image *I* in x column and y row positions.

```
FOR y=1 TO SizeY(Input) {
PreviousX=-1;
FOR x=1 TO SizeX(Input) {
                        if (Input(x,y)>0)       {
                                if (PreviousX<>-1) {
                                        Symmetry((x+PreviousX)/2, y)=1;
                                }
                        PreviousX=x;
                        }
                }
        }
```

Figure 3.3: The simplified symmetry extraction algorithm on binary images

This effective algorithm scans along the lines in the edge map and places a pixel to the symmetry map between two edge pixels. It is able to calculate one iteration on the symmetry levels so it should run three times as the resulting symmetry map contains

DOI:10.15774/PPKE.ITK.2024.006

the L3S. With the implemented algorithm we achieved a processing speed of 80 frame/sec at 320x240 resolution on a 2.4GHz Pentium CPU.

Temporal tracking

In general, the image may contain a number of symmetry-samples, which have arisen from errors in change-detection or from the complexity of the background; for examples, see Figure 3.1(f). However, even the existence of perfect symmetries in a single static image does not necessarily provide usable information about the image-content; for this, we track the changes of the symmetry samples by using temporal comparisons. We have implemented an effective tracking method using masks around the symmetries. The algorithm generates this mask around the L3S samples from their radii; such masks can be seen in Figure 3.1(g). This generation procedure is similar to reconstruction process in skeletonization [64].

The first L3S appears when the legs are opening and the last is detected just before the legs are closed; so a symmetry-pattern of a walking person's step corresponds to the movement of the legs from opening to closing. The detected L3S symmetries are filtered by their size, only symmetries 3 pixels long or longer are processed. In the following descriptions, one cycle denotes two steps. During tracking, the algorithm calculates the overlapping areas between symmetry masks in successive frames, and then constructs the symmetry patterns of the largest overlapping symmetries frame by frame as the series of symmetry samples. An overlapping mask sequence can be seen in Figure 3.1(h), and the symmetry pattern in Figure 3.1(i). The advantage of this simple algorithm is that it is tracking the complete leg movement and the associated structural changes, instead of just tracking selected feature points on the image by means of some optical correlation method. This inherent feature of the method increases the stability and the robustness of the results in cases where the edges of the target are partially "damaged" in some frames.

In our sample videos, we found that the most critical factor is the image refresh rate: we found that the rate of at least 10 frames/second is required.

Detection of walk patterns

The extended symmetry feature gives a specific pattern when it is tracked through the frames of 1-2 walking steps. However, it differs from other methods dealing with

DOI:10.15774/PPKE.ITK.2024.006

periodicity analysis [51][82] that we do not consider any specific structure in time or in space. Moreover, we can definitely differentiate between the symmetry pattern of walking legs and that of other parts, e.g. arms and head.

Representation and re-sampling

The extracted symmetry patterns are represented with the upper and lower end points (2 each) of the L3S in each frame, see Figure 3.4. Thus there are four 3D (space and time) coordinates, which correspond approximately to the "end-points" of the two legs.

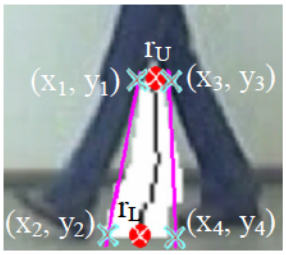


Figure 3.4: The end points used to define symmetries for the re-sampling and classification tasks.

Temporally these patterns depend both on the frame rate and the walking speed, so a pattern usually contains data from 5-30 frames, see Figure 3.5. All the symmetries composed of four or fewer frames are filtered out and not classified, because they are usually produced by noisy backgrounds. Before any further analysis, the data is normalized with respect to time for presenting an invariant description of the motion; we perform this task with Bezier spline interpolation [65]. This technique has the advantage that it performs two tasks: (i) data is re-sampled in a defined time interval with a fixed-point count; (ii) noise-filtering is performed on the trajectories, which results in a smoother symmetry pattern. The noise-cleaning is critical because in real scenes these patterns are often damaged, see Figure 3.5. The Bezier spline (B-B spline) [65] is a good choice because the effect of base points is global; so the presence of some damaged points, coming from erroneous symmetry extraction and unstable video frame rate, does not cause significant change in the whole trajectory.

DOI:10.15774/PPKE.ITK.2024.006

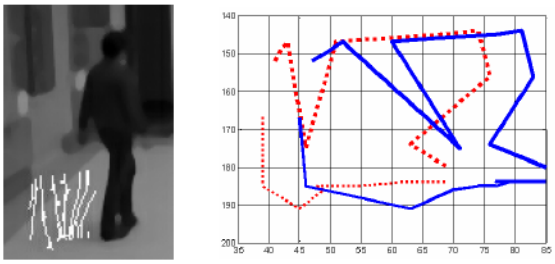


Figure 3.5: Original symmetry pattern and the trajectories of 9 frames. The four curves (trajectories) are the upper and lower – both left and right - end points of the symmetry sample expanded with its radius. The input contains severely corrupted data.

This time-extended data representation permits the integrated analysis of data obtained from several cameras where the frame rates are different and unstable (e.g. network cameras); the extracted features must be resampled with a continuous time-division. The result of Bezier spline interpolation of data can be seen in Figure 3.6. The current implementation generates 100 interpolated points of both coordinates of every end point. These points are termed with the following vectors (each has dimension of 100) $\overline{x}_1, \overline{y}_1, \overline{x}_2, \overline{y}_2, \overline{x}_3, \overline{y}_3, \overline{x}_4, \overline{y}_4$.

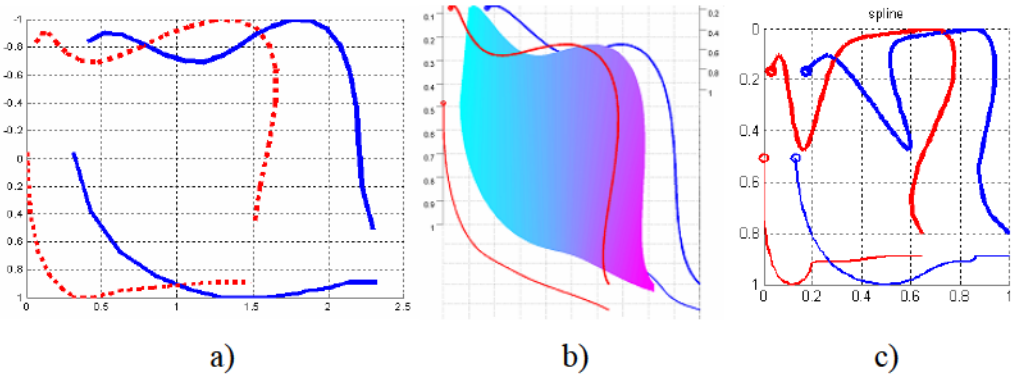


Figure 3.6: Interpolated trajectories of 100 points by using B-B splines (a) and B-splines (c) and the numerically integrated surface (b) of the pattern defined by eq. (3.3). (Input data is the same as for Figure 3.5.) The surface is formed from the interpolated upper and lower end points of symmetries which represents the height of the visible area of leg-opening.

A popular curve interpolation method is that known as NURBS (Non Rational B-Spline). This is an extended B-spline method, and the base points have local effect. However, for our application it has drawbacks: its computation cost is higher than for the Bezier-spline method, and the results may be less good. The results of an 8th-order B-spline processing, on the same input data as before, are shown in Figure 3.6(c). Note that the damaged data gives a less smooth output than the previous method.

DOI:10.15774/PPKE.ITK.2024.006

Dimension reduction

The interpolated 3D (XYT) points are rearranged into a row-vector with dimension of 800 (because it is the concatenation of the eight vectors of coordinates):

$$\tilde{x} = [\overline{x}_1, \overline{y}_1, \overline{x}_2, \overline{y}_2, \overline{x}_3, \overline{y}_3, \overline{x}_4, \overline{y}_4] \tag{3.1}$$

The linearity of the time coordinate makes a smooth time-division (time is linearly related to the successive samples thanks to the re-sampling). Consequently, we can omit this coordinate; it has no discriminative information content. After we center the patterns for both the x and y, both coordinates are normalized using a constant chosen such that max(y)= 1 and min(y)= –1; we do this because we have found that the y-size of the patterns varies less than does the x-size. We do not normalize with individual coefficients for x and y, since in that case the information content of the ratio of x and y values would be lost.

A well-known technique for dimension reduction is the PCA method [66]. To find the principal components of the distribution of the feature space we first obtain the mean and the covariance matrix (Σ) of the data set. Then we can compute $N \leq \text{rank}(\Sigma)$ nonzero eigenvalues and the associated eigenvectors of Σ based on SVD. The eigenvectors associated with a small number of the largest eigenvalues correspond to large changes in training patterns; thus a transformed matrix can be constructed from eigenvectors to project the original data into a parametric eigenspace with a drastically reduced number of dimensions. We considered the space spanned by the 3 most significant eigenvectors of the covariance matrix of the interpolated data set that account for 93% of the variation in the input space: we call this the Eigenwalk space. The associated eigenvectors form the eigenspace transformation matrix.

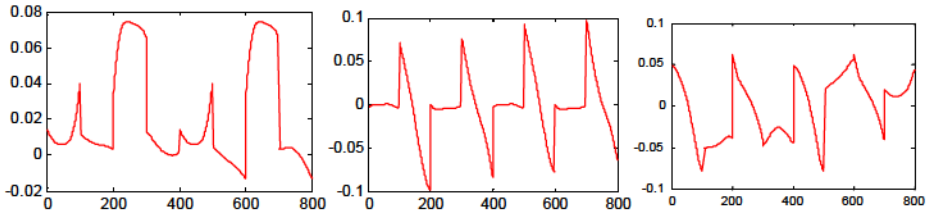


Figure 3.7: The first three eigenvectors obtained by PCA training.

From Figure 3.7 we can see that these eigen-walks are periodic, which reveals the construction method of raw data. Furthermore we can determine that the dominant information is the horizontal (x) directional motion of lower end points (first eigen-

DOI:10.15774/PPKE.ITK.2024.006

vector) and walk has a characteristic symmetry on the vertical (y) directional motion of end points (second eigen-vector).

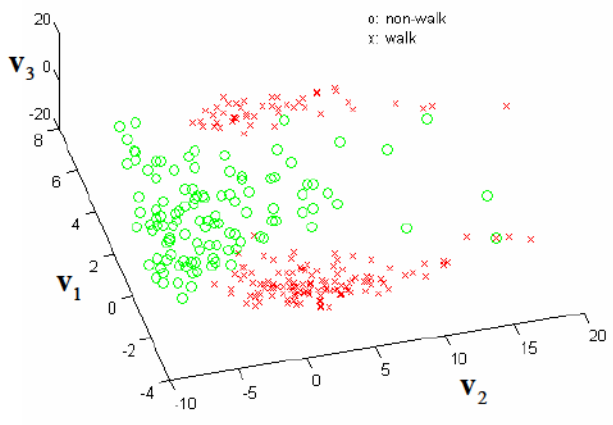


Figure 3.8: "Walk" and "non-walk" patterns in the eigenspace. Where $\mathbf{v}_1$, $\mathbf{v}_2$ and $\mathbf{v}_3$ are first three the eigen-vectors.

Figure 3.8 demonstrates the results using the test-set of labeled "walk" and "non-walk" symmetry patterns. This drastically reduced number of dimensions greatly assists in increasing the classification speed, which is an important factor in real-time applications.

Non-linear classification of symmetry patterns

Level 3 symmetries can also appear in other parts of the image, not only between the legs; and the tracking method also collects all of these related symmetries, see Figure 3.1(f). Walk patterns lie on a non-linearly shaped manifold in the eigenspace, see Figure 3.8. The classification process is carried out via non-linear method, namely Support Vector Machine (SVM) [67] with radial basis kernel function:

$$k(\overline{x},\overline{x_i}) = e^{-\frac{\|\overline{x}-\overline{x_i}\|^2}{2\sigma^2}} \tag{3.2}$$

The training data set, assembled from indoor video sequences, contained 750 "walk" and 14200 "non-walk" patterns in the eigenspace. The parameter (σ) was determined in the interval 0.1-6.0; from this an optimal value is 0.4 (see Figure 3.9) where the valid classification rate is 93.8% on the training set with 217 support vectors. Our main goal was to reliably detect human movements, but at the same time with a false-positive (5.2%) detection rate as small as possible (the false-negative rate was 1.0%).

DOI:10.15774/PPKE.ITK.2024.006

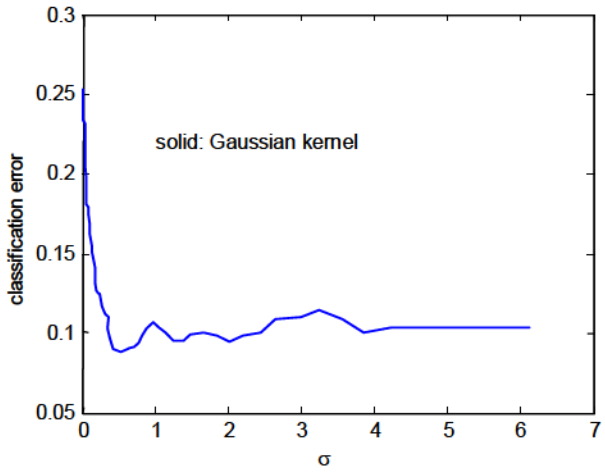


Figure 3.9: Relation between the kernel parameter and the classification error rate for the Gaussian kernel.

Figure 3.10 shows a typical surveillance video shot, demonstrating the detected L3S patterns containing noises and real walking patterns. For similar image sequences the number of detected L3Ss is about 10-40, from which real walking patterns are about 1-5 with the above detection ratio.

a) b)

c)

Figure 3.10: Representative indoor shot: a) L3S, b) output of tracking c) detected walk patterns.

DOI:10.15774/PPKE.ITK.2024.006

3.1.2. Identification of leading leg

According to our terminology, the leading leg is the "standing" leg, which at that instant carries the person's weight (see Figure 3.11(b) and (c)). In this section we present a method to determine, from one detected walk cycle (two consecutive steps), whether the leading leg is the right or the left leg by estimating 2D direction of walk and the "ratio" of consecutive walk patterns.

The 2D motion vector on the image-plane, and the walker's gait-period, can be extracted directly from the detected patterns: we estimate the motion vector by fitting a regression line to the last half-trajectory of the lower two points of the pattern.

The non-rigid human body during a walking cycle has a useful property, which assists us in recognizing the leading leg. Depending on the 3D walk-direction, and on which is currently the leading leg, one leg or the other practically obscures the visible area between the legs (Figure 3.11(b) and (c)).

a) b) c) d) e)

Figure 3.11: a) An image showing the location of the derived symmetry pattern (marked with white border; "x" marks a feature-point. b), c) Illustrations of our definition of "leading leg"; the "standing" or leading leg is the right leg in b), and the left leg in c) (legs highlighted manually). d), e) The detected patterns for the same steps as shown in b) and c); the 2D direction is bottom-left to upper-right (case 2 in Table 3-I).

During one cycle, the left leg and right leg in turn are in the leading position. The above-described method can detect one step. To connect two successive steps as one walk-cycle, we calculate the 2D displacement vector of a detected step, and then search for another step (walk pattern) in the estimated 2D position and at a time-point after a forecasted walk-period.

During a walk-cycle (two consecutive steps, see Figure 3.11(d) and (e)) the ratio of the visible leg-opening areas, together with the 2D direction on the image-plane, can be used to identify which is the leading leg. The visible leg-opening area is approximated by the surface defined by symmetries between the legs from consecutive frames. To measure the area between the legs, we used a numerical

DOI:10.15774/PPKE.ITK.2024.006

integral of the surface defined by the interpolated patterns (3.3) (see Figure 3.6(b)). The area of surface was approximated by dividing it into triangles and summing areas of triangles:

$$area \approx \sum_{i=1}^{n-1} \left(\begin{array}{l} \left\| (\vec{r}_U(i) - \vec{r}_L(i)) \times (\vec{r}_U(i+1) - \vec{r}_L(i)) \right\| + \\ \left\| (\vec{r}_U(i+1) - \vec{r}_L(i)) \times (\vec{r}_L(i+1) - \vec{r}_L(i)) \right\| \end{array} \right) \tag{3.3}$$

where n=100, the number of interpolated points and, r_U and r_L are the upper and lower midpoints of the interpolated patterns (see Figure 3.4):

$$\vec{r}_U(i) = \left[\frac{x_1(i) + x_3(i)}{2}; \frac{y_1(i) + y_3(i)}{2} \right], \vec{r}_L(i) = \left[\frac{x_2(i) + x_4(i)}{2}; \frac{y_2(i) + y_4(i)}{2} \right] \tag{3.4}$$

where the running index i is along the trajectories of the symmetry pattern.

Table 3-I summarizes the relationship between the leading leg and the ratio of surfaces from two successive patterns. A limitation of the described method is that it cannot identify the leading leg when the motion is parallel to the camera plane, since in such cases the areas are nearly equal (cases 3, 4 and 9, 10 in Table 3-I).

TABLE 3-I: SURFACE DEPENDENCIES ON 2D WALK-DIRECTION AND LEADING LEG.

Case	2D Dir	Leading Leg	Ratio
1	↗	Right	>1
2		Left	<1
3	→	Right	≈ 1
4		Left	≈ 1
5	↘	Right	<<1
6		Left	>>1
7	↖	Right	<1
8		Left	>1
9	←	Right	≈ 1
10		Left	≈ 1
11	↙	Right	>>1
12		Left	<<1

3.1.3. Experimental results

Careful implementation of the method with the new filtering and detection step resulted in 10-15 msec processing speed for a symmetry pattern, on a state of the art desktop PC. The number of extracted symmetries affects the speed of filtering damaged points (interpolation). Hence, the processing speed depends on cameras frame rate.

DOI:10.15774/PPKE.ITK.2024.006

During the test we used the frame sequences as captured, recognizing walking patterns real-time. We have tested our methods using test-inputs from both indoor and outdoor videos, where the following factors were varied: camera viewpoint, number of “targets”, and image-capture rate. These videos contained 420 steps and 150 walk-cycles in the indoor scenes, and 350 steps and 110 cycles in outdoor environments. Figure 3.12 and Figure 3.13 show sample results of symmetry pattern extraction in various videos. As it can be seen from results the algorithm performs well in case of very different lighting conditions, image quality, background and video frame rate (10, 15 and 25 FPS).

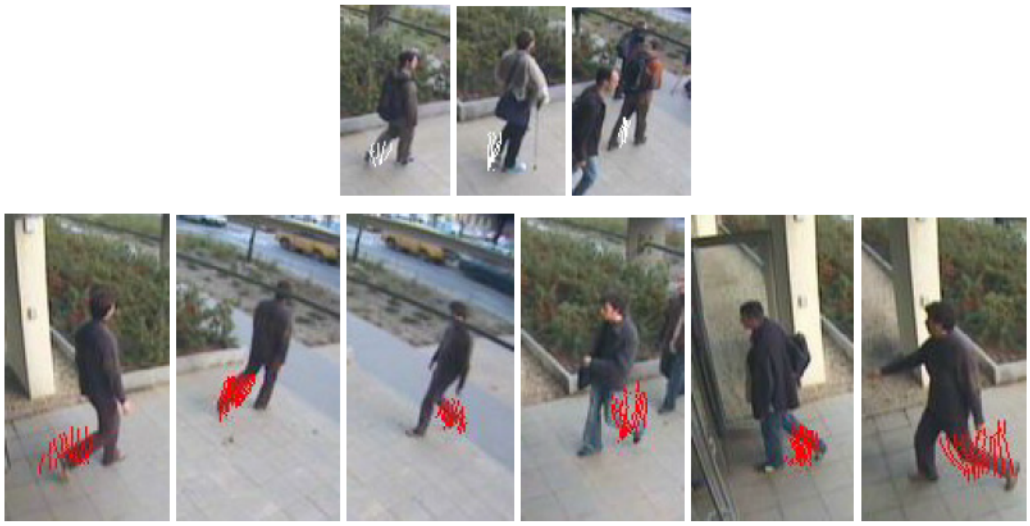


Figure 3.12: Detection of symmetry patterns in various outdoor videos.

DOI:10.15774/PPKE.ITK.2024.006

Figure 3.13: Detection of symmetry patterns in various indoor videos.

Figure 3.14: Detection of symmetry pattern in case of poor silhouette extraction (reflection on ground-plane causes error).

DOI:10.15774/PPKE.ITK.2024.006

Figure 3.15: Typical problematic cases illustrate the limitations of symmetry extraction and tracking methods: back-view, long coat, parallel overlapping and hidden legs.

There are several obvious limitations of the tracking algorithm. It is unable to detect direct front-view walks. Also, when the leading leg covers the rear leg, the symmetries do not appear. In Figure 3.15, we summarized some practical limitations of the symmetry tracking method.

We obtained a detection rate of 78.1% for outdoor and 89.5% for indoor videos, in cases where the leg motion (and the leg opening) was visible (detailed results in Table 3-II).

TABLE 3-II: EXPERIMENTAL RESULTS ON DETECTION OF WALK PATTERN

Method	Data set	Detection Rate	False-Positive	False-Negative
KFDA [8] (Gaussian kernel)	Training	89.2%	8.2%	2.6%
SVM (Gaussian kernel)	Training	93.8%	5.2%	1%
KFDA [8]	Indoor test	75.7%	15.3%	9%
SVM	Indoor test	89.5%	8.9%	1.6%
SVM	Outdoor test	78.1%	14.1%	7.8%
LDA	Test	66.4%	22.8%	10.8%

The method could not detect "near-frontal" human movements (motion directly toward camera, or nearly so), nor steps in two special cases where the walker is approaching "close" (Table 3-I, 5-6 and 11-12 cases), viz.: (i) top left to bottom right, and right leg leading; or (ii) top right to bottom left, and left leg leading. The leading leg identification worked well, with 99% correct identification in cases where the walk-cycle was detected correctly. This walk-detection procedure has been implemented in the camera-system of university campus.

DOI:10.15774/PPKE.ITK.2024.006

3.2. Statistical evaluation of video sequences

As with most "normal" multi-camera configurations, in the camera-mirror setting the transformation may be computed by using corresponding point pairs. A corresponding point pair may be defined as the knowledge of the position both of the original point, and of its transformation (in our case, its reflection). For a typical outdoor image, small objects (such as people) visible in the field of view of the virtual camera have textures which are low-detailed (or are missing texture altogether, for shadows); this is a principal reason why the extraction of correspondences is such a challenging task in this situation. In fact, for a static image the confident identification of the image of the virtual camera is impossible, without knowledge of the model. Fortunately however, the use of video sequences gives the possibility of the extraction of some additional information based on motion. The section describes a feasible method for correspondence extraction based on motion information determined from image-sequences.

The assumptions we use are:

- The camera is static in position
- The mirror is planar
- There is only one mirror in the image (and in case of shadows, only one, flat, ground-plane)
- There is only one point light source; the sun
- The imaging surface of the camera is not parallel to the mirror surface
- There is sufficient motion in the video sequence to generate reliable motion-statistics
- Nonlinear lens distortions can be neglected.

3.2.1. Test sequences

Video data for our tests was obtained in various environments, both indoor and outdoor. No video filtering for image enhancement was used. The capture rate was 20 fps for all the sequences. The indoor sequences were captured using digital cameras while analog surveillance camera was used for the outdoor videos. The test data is derived from four video sequences (see Figure 3.16). The “Ants” and “Mice”

DOI:10.15774/PPKE.ITK.2024.006

sequences are recordings of indoor experimental situations containing a large reflective surface; the first contains numerous point-like objects (ants), while in the second there are relatively large moving objects (two mice). The other two test inputs, the "Shop" and the "Shadow" videos, are typical outdoor surveillance sequences. The "Shop" video presents a challenging situation: only a small part of the reflective surface is visible; and the mirror is positioned far from the motion paths, so that the VP is nearly at infinity. The results of cast-shadow geometrical modeling will be demonstrated using the "Shadow" sequence. Table 3-III summarizes the main properties of these video sequences.

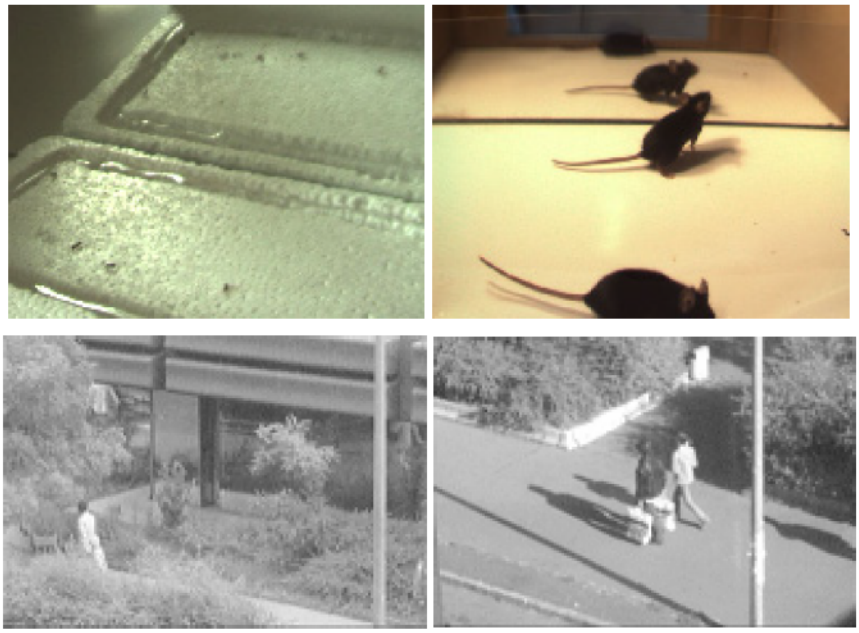


Figure 3.16: Frames of the test videos: "Ants", "Mice", "Shop" and "Shadow" videos

TABLE 3-III: TEST SEQUENCES

Name	Length (frames)	Resolution (pixel)	Description
Ants	140249	640x480	Contains point-like moving objects, with a plane mirror
Mice	25564	640x480	Contains relatively large moving objects, with a plane mirror.
Shop	200476	320x240	Only a small part of the reflective surface is visible.
Shadow	70185	320x240	Cast shadow is visible on the groundplane.

DOI:10.15774/PPKE.ITK.2024.006

3.2.2. Introducing co-motion statistics

Our correspondence-detection method is based on processing of the so-called co-motion statistics [7]. These statistics have been successfully used for image registration in case of wide-baseline camera pairs.

Briefly, the co-motion statistics of a point is a descriptor of spatial correlations to the other image-points. Its algorithmic implementation is executed with the temporal integration of motion masks; which leads to an approximation of the co-motion statistics. Here the statistics are collected from only one camera (in contrast to the situation in [7]), which gives us immediate accessibility to the *local co-motion statistics*. These statistics come from the temporal summation of the binarized motion masks; these are $m_t(\mathbf{x})$ where t is the frame position and the 2-D vector $\mathbf{x}$ is the point in the image. Thus:

$$m_t(\mathbf{x}) = \begin{cases} 1, & \text{where change is detected in point } \mathbf{x} \\ 0, & \text{otherwise} \end{cases} \tag{3.5}$$

We introduce the notation of the true foreground mask, (this is available only theoretically, because no algorithm is able to detect the foreground without errors in every situation):

$$\hat{m}_t(\mathbf{x}) = \begin{cases} 1, & \text{where object is moving in point } \mathbf{x} \\ 0, & \text{otherwise} \end{cases} \tag{3.6}$$

For the sake of later simplification we also introduce the notation S, the set of pixels in the image: $S = \left\{ \mathbf{x} \middle| \mathbf{x} \in \mathbb{R}^2 \text{ and } \mathbf{x} \text{ is in the image} \right\}$. The global motion statistics, which is the motion intensity, is formalized with the relationship:

$$P_g(\mathbf{x}) = \frac{\sum_t m_t(\mathbf{x})}{\Delta t} \tag{3.7}$$

(Because of the discrete time steps, Δt denotes the frame count of the processed sequence.) The global motion statistics of sample videos are demonstrated in Figure 3.17.

DOI:10.15774/PPKE.ITK.2024.006

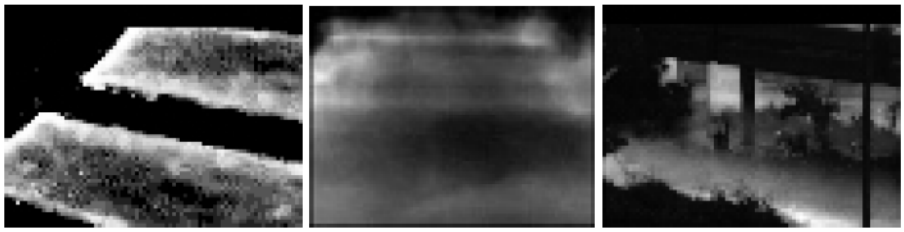


Figure 3.17: Global motion statistics: "Ants", "Mice" and "Shop". The lighter is the higher motion frequency.

The concurrent-motion probability of an arbitrary image-point $\mathbf{u}$ with another image-point $\mathbf{x}$ may be defined with the following conditional-probability formula:

$$f(\mathbf{u},\mathbf{x}) = P_{co}(\mathbf{u}|\mathbf{x}) = \frac{\sum_t m_t(\mathbf{x}) m_t(\mathbf{u})}{\sum_t m_t(\mathbf{x})} \tag{3.8}$$

In the implementation the above-defined $f(.)$ after normalization (i.e. $\sum_{\mathbf{u}\in S} f(\mathbf{u},\mathbf{x}) = 1$) as a 2-D discrete PDF (probability distribution function) is assigned to every pixel in the image. It follows that there will be local maxima (peaks) in the probability-maps in positions where motion was often detected concurrently. Sample statistics can be seen in Figure 3.18.

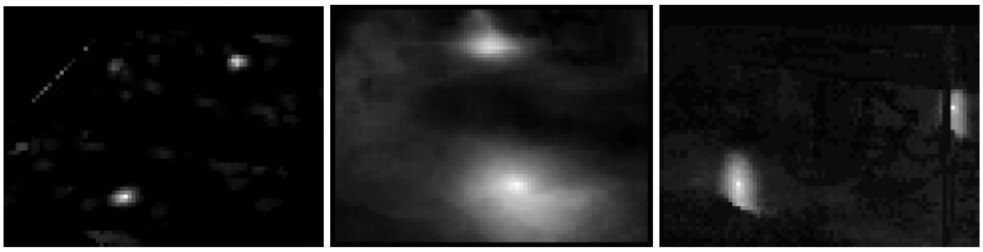


Figure 3.18: Samples from co-motion statistics, for "Ants", "Mice", and "Shop" sequences.

Because of the circumstance that in the currently implemented algorithm the co-motion statistics are attached to every point in the image as a 2-D array, the resolution of images must be reduced in order to restrict the required memory usage to a practical figure. For the original image resolution (e.g. 640 by 480 pixels) the memory requirement would be unmanageably large (hundreds of Gigabytes); consequently, a reduced 80x60 pixels image resolution will be used. To demonstrate that this scaling does not cause problems during model estimation we shall discuss the effect of the scale factor on the accuracy of the experimental results. As we shall see, thanks to the model-based processing of statistics a sub-pixel accuracy can be achieved using the

DOI:10.15774/PPKE.ITK.2024.006

automatically extracted correspondences; accuracy which is in fact commeasurable with that of the manual results derived employing the original full resolution.

We proposed a method in [40] which overcomes the limitations of numerical computation of statistics running on the full-sized image in a specific problem. That non-model based method is capable to detect the common field-of-view between cameras.

3.2.3. Modell-based processing of motion statistics

The basic assumption upon which the use of statistics for corresponding point extraction is founded is that the statistics in question has to contain some information about the position of the concurrent point. This information is indeed available, because the series of detected temporal concurrent motions causes a peak in the 2-D distribution defined by (3.8). These distributions are normal distributions because the *central limit theorem* says that the cumulative distribution function of independent random variables (each have an arbitrary probability distribution with mean and finite variance) approaches a normal distribution [39]. In case of a visible reflective surface two peaks are probable in the co-motion statistics, thus the PDF is modeled with a simple Gaussian mixture model (GMM) with two components:

$$\begin{aligned} P_{co}\left(\mathbf{u}\middle|\mathbf{x}\right) &\approx w_{near}^{\mathbf{x}}\mathrm{N}\left(\mathbf{u},\boldsymbol{\mu}_{near}^{\mathbf{x}},\Sigma_{near}^{\mathbf{x}}\right)+w_{coll}^{\mathbf{x}}\mathrm{N}\left(\mathbf{u},\boldsymbol{\mu}_{coll}^{\mathbf{x}},\Sigma_{coll}^{\mathbf{x}}\right) \\ &= P_{near}\left(\mathbf{u}\middle|\mathbf{x}\right)+P_{coll}\left(\mathbf{u}\middle|\mathbf{x}\right), \text{ and } w_{near}^{\mathbf{x}}+w_{coll}^{\mathbf{x}}=1 \end{aligned} \tag{3.9}$$

where the 2-D normal distribution is defined by:

$$\mathrm{N}\left(\mathbf{x},\boldsymbol{\mu},\Sigma\right)=\frac{1}{\sqrt{\left(2\pi\right)^{2}\left|\Sigma\right|}}\exp\left(-\frac{1}{2}\left(\mathbf{x}-\boldsymbol{\mu}\right)\Sigma^{-1}\left(\mathbf{x}-\boldsymbol{\mu}\right)\right) \tag{3.10}$$

The model parameters can be established by using the simple EM algorithm [33] or one of its variants [34]. The first component in formula (3.9), with weight $w_{near}^{\mathbf{x}}$, applies to points in the near vicinity of the investigated point $\mathbf{x}$, termed by $P_{near}\left(.\right)$. The second component (with weight $w_{coll}^{\mathbf{x}}$) describes the far concurrent movements (e.g. in the reflection), this component is denoted by $P_{coll}\left(.\right)$. It can be expressed geometrically with $\left\|\boldsymbol{\mu}_{coll}^{\mathbf{x}}-\mathbf{x}\right\|>\left\|\boldsymbol{\mu}_{near}^{\mathbf{x}}-\mathbf{x}\right\|$. Sample co-motion statistics can be found in Figure 3.19, where the estimated models are also displayed. Note that this model estimation is not

"intelligent"; a GMM is aligned to every PDF where two non-zero peaks are available. This is why the initial set has a large number of correspondences, which include false point pairs.

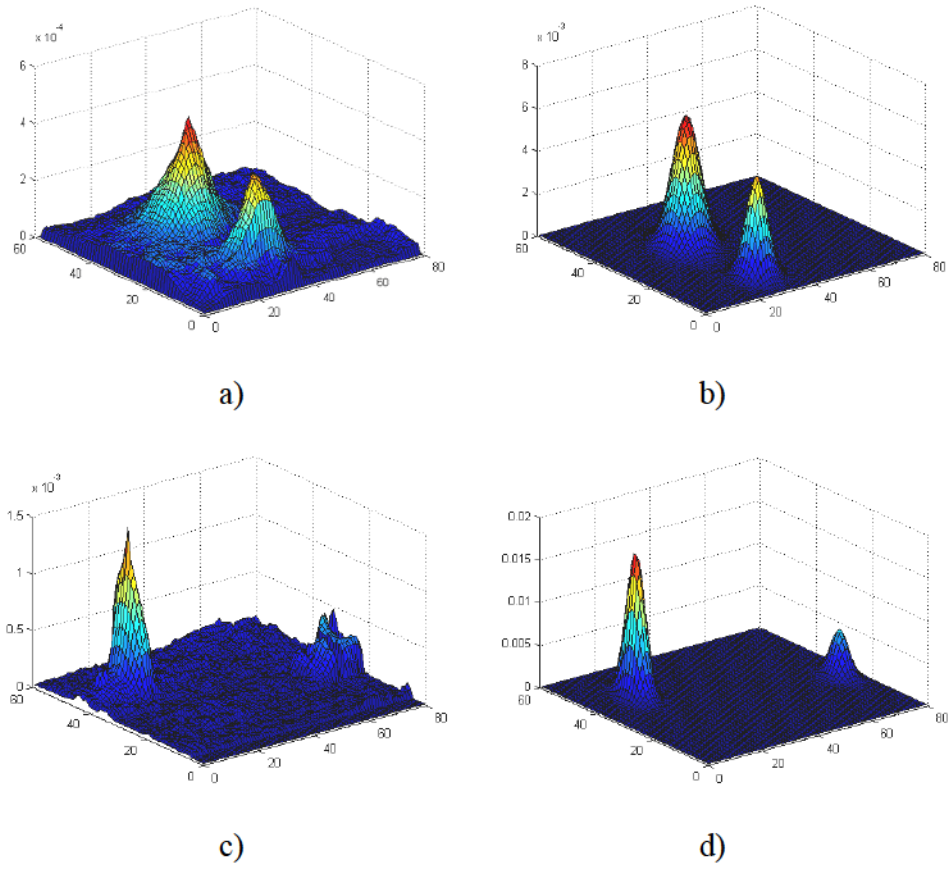


Figure 3.19: A co-motion statistics of the "Mice" and "Shop" videos is displayed as a 3-D surface in a) and c), while their GMM estimations are in b) and d). The higher peak corresponds to $P_{near}(.)$, and the lower one to $P_{coll}(.)$.

3.2.4. Robustness analysis

The main advantage of the above described detection method based on correspondences in co-motion statistics is that there is no need for good image quality, nor for an accurate motion detection method (which is usually a time consuming task). Nonetheless an important question is the method's robustness, i.e. the effect of noise on the results obtained. The robustness can be measured by the difference between the co-motion probabilities of the real concurrent point and of any other arbitrary points. This is an acceptable measure because the GMM can be estimated accurately only when the Gaussians are the most significant peaks on the

DOI:10.15774/PPKE.ITK.2024.006

PDF. In this section after some simplification an expression will be given which is based on elementary probability theories.

Now we investigate a set of different 2-D points $\mathbf{a}, \mathbf{r}$ and $\mathbf{u}$ where $\mathbf{a}$ is a point corresponding to a moving object, $\mathbf{r}$ is its reflection point (we assume that it is visible) and $\mathbf{u}$ is an arbitrary point in the image. We can define the two independent events: $A_{\mathbf{x}}$ (there is real motion at the given point) and $B_{\mathbf{x}}$ (the motion detector fails), and furthermore we introduce the notations of probabilities $f_o^{\mathbf{x}}$ and $f_n^{\mathbf{x}}$, respectively:

$$f_o^{\mathbf{x}} = P_o\left(\hat{m}_t(\mathbf{x}) = 1\right) = P_o\left(A_{\mathbf{x}}\right) = q \text{ and } f_n^{\mathbf{x}} = P_n\left(m_t(\mathbf{x}) \neq \hat{m}_t(\mathbf{x})\right) = P_n\left(B_{\mathbf{x}}\right) = p \quad (3.11)$$

The parameters are the *real* motion intensity q and the error rate p (false positive and false negative detection rates of the algorithm), respectively. The probability of motion detected at pixel $\mathbf{a}$ in time t is:

$$P_m\left(m_t(\mathbf{a}) = 1\right) = P\left(A_{\mathbf{a}}\overline{B}_{\mathbf{a}} + \overline{A}_{\mathbf{a}}B_{\mathbf{a}}\right) = f_o^{\mathbf{a}}\overline{f}_n^{\mathbf{a}} + \overline{f}_o^{\mathbf{a}}f_n^{\mathbf{a}} = (1-p)q + (1-q)p \quad (3.12)$$

Where the overbar indicates the reverse events. This probability reflects the *detected* motion intensity. The effect of the mirror is summarized below. The conditional probability $P_m\left(A_{\mathbf{r}} \middle| A_{\mathbf{a}}\right)$ (probability of the concurrent motion of original point and reflection, see (3.8)) is equivalent to 1, because we assume that the visible point ($\mathbf{a}$) has reflection point ($\mathbf{r}$) which is also visible. Thus,

$$P_m\left(A_{\mathbf{r}}A_{\mathbf{a}}\right) = f_o^{\mathbf{a}} = q \quad (3.13)$$

Similarly, when there is no object in $\vec{a}$ (the object is not visible),

$$P_m\left(\overline{A}_{\mathbf{r}}\overline{A}_{\mathbf{a}}\right) = \overline{f}_o^{\mathbf{a}} = 1-q \quad (3.14)$$

Note that these formulas do not take account of the effect of noise. Since we assumed that the reflection exists in every case where the object is visible, then:

$$P_m\left(\overline{A}_{\mathbf{r}} \middle| A_{\mathbf{a}}\right) = \frac{P_m\left(\overline{A}_{\mathbf{r}}A_{\mathbf{a}}\right)}{P_m\left(A_{\mathbf{a}}\right)} = 0 \Rightarrow P_m\left(\overline{A}_{\mathbf{r}}A_{\mathbf{a}}\right) = 0 \quad (3.15)$$

Thus the joint probability (the detector works correctly, and there is motion at both points) may be written in the form:

$$P\left(A_{\mathbf{a}}\overline{B}_{\mathbf{a}}A_{\mathbf{r}}\overline{B}_{\mathbf{r}}\right) = P\left(A_{\mathbf{a}}A_{\mathbf{r}}\right)P\left(\overline{B}_{\mathbf{a}}\right)P\left(\overline{B}_{\mathbf{r}}\right) = f_o^{\mathbf{a}}\overline{f}_n^{\mathbf{a}}\overline{f}_n^{\mathbf{r}} = q(1-p)^2 \quad (3.16)$$

DOI:10.15774/PPKE.ITK.2024.006

Furthermore, for the case where there is motion in neither of the two points, and the detector fails at both points, we have:

$$P\left(\overline{A}_{\mathbf{a}}B_{\mathbf{a}}\overline{A}_{\mathbf{r}}B_{\mathbf{r}}\right)=P\left(\overline{A}_{\mathbf{a}}\overline{A}_{\mathbf{r}}\right)P\left(B_{\mathbf{a}}\right)P\left(B_{\mathbf{r}}\right)=\overline{f}_{o}^{\mathbf{a}}f_{n}^{\mathbf{a}}f_{n}^{\mathbf{r}}=\left(1-q\right)p^{2} \quad (3.17)$$

Because of (3.15), it was assumed that the reflection of an original point could not be hidden

$$P\left(A_{\mathbf{a}}\overline{B}_{\mathbf{a}}\overline{A}_{\mathbf{r}}B_{\mathbf{r}}\right)=0 \quad (3.18)$$

Finally, we have the case where there is no motion at the original point but there is some other real moving object in the plane of the mirror:

$$P\left(\overline{A}_{\mathbf{a}}B_{\mathbf{a}}A_{\mathbf{r}}\overline{B}_{\mathbf{r}}\right)=\overline{f}_{o}^{\mathbf{a}}f_{n}^{\mathbf{a}}f_{o}^{\mathbf{r}}\overline{f}_{n}^{\mathbf{r}}=\left(1-q\right)pq\left(1-p\right) \quad (3.19)$$

The conditional probability of the true co-motion probability is expressed by using the above defined expressions, where the detected motion is present at point **a** (because the algorithm collects motion information only when there is detected motion):

$$P_{m}\left(m_{t}\left(\mathbf{r}\right)=1\middle|m_{t}\left(\mathbf{a}\right)=1\right)=\frac{f_{o}^{\mathbf{a}}\overline{f}_{n}^{\mathbf{a}}\overline{f}_{n}^{\mathbf{r}}+\overline{f}_{o}^{\mathbf{a}}f_{n}^{\mathbf{a}}f_{o}^{\mathbf{r}}\overline{f}_{n}^{\mathbf{r}}+\overline{f}_{o}^{\mathbf{a}}f_{n}^{\mathbf{a}}f_{n}^{\mathbf{r}}}{P_{m}\left(m_{t}\left(\mathbf{a}\right)=1\right)}=\frac{q\left(1-p\right)^{2}+\left(1-q\right)pq\left(1-p\right)+\left(1-q\right)p^{2}}{\left(1-p\right)q+\left(1-q\right)p} \quad (3.20)$$

Because of the motion at points **u** and **a** are uncorrelated (and the events are independent of each other), the following probability expression describes the false co-motion probability between the two points:

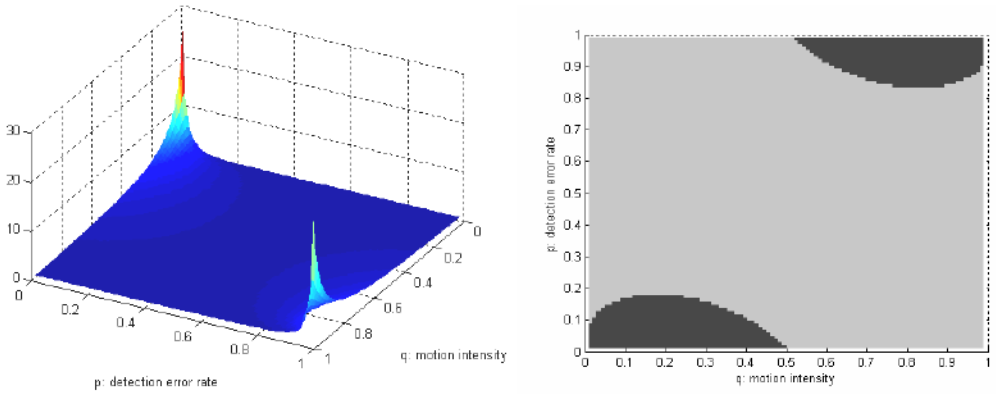
$$P_{m}\left(m_{t}\left(\mathbf{u}\right)=1\middle|m_{t}\left(\mathbf{a}\right)=1\right)=\frac{P_{m}\left(m_{t}\left(\mathbf{u}\right)=1\right)P_{m}\left(m_{t}\left(\mathbf{a}\right)=1\right)}{P_{m}\left(m_{t}\left(\mathbf{a}\right)=1\right)}=p+q-2pq \quad (3.21)$$

The proposed corresponding point extraction method works dependably when the probability of true co-motion (3.20) is significantly larger than the probability of false co-motion (3.21). This difference assists us to determine the model parameters (see (3.9)) accurately; thus the ratio of these probabilities may be regarded as an indirect index to measure the robustness of the correspondences extraction:

$$\eta=\frac{P_{m}\left(m_{t}\left(\mathbf{r}\right)=1\middle|m_{t}\left(\mathbf{a}\right)=1\right)}{P_{m}\left(m_{t}\left(\mathbf{u}\right)=1\middle|m_{t}\left(\mathbf{a}\right)=1\right)}=\frac{q-pq\left(1+q\right)+p^{2}\left(1-q+q^{2}\right)}{\left(p+q-2pq\right)^{2}}>1 \quad (3.22)$$

DOI:10.15774/PPKE.ITK.2024.006

We may mention that this inequality is always satisfied; this is provable by investigating the difference sum between the numerator and the denominator. Taking the case with the larger ratio is an easy and accurate way to extract the GMM parameters. The next figure demonstrates the dependencies of the ratio η to the detection error rate and the motion intensity.



a) b)

Figure 3.20: The value of η with varying motion intensity and detection error rate. We experimentally define the reasonable cases with $\eta > 2$, see b), which shows the "good" regions.

The results point out an interesting property, namely the existence of a second area where q>0.5 and p>0.5. In this case the motion intensity and the detection error rate are both very high. It is equivalent to the extraction of correspondences by using the non-moving (static) points. We can call this case "concurrent non-moving points". Nonetheless, the utilization of this property in real scenes is difficult because there are regions (such as the sky) where any motion rarely occurs, and consequently these regions could be concurrent for any point. In our experience the usual condition $\eta > 2$ is sufficient for the extraction of correspondences. The acceptable configurations of parameter-space can be seen in Figure 3.20(b), where the wide range of the acceptable p and q confirm applicability of our motion statistics based method in various environment.

Another important and practical question is the length of video sequence necessary to produce valid results. The knowledge of the minimal length assists us to start the optimization process at a moment when good accuracy may be achieved. An estimation of the required length (frame count) is possible by using the Bernoulli's theorem or *weak law of large numbers* [35]:

DOI:10.15774/PPKE.ITK.2024.006

$$P\left(\left|\frac{k}{n}-q\right|\geq\varepsilon\right)\leq\frac{q(1-q)}{\varepsilon^2 n}\leq\chi \tag{3.23}$$

The above formula estimates the probability of the deviation of relative frequency (with k/n, where k is the number of positive cases n is the total number of cases) from a probable value q with a predefined allowable error value ε. It gives an estimated value of n which is the total sample count. In our case this formula represents a way to determine the necessary frame count to achieve the desirable co-motion statistics when the motion probability (q) can be estimated for the sequence. For this purpose we have to define an appropriate error value (ε) which may be computed for a given q by using (3.22) where the condition $\eta>2$ is satisfied. In our context this condition means that the extraction of co-motion statistics is possible. The last parameter to be determined is χ, defined by (3.23). This parameter defines the probability that the stipulated error rate will be achieved. An acceptable value is $\chi=0.05$, which means that the allowable error rate will be achieved in 95% of cases, and the probability that the deviation is larger than the given value is smaller than 0.05. In this case the following inequality for n is implied:

$$\frac{q(1-q)}{\varepsilon^2\chi}\leq n \tag{3.24}$$

Our experiments relating to this theoretical investigation are presented in a later section.

DOI:10.15774/PPKE.ITK.2024.006

3.3. Conclusions

In this chapter, we have introduced a robust pedestrian detection and gait feature extraction method. It is able to achieve a reliable detection rate in indoor or outdoor camera configuration and environmental conditions using an invariant and effective data representation in the Eigenwalk space, based on spline interpolation and a dimension-reduction technique. A novel method for leading leg identification has been presented; this is a possible gait characteristic for walker registration between multiple cameras capturing different views of the same target. An important goal was to use this feature for the purpose of multiple-camera registration.

We have introduced a novel application of co-motion statistics applicable to video sequences, which allows estimation of the vanishing-point in a geometrical model of a view containing a plane mirror. The mirror-reflection may be replaced with the image from a virtual projective camera. To determine the geometrical connection between the real and the virtual views (even in a wide-baseline configuration) we made use of the extra information that is provided by the video sequence; the necessary corresponding point-pairs are retrieved from the concurrent-motion statistics.

In conclusion, the main advantage of our method is that it is not dependent on scene content and it is robust in situations where manual configuration is difficult. However, its prerequisite is that the input is available as a video sequence; and to achieve good results a sufficiently large frame count and the presence of sufficient target motion are necessary.

DOI:10.15774/PPKE.ITK.2024.006

4.Estimation of model parameters

Key steps for parameter estimation problems are the data preprocessing and the final parameter optimization. Both steps are crucial in point of accuracy and robustness. Most approaches use non-linear optimization after the initial guess about the parameter vector. This is because there is no analytical form of solution, thus the use of some searching method is necessary in the parameter space.

In this chapter we describe methods for outlier rejection which nature depends on the properties of the model and the data. We will show that the model parameters can be determined from the extracted features with notable accuracy.

DOI:10.15774/PPKE.ITK.2024.006

4.1. Homography computation using DLT and RANSAC

An automatic registration method needs a feature selection and matching algorithm to select corresponding points between the views obtained from two cameras. In 3D motion-based camera calibration, a major problem is to estimate the height of the motion above the defined ground-plane. In our case, however, since we can detect the legs in motion, their lower point can conveniently be used for registering common points on the ground-plane.

To detect such corresponding points, we use our walk-detection and leading-leg identification methods. Both methods provide information, which is useful in matching points between the two views: detected walk patterns must be concurrent in both views; and, likewise, the leading leg must be the same. In both views, the central lower points of the detected walk-patterns are the feature points (e.g. the one marked with a black “x” in Figure 3.11(a)). Searching for its counterpart in the other view follows the extraction of a feature point from one of the views. The algorithm searches for con-current points by examining the timestamps of points, and for points, which were detected during walk cycles with the same leading leg. Nevertheless, none of these features is unique for person identification (in cases where more than one person is visible). This fact results in some outliers in the detected points. However, because the leading leg is a stronger feature for identification, there are fewer outliers than if we were to use only the concurrent condition. For the estimation of the transformation H that maps points of one camera scene onto the other, and for rejection of outliers from the set of candidate point-pairs, we have implemented both the simple DLT (Direct Linear Transformation) method, and its extension using the RANSAC (RANdom SAmple Consensus) algorithm [24]. In DLT a simple linear solution for H may be derived from the expression of (2.1). The RANSAC algorithm partitions the data set into inliers and outliers, and also delivers an estimate of the model H (the homography in our case).

DOI:10.15774/PPKE.ITK.2024.006

4.1.1. Experimental results

We evaluated the registration algorithm by using surveillance cameras placed in a public area located in the university building. The angle between the view-axes of the two overlapping cameras employed was nearly 90° (hence, to detect corresponding points using standard optical methods would be difficult). In the non-overlapping layout the two cameras are placed oppositely to each other.

Overlapping views

In our series of tests, the successfully detected and classified walk-patterns were 241 for the first camera, and 220 for the second camera (see Figure 4.1). In our system the cameras are approximately synchronized, but there is a small temporal drift between the walk patterns generated by each camera; hence we define a permitted time-window for events, which are classed as "concurrent". This time-window for concurrent checking was 5 frames. After such checking, there remained 46 concurrent corresponding points (S1 dataset) and 8 with the leading leg verified (S2 dataset). We found 15 invalid points in the S1 dataset. Table 4-I summarizes the results of the simple DLT and the RANSAC+DLT methods applied to several combinations of the S1 and S2 datasets (cases 1 to 5). We assessed the accuracy of the computed transformations (rightmost column) using manually-selected control points.

Because of the near-orthogonal orientation of the two cameras used for the tests, the algorithm can rarely detect two successive walks for leading-leg identification, and therefore there are only a few points in the S2 dataset. Nevertheless, in case 1, all the points in S2 are correct points; and the simple DLT method can compute a good transformation. The DLT method fails when there are outliers (as for the S1+S2 dataset), and in this case (case 2) the position error is extremely high. In cases 4 and 5, the RANSAC algorithm has managed to reject the outliers from S1, and the DLT method then computes an appropriate transformation. In case 3, RANSAC+DLT fails to give good accuracy because there are only a few points in the S2 dataset.

DOI:10.15774/PPKE.ITK.2024.006

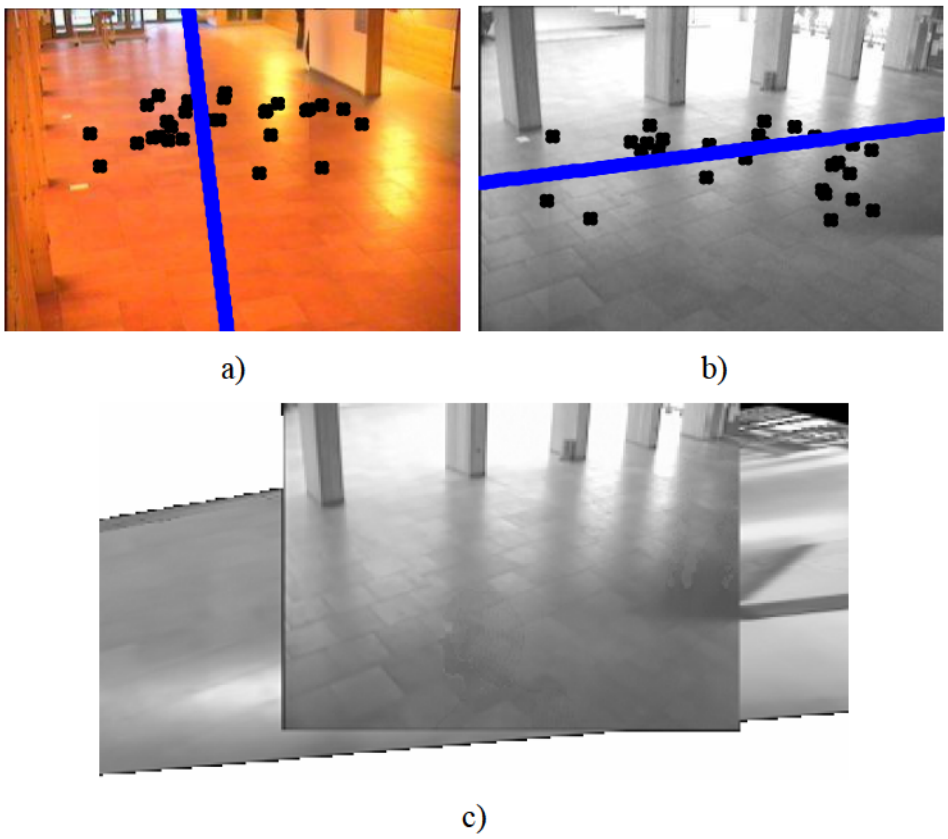


Figure 4.1: Transformation from the first-camera view (left) to the second (right): Detected corresponding points, and a synthetic line-trajectory in a) and b) and alignment of views in c).

TABLE 4-I: EXPERIMENTAL RESULTS ON DATA FROM "ENTRANCE" CAMERAS (RANSAC DISTANCE THRESHOLD IS T=0.01)

Case	Input	Points	Correct points	Detected outliers	Avg. error in pixel
Method: DLT					
1	S2	8	8	-	6.4
2	S1+S2	54	39	-	250.2
Method: RANSAC+DLT					
3	S2	8	8	4	12.5
4	S1+S2	54	39	25	7.8
5	S1	46	31	28	6.2

In the indoor test sequences the height of people was 115 pixels and their width was 40 pixels in average. The camera view registration is a good test bed for the evaluation of the proposed feature extraction method. The not so high average error of alignment (cc. 5% relative error to the height of object) with respect to the object size proves the usability of the localized features.

DOI:10.15774/PPKE.ITK.2024.006

To summarize the test-results: the DLT method is fast enough to run in real-time, but it needs an input containing only "good" points (like our S2 dataset). On the other hand the RANSAC algorithm can successfully reject outlier points (such as contained in our S1 dataset) but it does require much more computing time (5-20 seconds).

Non-overlapping views

In the last experiment, we aligned images of cameras with non-overlapping field of view. The schematic map of experiment and the images of the "Main hall" and the "Entrance" cameras are shown in Figure 4.2.

It can be seen that the field of views of the cameras are not overlapped because of the wall between "Main hall" and "Entrance" areas but virtually they do. The estimation of the homography is based on line correspondences and not on point correspondences as in previous experiments. Two successive walk steps were detected and a line was calculated through them.

The major assumption in this experiment is that people are moving along straight lines from "Main hall" to "Entrance" and vice versa. Every line from one view was paired with every line in the other view and the RANSAC algorithm was used for the estimation of the model and rejection of outliers.

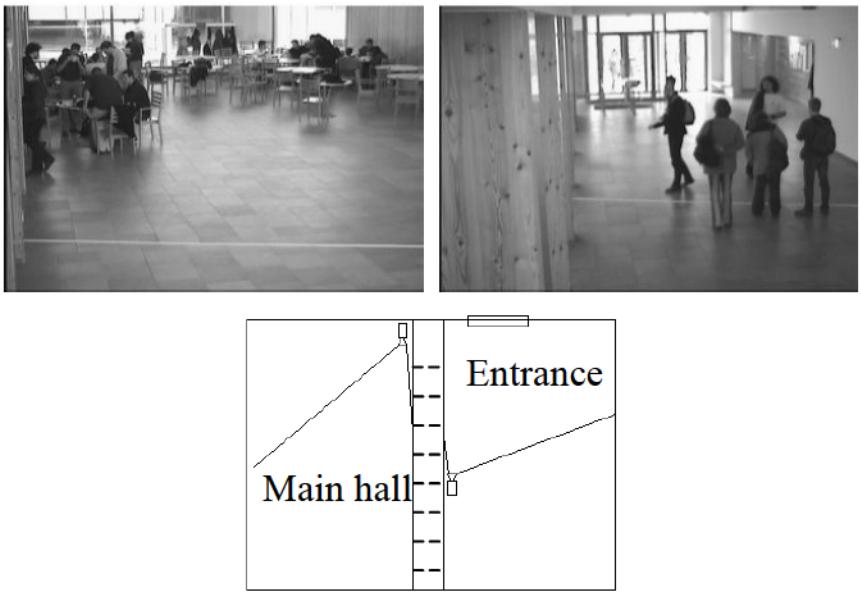


Figure 4.2: Images of "Main hall" and "Entrance" cameras with control lines on the ground (marked with two long paper tapes) for verification. Schematic map of the experiment: placement of cameras and their field of views.

DOI:10.15774/PPKE.ITK.2024.006

The results of aligned images are shown in Figure 4.3. The results are based on 235 detected walks; 42 walk cycle (two walks form a line fragment as mentioned above) and 9 inliers left after the RANSAC has been performed. The average deviation of the gradient of the real “paper tape” lines was 12.5°.

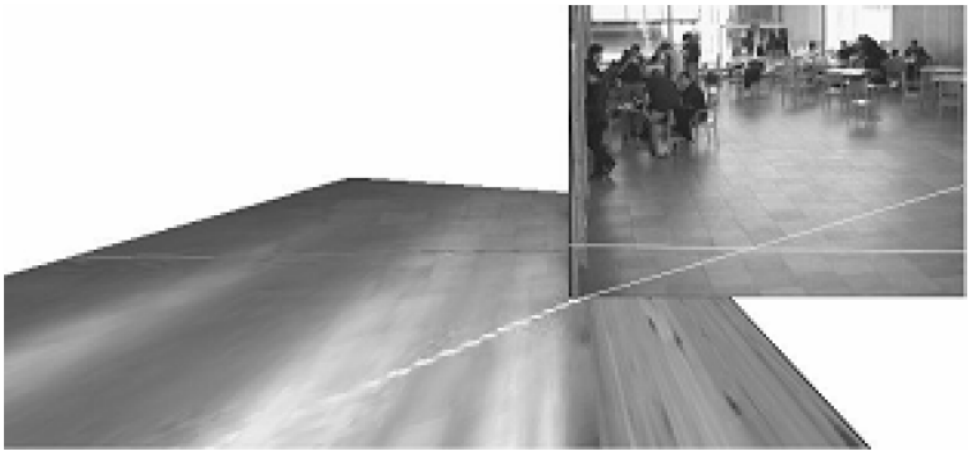


Figure 4.3: Result of alignment of non-overlapping views with the highlighted control lines.

DOI:10.15774/PPKE.ITK.2024.006

4.2. Vanishing point determination

The determination of VP coordinates (the model parameter) is performed as the computation of the intersection of lines defined by corresponding point-pairs. There are several approaches for model estimation in camera systems, good surveys can be found in [24] and [36]. The methods commonly used in 3-D reconstruction in camera-mirror case are e.g. the *data symmetrization* [37][25] and searching window technique [23]. These methods can tolerate only a small amount of noise in the point positions (and, likewise, it cannot handle outliers probably). Generally, the model determination problems need some robust estimator [24] that handles the function as an (global) optimization task, thus there is no analytic solution [49] for such robust parameter estimation procedures, for survey see [36].

In our implementation the parameter estimation comprises two steps. Firstly we reduce the number of outliers, and secondly a non-linear optimization is performed for final VP estimation. The inclusion of the information extracted from global and co-motion statistics into the objective function conveniently provides an approach which is slightly differs from previous methods.

4.2.1. Corresponding points in single view

The extraction of corresponding points in related image-pairs is an extensively-studied research area in computer vision [59][24]. But despite this, the number of specialized methods which are applicable to camera-mirror scenes is rather limited. In [28] a method was introduced which uses the so-called support lines of the silhouettes of the image-components in order to find corresponding point-pairs. This method assumes that the whole 3-D mirror hull (the convex hull of the object and its mirror image) is always visible by the camera, and that there is only one object in the scene. In an outdoor environment, however, these assumptions are usually not satisfied. In our approach, we do not require any sophisticated feature-detection in achieving matching of the views.

4.2.2. Outlier rejection

Depending on the configuration of the observed scene, not every moving point will necessarily have a visible reflection, thus there is a need for a filtering before the

DOI:10.15774/PPKE.ITK.2024.006

optimization step is performed. The following method enables us to exclude a considerable proportion of these outliers (points which have no reflection).

$$\varepsilon_1 < \frac{w^{\mathbf{x}}_{near}}{w^{\mathbf{x}}_{coll}} < \varepsilon_2 \tag{4.1}$$

The required subset of S can be defined with points that satisfy the above condition:

$$S_2 = \left\{ \mathbf{x} \middle| \mathbf{x} \in S \text{ and } P_{co}\left(. \middle| \mathbf{x}\right) \text{ exists} \right\} \tag{4.2}$$

For the purpose of this discrimination we make use of the orientation of the line determined by the two points of a corresponding point pair. In case of inliers these lines point to (or near to) the VP, thus the directions are clustered around a characteristic value, depending on the scene setting. This is a well-known technique for processing motion vectors in navigation tasks. For detailed description of this technique and implementation issues we refer to [41].

For the purpose of this discrimination we make use of the included angle between the horizontal line and the line determined by the two points of a corresponding point pair, see figure.

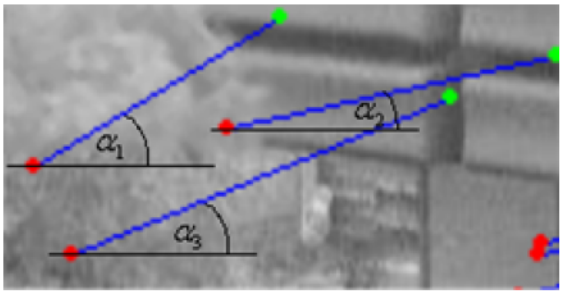


Figure 4.4: Interpretation of included angle for the computation of orientation histogram of corresponding point.

In case of inliers these lines point to (or near to) the VP, thus the directions are clustered around a characteristic value, depending on the scene setting. To filter out the lines that do not point in the most probable direction is straightforward, by using a weighted histogram of angles computed by using point pairs from S_2. The weighting factor comes from the global motion statistics which assists us to prove the dominance of the relevant point pairs:

$$h(k)_{k=1..N} = \sum_{\mathbf{x} \in S_2,\, \alpha_k < \left(\boldsymbol{\mu}^{\mathbf{x}}_{coll}, \boldsymbol{\mu}^{\mathbf{x}}_{near}\right)_{\alpha} \le \alpha_{k+1}} P_g(\mathbf{x}) \tag{4.3}$$

DOI:10.15774/PPKE.ITK.2024.006

where we define N bins on the (0..π) interval, and the interval limits are denoted by α_k. The expression $(\mathbf{a},\mathbf{b})_{\measuredangle}$ symbolizes the angle between the horizontal and the line through points $\mathbf{a}$ and $\mathbf{b}$. It is obvious that there will be one or more peaks in this histogram. Because the pre-filtering defined by (4.1) has removed a large proportion of outliers, the main peak in the histogram can be associated with the true inliers.

The histograms (before and after outlier rejection) and the filtered correspondences can be seen in Figure 4.5. This simple method also works well in indoor images (e.g. our "Ants" and "Mice" sequences) because not only the points actually at the main peak are retained in the final data set, but also the points around this peak.

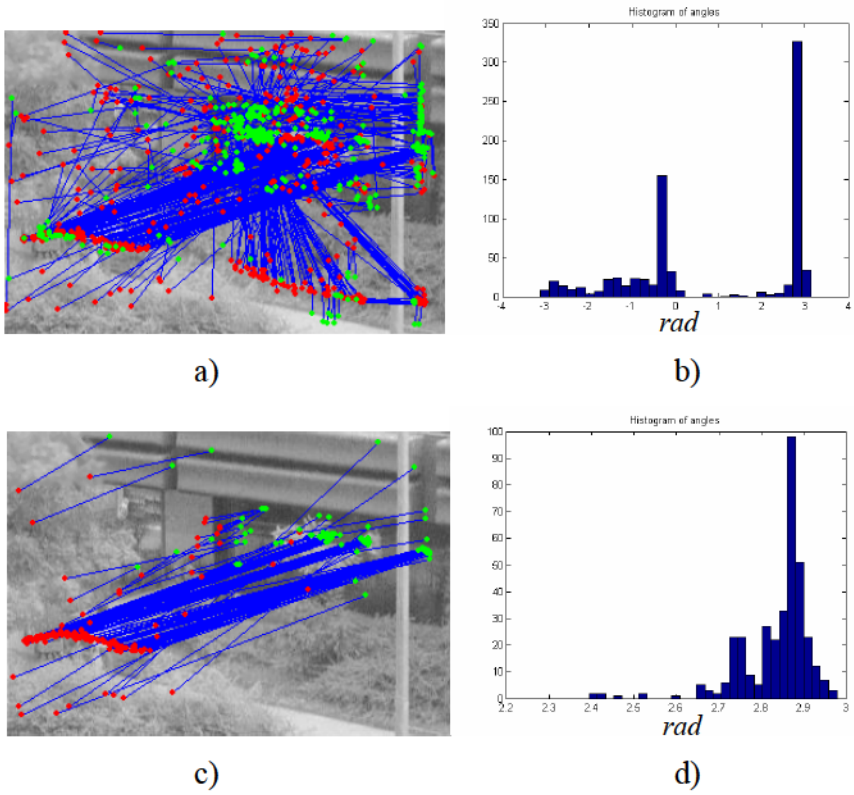


Figure 4.5: Rejection of outliers for the "Shop" sequence. Only the directions corresponding to the main peak (mode) of the histogram (determined from the line directions) will be used for later computations. a) before rejection (only 320 of the total 3566 point pairs are displayed), c) after rejection (382 point pairs); b) and d) show the corresponding histograms of angles.

4.2.3. Optimization procedure

Because the VP is computed from line intersections, the remaining outliers and the measurement errors in point coordinates cause considerable problems during parameter estimation. The most notable problem with outdoor scenes (e.g. the "Shop" video, see Figure 4.5(c)) is that the near parallel lines through corresponding points

DOI:10.15774/PPKE.ITK.2024.006

lead to large deviations in the position of line intersections. Furthermore, the near parallel lines intersect each other only toward infinity. The method that has been introduced utilizes the extra information extracted from videos: the motion statistics and the large number of correspondences for the computation of geometric model fitting.

In the following section we introduce the goodness-of-fit function to assess the fitting of a possible VP position to the extracted corresponding points; the "best" VP is the argument of the goodness-of-fit function at its global maximum:

$$VP = \mathbf{c}' = \arg\max_{\mathbf{u}} \sum_{\mathbf{x}\in S_2} P_g(\mathbf{x}) P_{coll}\left(\vec{\delta}(\mathbf{x},\mathbf{u})\middle|\mathbf{x}\right) \tag{4.4}$$

The function $\vec{\delta}(\mathbf{x},\mathbf{u})$ returns the 2-D position related to the largest value of the Gaussian function corresponding to $P_{coll}(\mathbf{v}|\mathbf{x})$ where the points $\boldsymbol{\mu}^{\mathbf{x}}_{near}$, $\mathbf{u}$ and $\mathbf{v}$ are collinear:

$$\vec{\delta}(\mathbf{x},\mathbf{u}) = \arg\max_{\mathbf{v}\in S_2} P_{coll}(\mathbf{v}|\mathbf{x}) \text{ and } \left\langle \tilde{\boldsymbol{\mu}}^{\mathbf{x}}_{near} \times \tilde{\mathbf{u}}, \tilde{\mathbf{v}} \right\rangle = 0 \tag{4.5}$$

Note that this expression has a closed form solution [49].

In summary, this approach enables a small shift around point $\boldsymbol{\mu}^{\mathbf{x}}_{coll}$ (center of concurrent area, see (3.9)) while point $\boldsymbol{\mu}^{\mathbf{x}}_{near}$ is fixed. The allowable shift is defined with the Gaussian distribution ($P_{coll}(.)$), which is an approximation of the size of occurring objects in point $\boldsymbol{\mu}^{\mathbf{x}}_{coll}$. This is another advantage of co-motion statistics; the admissible error (in position) at every point may be estimated from the statistics, and the above-described optimization method uses this information too. Figure 4.6, below, shows the process of optimization.

DOI:10.15774/PPKE.ITK.2024.006

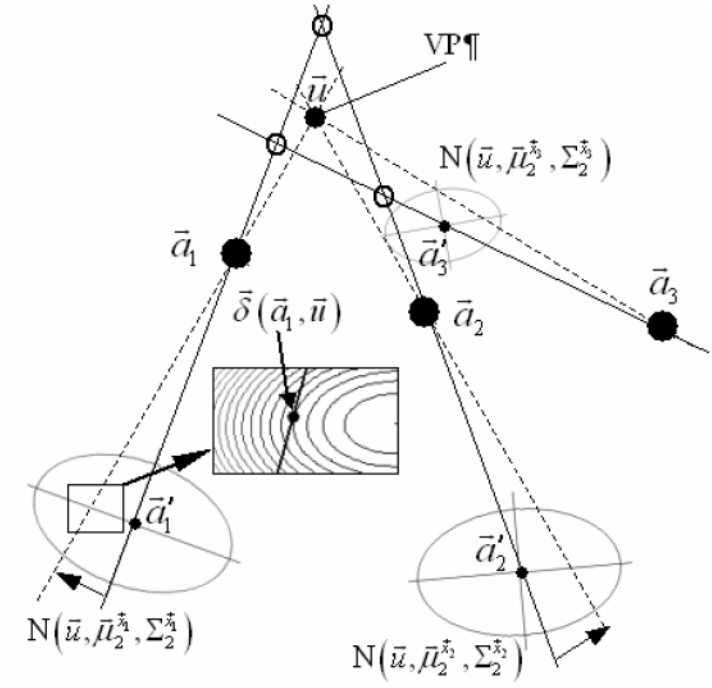


Figure 4.6: The three correspondences illustrate the optimization process. Open circles show the initial intersections (these differ from one another). Dashed lines are those drawn to the modified points after optimization is completed. The meaning of vector-function $\vec{\delta}(.)$ is demonstrated: it returns the position on the line where the Gaussian is maximal.

Goodness-of-fit values are illustrated in Figure 4.7 with a contour graph.

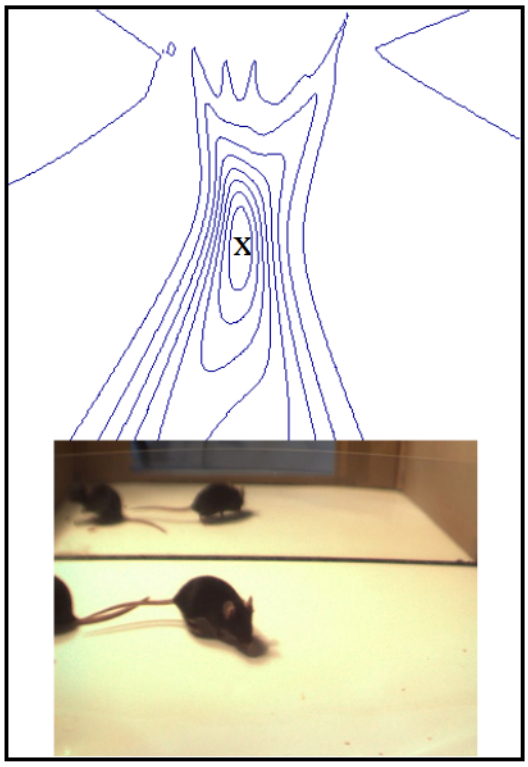


Figure 4.7: Goodness-of-fit function represented using a contour graph; the VP is marked with "x".

DOI:10.15774/PPKE.ITK.2024.006

More general criteria can be derived which enable a correction not only in $\boldsymbol{\mu}^{\mathbf{x}}_{coll}$ but in point $\boldsymbol{\mu}^{\mathbf{x}}_{near}$ as well (this modification affects (4.4)and (4.5)). Unfortunately, after this modification (4.5) will not have an analytic solution; thus during the optimization an algorithmic scan would have to be run for every corresponding point pair, instead of a simple equation substitution.

The knowledge of the VP position assists in the partitioning of the GMM components - corresponding to an arbitrary co-motion statistics - into two parts: one for the original point, and the other for its reflection. This is simple because using property 2 the nearest point to the VP is the reflection. A new notation is introduced in the following conversion of (3.9):

$$P_{co}\left(\mathbf{u}\middle|\mathbf{x}\right) = P_{orig}\left(\mathbf{u}\middle|\mathbf{x}\right) + P_{refl}\left(\mathbf{u}\middle|\mathbf{x}\right) \tag{4.6}$$

where $\left\|\boldsymbol{\mu}^{\mathbf{x}}_{orig} - \mathbf{c}'\right\| > \left\|\boldsymbol{\mu}^{\mathbf{x}}_{refl} - \mathbf{c}'\right\|$. Based on this repartitioning we can define the following pair of 2-D PDFs for motion probability; one for motion probability in the foreground region:

$$P_{fg}\left(\mathbf{x}\middle|F\right) = \sum_{\mathbf{u}\in S_2} P_g\left(\mathbf{u}\right) P_{orig}\left(\mathbf{x}\middle|\mathbf{u}\right) \tag{4.7}$$

and the other for motion probability in the reflection region:

$$P_{rf}\left(\mathbf{x}\middle|F\right) = \sum_{\mathbf{u}\in S_2} P_g\left(\mathbf{u}\right) P_{refl}\left(\mathbf{x}\middle|\mathbf{u}\right) \tag{4.8}$$

4.2.4. Experimental results

Our basic motion detection method [12] is founded on the background model introduced by Stauffer; his accurate but rather time-demanding foreground detector is a good basis for further classification of the foreground mask. Its output has already been mentioned in formula (3.5). The initial color-based shadow detection method is a modification of the corresponding part of SAKBOT. The use of both the shadow and motion mask together is possible after the following modification of (3.8):

$$f_{sh}\left(\mathbf{u},\mathbf{x}\right) = P_{co}\left(\mathbf{u}\middle|\mathbf{x}\right) = \frac{\sum_t m_t\left(\mathbf{x}\right) s_t\left(\mathbf{u}\right)}{\sum_t \left(m_t\left(\mathbf{x}\right) + s_t\left(\mathbf{u}\right)\right)} \tag{4.9}$$

DOI:10.15774/PPKE.ITK.2024.006

In the formula $s_t(\mathbf{u})$ denotes the binarized shadow mask. There are three disjunctive classes in this case (foreground, shadow, and background): a given point may not be in both the foreground mask and the shadow mask.

In our experiment the error rate of the foreground detector was 2-5%; while the simple color based shadow detector achieved nearly a 75% success rate (we deliberately did not tune the color model parameters, in order to test the method's robustness). The outputs of the foreground and shadow detection algorithms are illustrated in Figure 4.8.

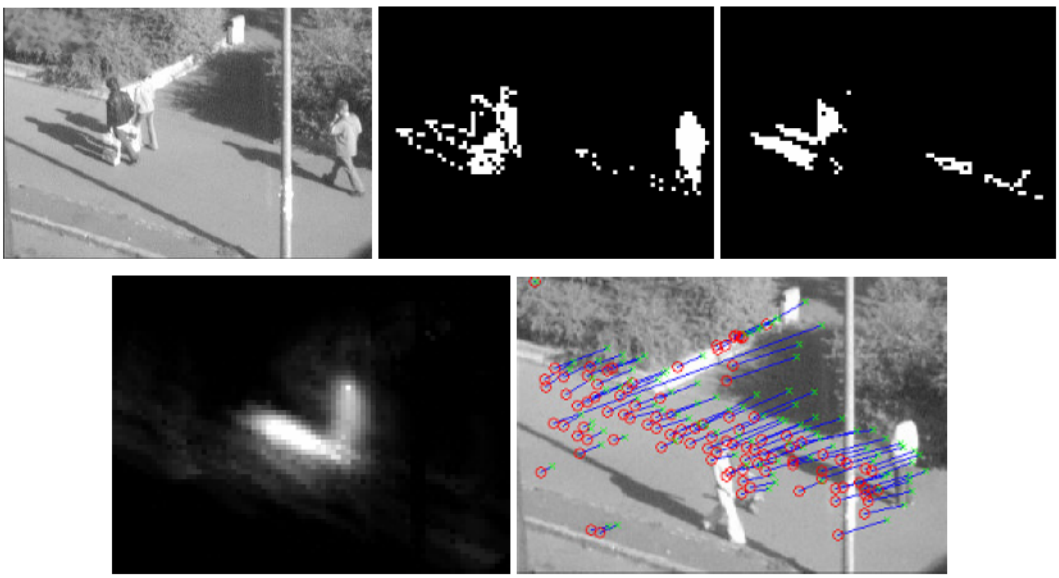


Figure 4.8: Computation steps: input image, foreground and shadow masks, a co-motion statistic and extracted correspondences in "Shadow" sequence. The corresponding points are the extracted object-shadow point pairs after outlier rejection.

The numerical results of VP estimation are summarized in Table 4-II. The numbers of point pairs in the processing steps and the *goodness-of-fit* value corresponding to the optimized VP are displayed for 4 different test videos. The results are illustrated in Figure 4.9, where the collinearities are observable, and which demonstrate the accuracy achieved. (Because of the large coordinate values, the VPs themselves are not shown in the images.)

DOI:10.15774/PPKE.ITK.2024.006

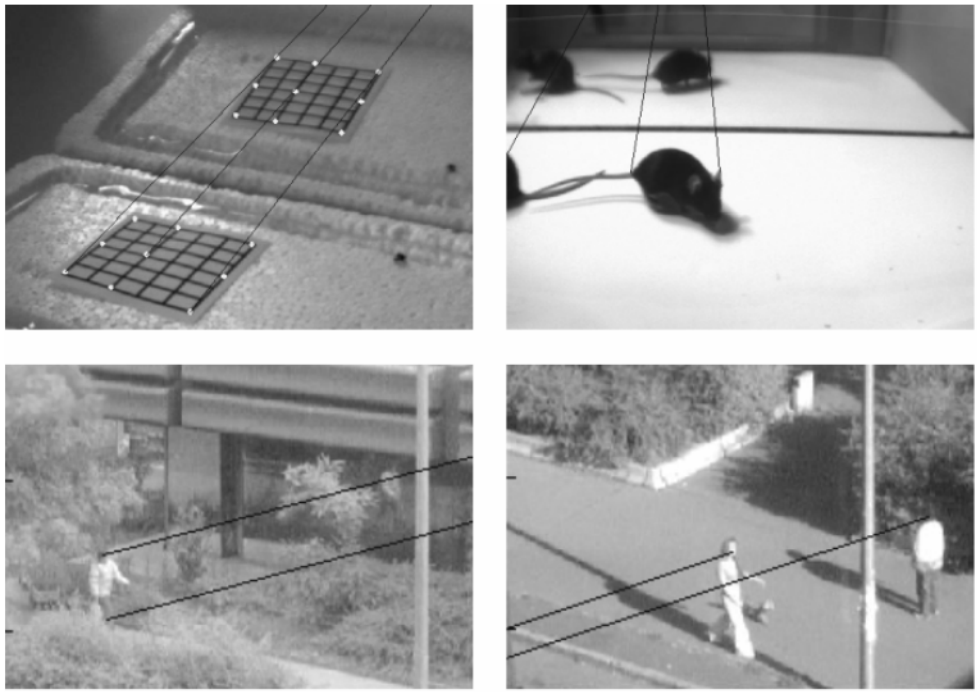


Figure 4.9: Results are demonstrated with the collinearities of VP, original point and reflected point.

In the Table 4-II, the *true* VP values are based on manual extrapolation. It is important to note that – as is also demonstrated in the sample images – the apparent inaccuracy in the VP's position in case of the "Shop" video has not caused a perceptible error in the collinearity. This is because these VPs are nearly at infinity.

TABLE 4-II: RESULTS ON MODEL OPTIMIZATION

Sequence Name	Point# (S_2)	Vanishing point			Fit	Model error
		Initial*	Optimized	Ground truth		
Ants	1218	(1081,-629)	(1103,-674)	(1128,-676)	0.9451	0.38°
Mice	2587	(253,-237)	(256,-260)	(260,-258)	0.9614	2.06°
Shop	382	(300,106)	(675,-163)	(1200,-200)	0.8623	1.06°
Shadow	3509	(-13,53)	(-1918,680)	(2110,850)	0.9847	4.58°

*: The initial estimate is given by LMS method.

Probably the larger objects (as in the "Mice" video) resulted in better goodness-of-fit values because the allowable margin is larger than is the case for smaller objects such as occur in the "Ants" and "Shop" sequences. In evaluation, the accuracy of the computed VP is conspicuous for the "Ants" video; the reason is that the small and rarely-moving objects generated accurate corresponding point pairs. The error in the "Mice" video arises from the large objects and the "strong" projection. The relatively large error in the "Shadow" video is because of the rather small track in the image

DOI:10.15774/PPKE.ITK.2024.006

where motion is apparent. The inlier points cluster around this track and thus did not provide the uniform distribution in the whole image which would allow a better result.

Limits on scale factor

In our tests, the image resolution was reduced to only 80x60 pixels. Because this step may seem rather drastic, we also derived the model at different scales in order to demonstrate that the scaling has no adverse effect. The results and computational details (memory usage and computation time) are summarized in Table 4-III. As can be seen, even a much more drastically reduced resolution does not cause inaccuracy; the results are still near to the "true" results. This is a promising property of our statistical method; it allows sub-pixel precision. These results were generated by using a simple running-average process [44] for change detection in order to generate the motion mask, instead of the model-based background subtraction method [12]. This change detector is less precise than the latter method but it is widely used in common applications, and we used it exclusively for the generation of the results in Figure 4.10. The error rate of this detector is significantly larger than that of the more sophisticated model-based method. Thus, the results support both the robustness and the applicability in low-resolution video streams where a simple change-detector is used. A sample motion mask can also be seen in Figure 4.10, where the detection errors are visible at every scale level.

TABLE 4-III: RESULTS AT VARYING SCALE FACTORS

Resolution (pixels)		80x60	64x48	40x30	32x24	20x15
Max. memory usage (MByte)		88	36	6	2.5	0.5
Computation time (milliseconds/frame)		25	20	18	15	12
VP of Sequence	Mice	(246,-307)	(244,-311)	(240,-337)	(238,-358)	(227,-365)
	Shop*	(670,-89)	(630,-60)	(735,-165)	(942,-107)	(474,31)

*: in the images below the effect of a large Euclidean error on the VP position can be seen; usually, it is negligible compared to the collinearities

DOI:10.15774/PPKE.ITK.2024.006

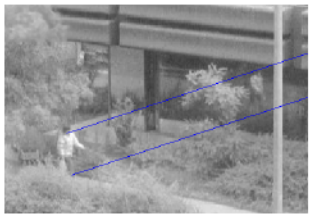
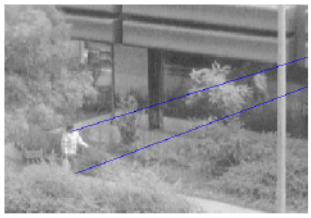
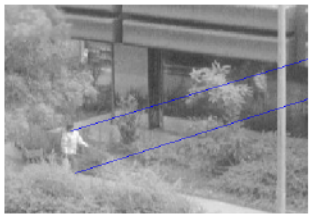
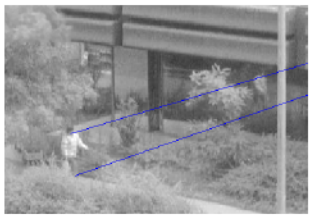
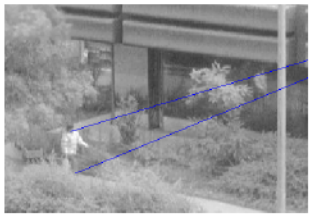
Resolution	Change detection mask	Result
80x60		
64x48		
40x30		
32x24		
20x15		

Figure 4.10: Results demonstrated for different scales. In the left column sample motion masks of "Shop" video are extracted using a simple running-average change detector; the right column demonstrates the results of VP estimation based on such ambiguous motion masks.

Effect of processed video length

We recall that the processed frame count is important, since there is a minimal information content which is necessary for the extraction of corresponding point pairs and for the determination of the VP position with acceptable accuracy. The formula (3.24) gives an estimation of the lowest necessary frame count, which in fact depends on the motion intensity and the detection error rate. Temporally, we found that the

DOI:10.15774/PPKE.ITK.2024.006

motion activity was smooth in indoor videos, but rather unbalanced in outdoor videos. The following table summarizes the numerical estimation of parameters.

TABLE 4-IV: ROBUSTNESS ANALYSIS RESULTS

Sequence name	η	q	p	ε	n
Ants	5.37	0.0023	0.02	0.0022	9482
Mice	5.39	0.0678	0.05	0.0638	310
Shop	3.32	0.0093	0.05	0.0053	6560

q : mean of global motion statistics (estimated motion intensity)

η : estimation based on the substitution of p and q into (3.22)

ε : based on q and η, it takes into account the condition $\eta > 2$

n : necessary frame count by substituting values into (3.24)

In Table 4-IV, the estimate of p is based on subjective judgment and q is the mean of global motion statistics. The low value of p is realistic, because it is related to the full image area. Furthermore, the capture rate was 20 fps in each sequence, and this is probably the reason for the relatively low q (motion intensity). The value η is based on (3.22) by using the value of q. It confirms that the parameters of GMM can be established.

Finally, Figure 4.11 summarizes the results of experiments on the "Shop" sequence at different processed frame counts. As can be seen, the value of Table 4-IV is close to the true required frame count. In Figure 4.11, we see that when the processed sequence length is larger than 6000 frames the VP is near to the true value. The oscillation of the motion intensity can be seen in Figure 4.11(b), which demonstrates that in the outdoor video the motion was not uniform. This caused the result of VP estimation to oscillate as well, although in practice this effect is fairly inconspicuous (~0.1°). We also found the same results on the other two sequences.

DOI:10.15774/PPKE.ITK.2024.006

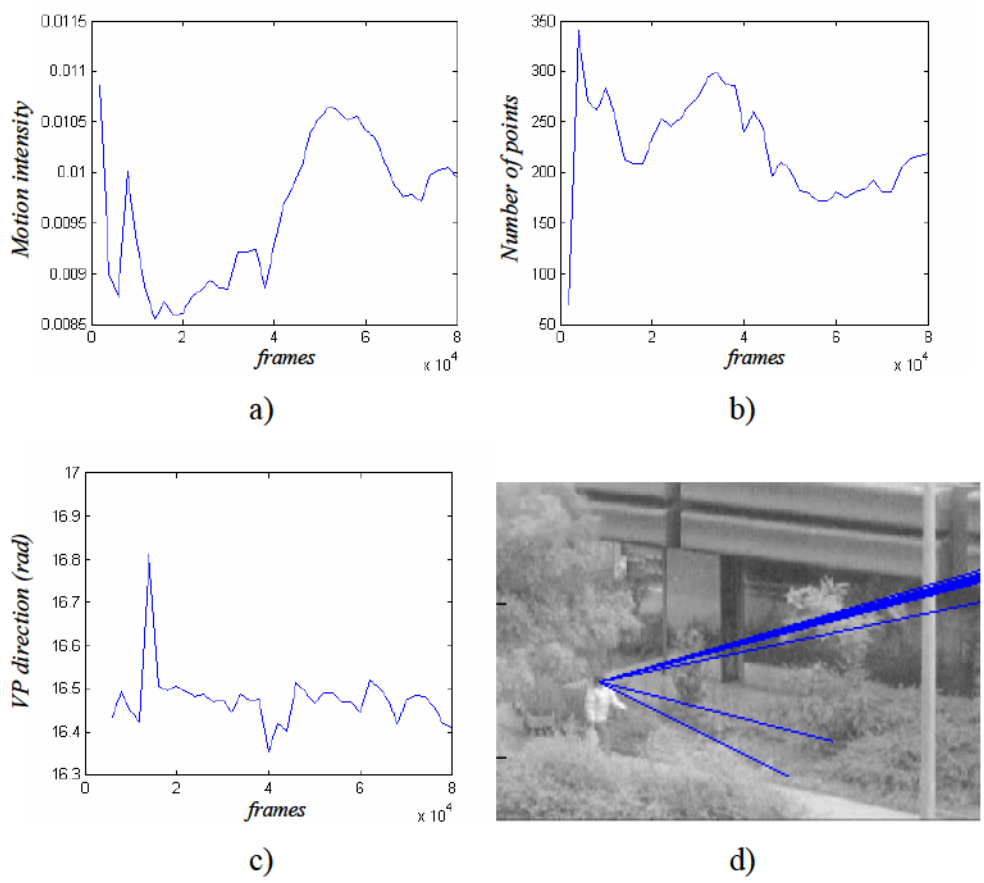


Figure 4.11: Results on the varying processed video length (in frames) of the "Shop" sequence. a) Global motion intensity; b) Number of extracted correspondences; c) Parameter of VP (in degrees; the true value is 16.5°); d) The epipolar lines. The convergence to the valid VP is also visible. In d) the results are displayed after 2000, 4000, 6000, ... frames, up to 80000 processed frames.

4.3. Computation of the vanishing line

In summary, the determination of the vanishing line is possible with knowledge of at least three corresponding line segments. These line segments can be computed from the apparent height of the same object as seen at different positions (depths) on the ground-plane. The objects may for instance be pedestrians [45], and the line segments denote their height. However, the precise detection of such non-rigid objects is a challenging task in outdoor images. In our framework the necessary height information can be easily determined from the local statistics. Because the statistics are generated from moving object masks, its model parameter – namely the covariance matrix – is the estimation about the average size of potential objects. The information derived from statistics is valid only if the following assumption is

DOI:10.15774/PPKE.ITK.2024.006

satisfied: there are regions where the same objects are moving with equivalent probability (e.g. pathway or road).

4.3.1. Height estimation of average shapes

The following figure illustrates the results of motion statistics in both indoor and outdoor sequences.

Figure 4.12: Sample frames in upper row, and raw motion statistics in the bottom row. The corresponding point is marked by 'x'

From this feature extraction the input for the further processing steps is the parameters of the covariance matrix $\mathbf{\Sigma}_{\mathbf{x}}$ in point $\mathbf{x}$ (see (3.9)). The dimensions and orientation of the average shape come from the eigen-value decomposition of the covariance matrix:

$$\mathbf{\Sigma}_{\mathbf{x}}\mathbf{v}_{\mathbf{x},i} = \lambda_{\mathbf{x},i}\mathbf{v}_{\mathbf{x},i} \quad i = 1,2 \tag{4.10}$$

These statistical characteristics are displayed in Figure 4.12.

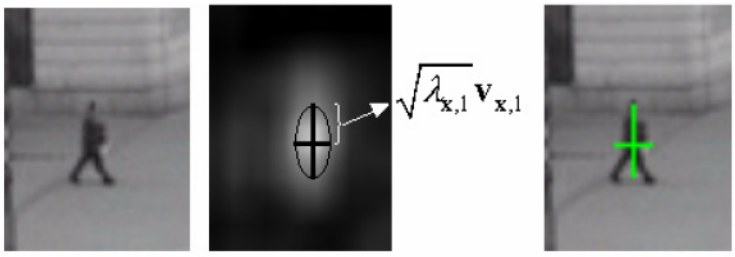


Figure 4.13: Example to shape properties: axes of normal distributions, derived from the eigen-value decomposition of the covariance matrix.

Finally, the height measurement comes from the projection (vertical component) of the most vertical eigenvector:

DOI:10.15774/PPKE.ITK.2024.006

$$\left(\lambda_{\mathbf{x},\max}, \mathbf{v}_{\mathbf{x},\max}\right) = \arg\max_{(\lambda,\mathbf{v})}\left(\lambda_{\mathbf{x},1}\left\langle \mathbf{e}, \mathbf{v}_{\mathbf{x},1}\right\rangle, \lambda_{\mathbf{x},2}\left\langle \mathbf{e}, \mathbf{v}_{\mathbf{x},2}\right\rangle\right), h_j = h_{\mathbf{x}} = \sqrt{\lambda_{\mathbf{x},\max}}\left\langle \mathbf{e}, \mathbf{v}_{\mathbf{x},\max}\right\rangle \tag{4.11}$$

where $\mathbf{e}$ denotes the vertical unit vector: $\mathbf{e} = \begin{bmatrix} 0 & 1 \end{bmatrix}$ and $\langle . \rangle$ is the dot product, respectively. These height estimations are displayed in Figure 4.14. For the sake of later simplification we transform the indices from vector (coordinate) form e.g. $\mathbf{x}$ to a simple scalar index j. Henceforward, j denotes a point in the image; viz. h_j is the height measurement in image-point j and vector $\mathbf{p}_j$ determines the coordinates of point j in the image. Because the scheme (4.11) utilizes information extracted from statistics, a more sophisticated form may be given for the height estimation which takes into account the uncertainty:

$$P\left(\hat{h}_j \middle| h_j\right) = \mathrm{N}\left(\hat{h}_j, h_j, \Sigma_{\Delta h_j}\right) \tag{4.12}$$

where

$$\Sigma_{\Delta h_j} = \sigma^2_{\Delta h_j} = \left(\sqrt{\lambda_{j,\max}} - h_j\right)^2 \tag{4.13}$$

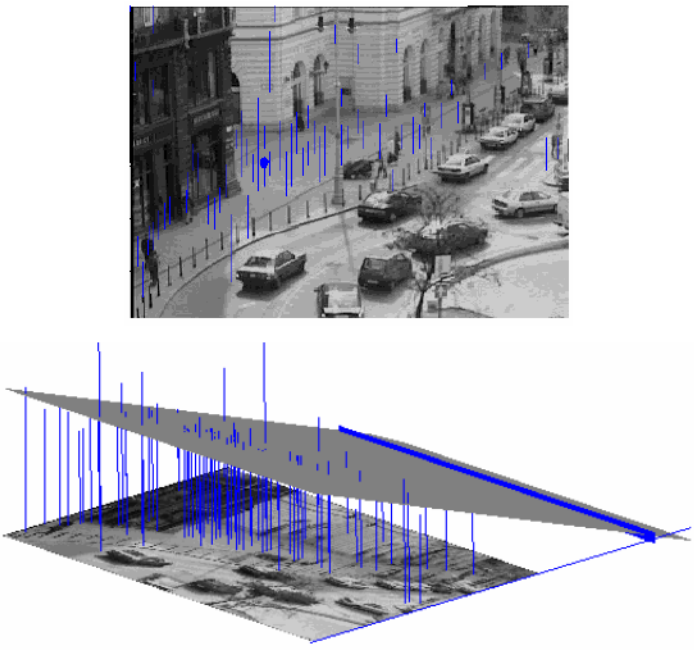


Figure 4.14: Samples from height estimations in outdoor environment.

DOI:10.15774/PPKE.ITK.2024.006

4.3.2. Outlier rejection and error propagation

Outlier rejection

In general, without making any prior assumptions about the scene every point may be paired to every other point. But the practical processing of this huge data-set requires that we have an effective way to drop "outlier" points and extract information for VL estimation.

First, we describe simple conditions which can be used to reduce the size of the data-set. The outlier rejection in this case is similar to dropping points where two objects are moving but are not the same size. Let j represents an arbitrary point in the image and k denotes another (corresponding) point: $j \neq k$

We reckon two points as corresponding points (which is probable, where same-sized objects are concerned) if

$$\sigma_1 < \frac{\lambda_{j,1}}{\lambda_{j,2}} / \frac{\lambda_{k,1}}{\lambda_{k,2}} < \sigma_2 \tag{4.14}$$

and

$$\Phi\left(\mathbf{v}_{j,1}, \mathbf{v}_{k,1}\right) < \alpha \tag{4.15}$$

where the notations come from the eigenvalue decomposition of the covariance matrices of two points, see (4.10), and $\Phi(.)$ denotes the angle between two vectors (the deviation of eigenvectors in our case). These simple conditions lead to a set of points where the objects have similar orientation and aspect ratio. Figure 4.15 demonstrates the point set (marked by circles) corresponding to a point (marked by x).

Figure 4.15: Corresponding points (marked with circles) are related to an arbitrary image point (marked by large 'x').

DOI:10.15774/PPKE.ITK.2024.006

After this preprocessing every point will have several probable corresponding point-pairs. However, several outliers remain, thus we have to use all points to determine vanishing points and an estimation about horizon.

Error propagation

An initial guess about the horizon can be computed using the height information of corresponding points to an arbitrary point j:

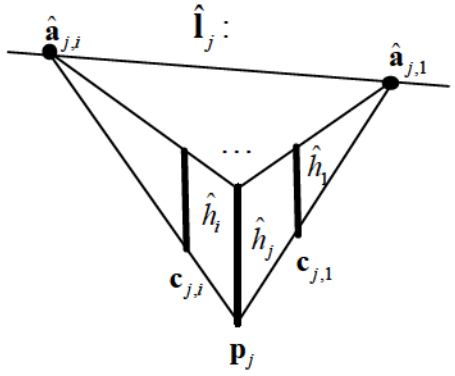


Figure 4.16: Using vertical size information to get the horizon ($\hat{\mathbf{l}}_j$) and vanishing points $\hat{\mathbf{a}}_{j,i}$. The 2D point $\mathbf{p}_j$ is an arbitrary image point, while $\mathbf{c}_{j,1}$ and $\mathbf{c}_{j,i}$ are two samples for corresponding points.

To simplify the further computations the transformation between height information and the 2D image plane is necessary. We have to compute point coordinates in the ground-plane, as it is demonstrated in the following figure.

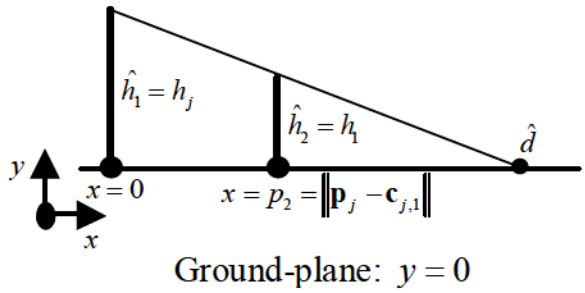


Figure 4.17: Determination of a vanishing point, which in ideal case lies in the horizontal vanishing line (horizon). The task may be summarized as the computation of $\hat{d}$ taking into account the inaccuracy of height measurements.

The determination of $\hat{d}$ without uncertainty comes from elementary algebra:

$$\hat{d} = \hat{h}_1 \frac{p_2}{\hat{h}_1 - \hat{h}_2} \tag{4.16}$$

DOI:10.15774/PPKE.ITK.2024.006

To derive a formula which contains the uncertainty – based on the method described in [47] – we define the relationship between the input and the output quantity in an implicit form. For this scheme we define the ideal input vector $\mathbf{X}$ and the observed vector $\hat{\mathbf{X}}$. The ideal parameter vector $\boldsymbol{\Theta}$ and the observed $\hat{\boldsymbol{\Theta}}$, respectively. The $\hat{\boldsymbol{\Theta}}$ and $\hat{\mathbf{X}}$ are related through an optimisation function $F(.)$, and $\hat{\boldsymbol{\Theta}}$ is determined by minimising $F\left(\hat{\mathbf{X}},\hat{\boldsymbol{\Theta}}\right)$. In this phase of our method the input measurements are height information about the objects, while the output is the estimated position of the intersection of ground plane and the line through points $\left(0,\hat{h}_1\right)$ and $\left(p_2,\hat{h}_2\right)$, see Figure 4.17. This line-plane intersection determines one point, accordingly the input vector is

$$\hat{\mathbf{X}}=\left[\hat{h}_1,\hat{h}_2\right] \tag{4.17}$$

and the observation is

$$\hat{\boldsymbol{\Theta}}=\left[\hat{d}\right] \tag{4.18}$$

The analytic curve function expressed as

$$F\left(\hat{\mathbf{X}},\hat{\boldsymbol{\Theta}}\right)=\hat{h}_1\left(p_2-\hat{d}\right)-\hat{h}_2\left(p_1-\hat{d}\right)=0 \tag{4.19}$$

Error propagation relates the uncertainty of input measurements to the perturbation of $\hat{\boldsymbol{\Theta}}$. Let $\boldsymbol{\Sigma}_{\Delta X}$ be the covariance matrix of measurements:

$$\boldsymbol{\Sigma}_{\Delta X}=\begin{bmatrix}\sigma_{\Delta X}^2 & 0\\ 0 & \sigma_{\Delta X}^2\end{bmatrix} \tag{4.20}$$

where

$$\sigma_{\Delta X}^2=\frac{\sigma_{\Delta h_j}^2+\sigma_{\Delta h_k}^2}{2} \tag{4.21}$$

Based on the covariance propagation theory [48], we have

$$\boldsymbol{\Sigma}_{\Delta\Theta}=2\sigma_{\Delta X}^2\left[\left(\frac{\partial g}{\partial\boldsymbol{\Theta}}\right)^T\right]^{-1} \tag{4.22}$$

where $\dfrac{\partial g\left(\mathbf{X},\boldsymbol{\Theta}\right)}{\partial\boldsymbol{\Theta}}$ is defined as

DOI:10.15774/PPKE.ITK.2024.006

$$\frac{\partial g}{\partial \boldsymbol{\Theta}} = 2\frac{\left(\frac{\partial F}{\partial \boldsymbol{\Theta}}\right)^2}{\left(\frac{\partial F}{\partial h_1}\right)^2 + \left(\frac{\partial F}{\partial h_2}\right)^2} \tag{4.23}$$

Thus, we have

$$\boldsymbol{\Sigma}_{\Delta\Theta} = \sigma_{\Delta\Theta}^2 = \sigma_{\Delta X}^2 \frac{\left(p_2 - \hat{d}\right)^2 + \hat{d}^2}{\left(\hat{h}_2 - \hat{h}_1\right)^4} \tag{4.24}$$

The result is illustrated in the following figure.

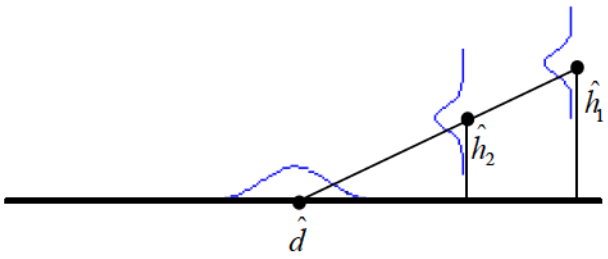


Figure 4.18: Simulation of error propagation from input data (height estimations) into 1D position coordinate. The two uncertainty heights are used to determine the intersection of line through these points and the x axis. The formula for uncertainty of this intersection was expressed by (4.24).

Finally, we have to convert the result of (4.24) into the 2D image plane. This conversion can be accomplished by constructing a 2D covariance matrix:

$$\boldsymbol{\Sigma}_{\Delta VP_{j,i}} = \mathbf{U}^T \begin{bmatrix} \sigma_{\Delta\Theta}^2 & 0 \\ 0 & 0 \end{bmatrix} \mathbf{U} \tag{4.25}$$

where $\mathbf{U}$ is the matrix of eigen-vectors (Note that, $\mathbf{U}\mathbf{U}^T = \mathbf{I}$.):

$$\mathbf{U} = \begin{bmatrix} \mathbf{v}_i \\ \tilde{\mathbf{v}}_i \end{bmatrix} \text{ and } \left\langle \mathbf{v}_i, \tilde{\mathbf{v}}_i \right\rangle = 0 \tag{4.26}$$

with eigen-vectors formed from the unit length vector through points $\mathbf{p}_j$ and $\mathbf{c}_{j,i}$:

$$\mathbf{v}_i = \frac{\mathbf{c}_{j,i} - \mathbf{p}_j}{\left\| \mathbf{c}_{j,i} - \mathbf{p}_j \right\|} \tag{4.27}$$

While the centroid (position of vanishing point defined by points j and i) is determined from the estimated distance $\hat{d}$ along the line with direction $\mathbf{v}_i$:

$$\hat{\mathbf{a}}_{j,i} = \mathbf{p}_j + \mathbf{v}_i \hat{d} \tag{4.28}$$

DOI:10.15774/PPKE.ITK.2024.006

Thus, we have the formula for probability density of measurement noise:

$$P\left(\hat{\mathbf{a}}_{j,i}\middle|\mathbf{a}_{j,i}\right)=\mathrm{N}\left(\hat{\mathbf{a}}_{j,i},\mathbf{a}_{j,i},\boldsymbol{\Sigma}_{\Delta VP_{j,i}}\right) \tag{4.29}$$

4.3.3. Optimization procedure in Hough space

Measurement conversion into Hough space

After the evaluation of the error propagation formula to every corresponding point pair we will have several uncertain 2D point coordinates. These estimations represent an initial guess about horizon, since the inliers of the data-set lie in the horizon. This line estimation problem is well known and there are several approaches to solve it in various cases: e.g. least squares (LS), total least squares (TLS) and Hough transformation [49][36].

- In short, our case has the following special properties:
- Error in both coordinates in the 2D plane (x and y).
- There is correlation between the noise in the two coordinates.
- The noise covariance matrices are different for different data points (heteroscedastic noise).
- Notable amount of outliers can be found in the dataset.

Because of these specific characteristics the line fitting is viewed as a global optimisation procedure. Generally, there is no analytic solution for the cases of heteroscedastic and correlated noise, where we assume that the noise in x is correlated to the noise in y, furthermore, the variance of the noise is not identical for all data points. Heteroscedastic regression problem in computer vision is studied in [49]. Both LS and TLS methods fail when the data-set contains outliers. Line fitting on such data-set needs a robust estimator, for survey see [36]. The Hough transform is an effective and popular way for line-fitting [50]. In the standard version, an accumulator array is used to collect the points which lie along the same line. The line is parametrized by (θ,ρ):

$$\rho = x\cos(\theta)+y\sin(\theta) \tag{4.30}$$

In this section the error propagation will be continued, and an optimal line parameter has been determined by using non-linear optimisation procedure. Because the Hough transformation generates sinusoidal voting patterns in the parameter space we will not

DOI:10.15774/PPKE.ITK.2024.006

use the same error propagation formula as in the previous section. In the end of the section a simple formula for the error estimation in the parameter space will be given. Let 2D point $\mathbf{a}_{j,i} = (x_i, y_i)$ be the unknown accurate position of the i[th] vanishing point introduced in the previous section. The measurements are $\hat{\mathbf{a}}_{j,i} = (\hat{x}_i, \hat{y}_i)$ based on the error propagation formula. Due to noise, $\mathbf{a}_{j,i} \neq \hat{\mathbf{a}}_{j,i}$. The probability density of measurement noise is modelled as a 2D heteroscedastic Gaussian, with correlated noise in formula (4.29). We define the line-fitting task as finding the maximum of the objective function:

$$\hat{\mathbf{l}}_j = \arg\max_{\mathbf{l}\in(\theta,\rho)} \sum_i P_g(\mathbf{p}_i) C_{j,i}(\mathbf{l}) \tag{4.31}$$

where the maximum value of probability (4.29) along the line $\mathbf{l}$ is defined by:

$$C_{j,i}(\mathbf{l}) = \max_{\mathbf{u}\in\mathbf{l}} P\left(\mathbf{u}\middle|\mathbf{a}_{j,i}\right) \tag{4.32}$$

(This line is also parametrized by $\hat{\mathbf{l}}_j = (\hat{\theta}_j, \hat{\rho}_j)$.)

The optimum value is determined by *unconstraint non-linear optimisation* of (4.31), the initial estimate is given by LMS method. The introduced formula handles the outliers, thus there is no need for robust M-estimator, where the error expressions are replaced by some saturation function [36]. We note that, the computation of (4.32) is simple; it has analytic solution, see [49] for details. Since the residual outliers cause an error in line-fitting, we define the error in line-fitting with a 2D Gaussian:

$$P\left(\hat{\mathbf{l}}_j\middle|\mathbf{l}_j\right) = \mathrm{N}\left(\hat{\mathbf{l}}_j, \mathbf{l}_j, \Sigma_{\Delta VL_j}\right) \tag{4.33}$$

where the covariance matrix is defined as

$$\mathbf{\Sigma}_{\Delta VL_j} = \begin{bmatrix} \tan^{-1}\left(\frac{\Delta l_Y}{\Delta l_X}\right)^2 & 0 \\ 0 & \Delta l_X{}^2 \end{bmatrix} \tag{4.34}$$

The notations are detailed in the following figure.

DOI:10.15774/PPKE.ITK.2024.006

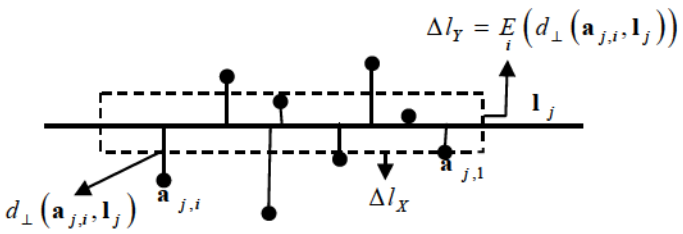


Figure 4.19: Demonstrating the parameters for the expression of line-fitting error (see (4.34)) in parameter space.

The function $d_\perp(.)$ computes the distance between the line $\mathbf{l}_j$ and point $\mathbf{a}_{j,i}$, while the expected value denoted by $E(.)$.

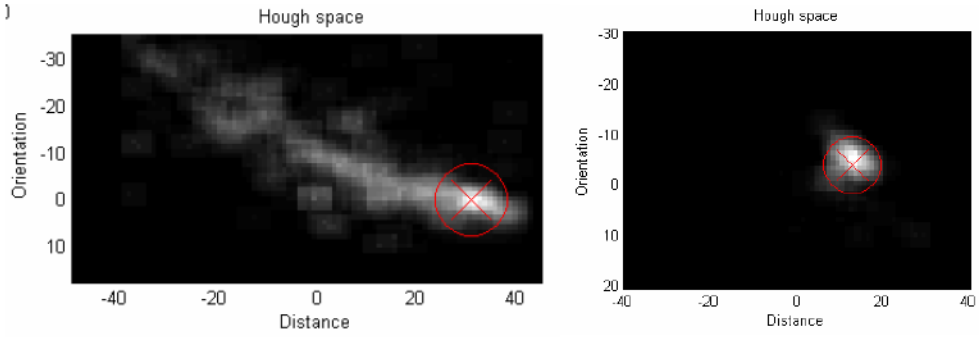
Thus, the guess about the horizon at point j is determined by (4.33) which describes uncertainty in the parameter space (2D Hough-space).

Final optimisation

The estimation about the horizon and the estimation error are attached to several points in the image. The accurate determination of horizon is carried out in parameter space using all estimations:

$$\mathbf{l}_h = \arg\max_{\mathbf{l}\in(\theta,\rho)} \sum_i P_g\left(\mathbf{p}_i\right) P\left(\mathbf{l}\middle|\mathbf{l}_i\right) \tag{4.35}$$

It has been fulfilled with the same optimisation technique as in previous section. The following figure displays the 2D parameter space which has been filled with numerically computed values of (4.35).



a) b)

Figure 4.20: The picture in a) depicts the Hough space of outdoor scene, while b) relates to indoor scene, respectively. The selected point is related to the most probable parameters of the horizon.

DOI:10.15774/PPKE.ITK.2024.006

4.3.4. Experimental results

We performed a practical evaluation of the method in which both indoor and outdoor videos were used as input. The parameters introduced in the previous sections are assigned the following values in empirical fashion: $\sigma_1 = 0.8$ and $\sigma_2 = 1.25$ in (4.14), while $\alpha = 10^\circ$ in (4.15). To determine the binary motion mask ($m(t,\mathbf{x})$) a motion-detection method was used which is based on the background model introduced by Stauffer [30].

The manual extrapolation of the vanishing line is a difficult task, because: i) there are not enough static features for accurate alignment; and ii) the objects are usually too small in case of outdoor images. The outdoor video used for testing shows not only pedestrians, but cars as well; this is why the parameter configurations (distance and orientation of the horizon line) in Hough space show scatter, see Figure 4.20(a). The deviation from optimal parameter values is much smaller in indoor case, see Figure 4.20(b).

The results demonstrated by straight line in 2D coordinate space after final optimisation of Hough space are displayed in Figure 4.21.

Figure 4.21: Horizon computation in indoor and outdoor videos.

4.4. Conclusions

A camera registration method has been presented which uses walk-parameters as features to identify corresponding points. The features we used (concurrent walk-steps, leading-leg identity and 2D motion vector) seem potentially to provide good

DOI:10.15774/PPKE.ITK.2024.006

data for the estimation of homography between two different camera views of the same scene and an occurring configuration of non-overlapping views. The registration method has been verified on an actual indoor camera surveillance system, and was able to provide real-time feature (walk) detection. This efficient camera registration proves the accuracy of the localization of our gait features.

We have shown that using the proposed algorithm it is feasible to compute the horizon with good accuracy even from a real-life noisy data set which contains several outliers. The proposed approach executes two statistical parameter optimization steps by using the benefits of error propagation formula.

DOI:10.15774/PPKE.ITK.2024.006

5.Improved extraction of foreground image mask

Moving object detection is a key issue in most computer vision applications especially for surveillance purposes. Depending on the scene settings the cast shadow usually generates problems while extracting moving objects (e.g. silhouettes). The problem occurs often in outdoor scenes and indoor configurations when the floor is a reflective surface. In most cases shadow can cause merging of objects, shape distortion and object losses. Thus, shadow detection is critical for accurate object detection, which is a relevant step of information extraction for further processing: tracking, event detection [68] or traffic monitoring [63].

This section focuses on the classification of motion mask. We will show that the use of geometrical model and statistical motion information can be integrated. The amount of pixels related to shadow and reflection is significantly reduced in the final foreground image mask. This improved foreground mask is a good basis for further processing steps.

DOI:10.15774/PPKE.ITK.2024.006

5.1. Introduction

Many approaches have been proposed in the literature that deal with shadow. A good survey can be found in [69][29]. Most of the publications are focused on the colour based shadow detection [63][69][31][71]. In order to remove shadow points, these methods have defined conditions in some proper colour space.

Method, called SAKBOT, was introduced in [63]. It was developed for moving object detection and tracking. This complex algorithm contains both colour and motion information. Additionally, the final foreground mask is improved with knowledge-based feedbacks. The method introduced in [70] utilizes some predefined object model present estimation about shadow pixels near to the detected objects. Summarising, the basic features that can be used to distinguish between shadow and object points are: colour, texture, motion. In spite of the notable amount of publications there is no approach which utilizes the general geometrical model of cast shadow and includes the geometrical characteristics into the classification process. Geometrical information is important in describing the creation of shadow. Like other characteristics, the geometrical description is not a unique feature, so without other features it is not sufficient for accomplishing of classification purposes in all cases.

5.2. Detection of reflections in Bayes inference

In this section we present a possible application of the determined VP and co-motion statistics. We demonstrate the use of the derived geometric model together with the co-motion statistics for the purpose of refining the classification of the foreground elements of the scene (the foreground mask). The essence of this task is the removal of those pixels from the foreground mask which correspond to reflection.

Firstly, we should explain why knowledge of the VP is not enough by itself for us to solve this classification problem effectively. This is because of the fact that, based on the model, only the fundamental constraint can be used, which is a point-line transformation. This means that for the identification of the reflection related to an arbitrary point we have to scan along the line and try to find the reflection of some (unknown) object by using a suitable correlation measure. This is a challenging task

DOI:10.15774/PPKE.ITK.2024.006

in case of low-detailed reflections; but in any case, this searching demands considerable computation time.

We shall not discuss all related issues on decision theory; we concentrate on a simple presentation on the decision as to whether or not a given point is to be considered as part of a reflection. The classification method we apply is based on the Bayes decision rule [73]. Several other (more complex) methods are also available for classification; for details see [73]. We will show how the geometric model and the statistics can be included into the class-conditional density function. Consider the two classes: reflection, and foreground $L=\{fg,rf\}$. Points in the motion mask belong to the classes $\Upsilon=\{\vartheta_{\mathbf{x}}|\mathbf{x}\in S,\vartheta_{\mathbf{x}}\in L,m_t(\mathbf{x})=1\}$ with the *a prior* probabilities

$$P(\vartheta_{\mathbf{x}}=fg)=P_{fg}(\mathbf{x}|F) \tag{5.1}$$

for occurrence of the foreground class, and

$$P(\vartheta_{\mathbf{x}}=rf)=P_{rf}(\mathbf{x}|F) \tag{5.2}$$

for occurrence of the reflection class (these probabilities were derived in the previous section, see (4.7) and (4.8). Furthermore, the decision rule may be written in the form: assign $\mathbf{x}$ to foreground class ($\vartheta_{\mathbf{x}}=fg$) if

$$P(\mathbf{x}|\vartheta_{\mathbf{x}}=fg)P(\vartheta_{\mathbf{x}}=fg)>P(\mathbf{x}|\vartheta_{\mathbf{x}}=rf)P(\vartheta_{\mathbf{x}}=rf) \tag{5.3}$$

Thus, the density function for a foreground pixel is formulated as

$$P(\mathbf{x}|\vartheta_{\mathbf{x}}=fg)=\max_{\mathbf{r}\in S}P_e(\mathbf{r}|\mathbf{x},F)P_{refl}(\mathbf{r}|\mathbf{x})m_t(\mathbf{r}) \tag{5.4}$$

This expression takes into account the possibility that a foreground pixel may have a reflection; this information comes from (4.6) and is symbolized with the term $P_{refl}(.)$. The first component in (5.4) is related to the geometric model, and is defined by

$$P_e(\mathbf{r}|\mathbf{x},F)=\mathrm{N}(\tilde{\mathbf{r}}^TF\tilde{\mathbf{x}},\mu_0=0,\sigma_0\approx 2) \tag{5.5}$$

It determines a line (and its surroundings) from a given point $\mathbf{x}$ through the VP. The use of a normal distribution (defined in (3.10)) assists us to increase the robustness; this is because we do not use a line with "one pixel" thickness, but rather a line with a thickness of σ_0 which enables error in VP position. Accordingly, (5.4) is equivalent to the probability that expresses that $\mathbf{x}$ has a reflection somewhere in the image at

DOI:10.15774/PPKE.ITK.2024.006

frame t. Based on the above discussion, the class-conditional function for the "reflection" class is given by

$$P\left(\mathbf{x}\middle|\vartheta_{\mathbf{x}}=rf\right)=\max_{\mathbf{r}\in S}P_e\left(\mathbf{r}\middle|\mathbf{x},F\right)P_{orig}\left(\mathbf{r}\middle|\mathbf{x}\right)m_t\left(\mathbf{r}\right) \tag{5.6}$$

Some of the 2-D probabilities and the classification results are demonstrated in Figure 5.1. Note that there are some cases when we should not make any decision during classification [73] (e.g. those points where there is no reflection). In these unclassifiable cases the products in (5.3) are conspicuously low. To eliminate these points we introduce a threshold value; it was determined experimentally that 10^{-6} is a suitable order of magnitude for this threshold for all test sequences.

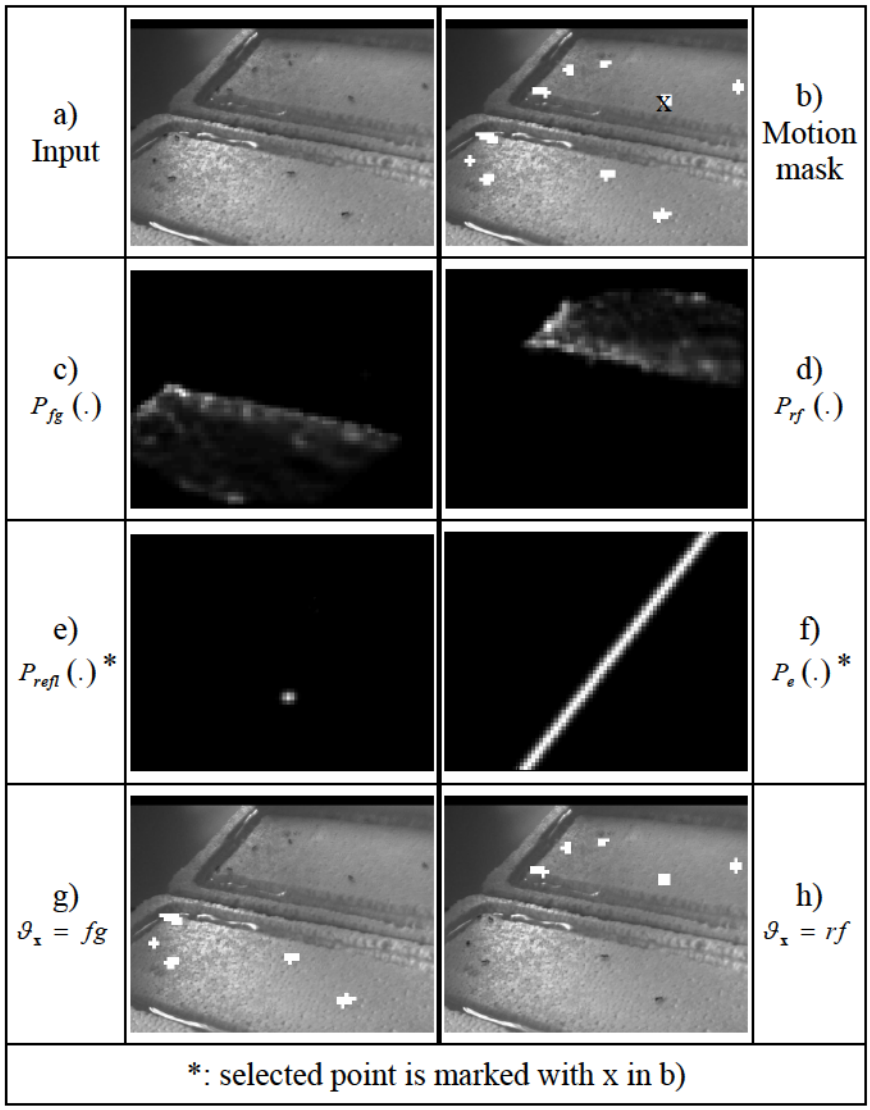


Figure 5.1: Main steps of the classification process which supports the removal of reflections from the foreground mask. For details see text.

DOI:10.15774/PPKE.ITK.2024.006

Finally, we have to discuss the effect of the outlier rejection steps. The resulting subset of image points (S_3) and the image scaling result in the fact that there will be points that do not have valid co-motion statistics. In the case of these points, there is need for interpolation by using the available valid statistics of the neighborhood points. In the classification step we have to estimate the $P_{refl}(\mathbf{r}|\mathbf{x})$ and $P_{orig}(\mathbf{r}|\mathbf{x})$ with the following formulas:

$$\tilde{P}_{refl}(\mathbf{r}|\mathbf{x}) = \frac{\sum_{\mathbf{u}\in C} P_g(\mathbf{u}) P_{orig}(\mathbf{x}|\mathbf{u}) P_{refl}(\mathbf{r}|\mathbf{u})}{\sum_{\mathbf{u}\in C} P_g(\mathbf{u})} \tag{5.7}$$

$$\tilde{P}_{orig}(\mathbf{r}|\mathbf{x}) = \frac{\sum_{\mathbf{u}\in C} P_g(\mathbf{u}) P_{orig}(\mathbf{x}|\mathbf{u}) P_{orig}(\mathbf{r}|\mathbf{u})}{\sum_{\mathbf{u}\in C} P_g(\mathbf{u})} \tag{5.8}$$

Where C is the set of the nearest neighbors of $\boldsymbol{u}$ where the co-motion statistics are valid. This missing information can be computed straight away after the model optimization, and thus it will not reduce the performance during classification.

5.2.1. Experimental results

Based on the manual validation, we have found that the error rate of foreground extraction was reduced. The proposed classification evaluated only points which were detected as foreground by [12]. The performance is characterized by the measures proposed in [29]: 'Detection Rate – DR' and 'False Alarm Rate – FAR'. These values are obtained as follows: DR=TP/(TP+FN) and FAR=FP/(TP+FN), where TP is the number of correctly detected object's pixels, FN the missed object's pixels, and FP the reflection pixels incorrectly detected as object's pixels. The DR and FAR rates for three sequences are shown in Table 5-I.

TABLE 5-I: THE DRS AND FARS FOR THREE VIDEO SEQUENCES

Sequence name	DR	FAR
Ants	0.991	0.031
Mice	0.892	0.075
Shop	0.822	0.120

The fundamental limitation of the classification procedure is that it is usually unable to disjoint the motion mask in cases when the real object mask and its reflection are linked. Figure 5.2: illustrates such situations. In summary, the results are promising in

DOI:10.15774/PPKE.ITK.2024.006

spite of the fact that the applied classification method is very simple. Some of the more sophisticated methods (e.g. MRF [31]) are able to increase the accuracy; but only with substantially higher computation cost.

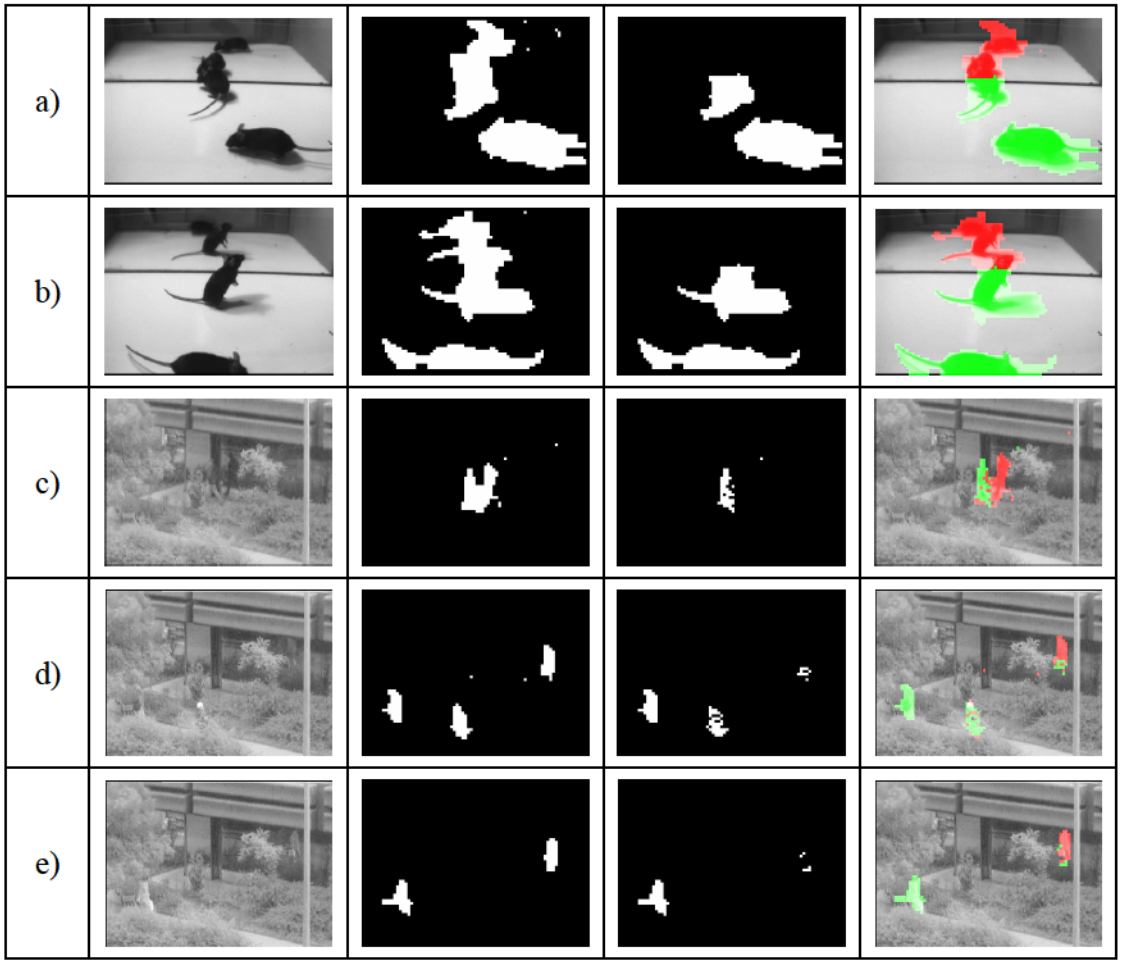


Figure 5.2: Challenging situations of foreground segmentation in scenes from the "Mice" and "Shop" sequences. In the detected motion mask for (a), (b) and (c), the object fuses with its reflection. The proposed method is able to remove only a small part of the reflection.

DOI:10.15774/PPKE.ITK.2024.006

5.3. Shadow removal using Bayesian iteration

The goal of shadow detection is to eliminate the shadow points from the extracted foreground mask. The foreground mask can be determined using several approaches. Our implementation based on [30] which is a popular background modelling method for the extraction of foreground mask. The output mask is defined by:

$$m_i = \begin{cases} 1, \text{ where change is detected} \\ 0, \text{ otherwise} \end{cases} \qquad (5.9)$$
$$M = \{m_i, i \in S\}$$

where S denotes all pixels in the image (index i corresponds to one pixel). In the followings I determines the input image and B denotes the computed background image, respectively. Intensity changes cause changes in object pixels as well as in cast shadow pixels.

Thus, the initial foreground-background mask M contains both object and shadow pixels as foreground.

The usual method to distinguish between moving cast shadow and object points is the investigation of pixels in Hue-Saturation-Value (HSV) colour space [63][31]. This pre-processing step is a simple filtering before higher level processing. We focus on strong shadow in outdoor environment, thus we have implemented only the condition related to Value (V):

$$c_i = \begin{cases} 1, \text{ if } \alpha \le \dfrac{V(I_i)}{V(B_i)} \le \beta \wedge V(I_i) < 0.6 \\ 0, \text{ otherwise} \end{cases} \qquad (5.10)$$
$$C = \{c_i, i \in S\}$$

Here C denotes a shadow mask with elements c_i equal "1" for shadow pixels and "0" otherwise, according to colour based conditions: α and β are bounds obtained from experiments.

In our experiments this colour based method works reliable only in case of weak shadow. If the shadow is strong, the ratio is around 0.4. Unfortunately, however the ratio changes to 0.9 near to the boundary of shadow. That is why shadow elimination

DOI:10.15774/PPKE.ITK.2024.006

is not possible by using only the colour information. Figure 5.3 displays the histogram of the ratio inside the manually selected shadow mask.

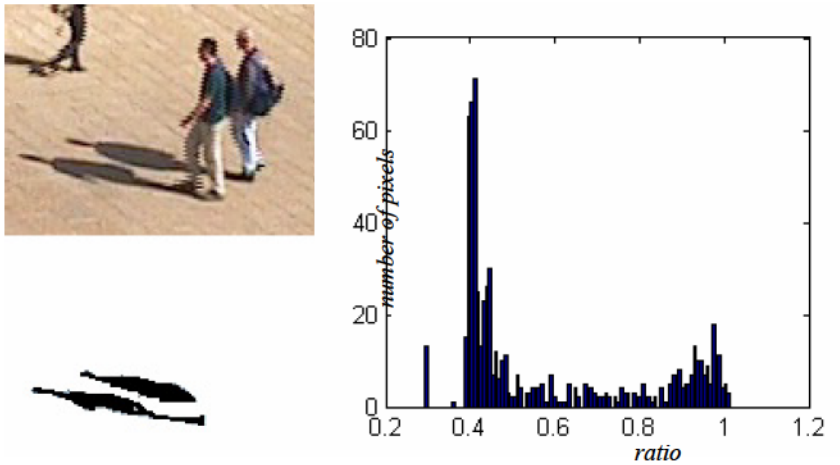


Figure 5.3: Difficulty of colour based shadow detection in case of strong shadow: the shadow region has not a histogram with only one peak, and thresholds α and β are not the same for the whole image.

Another problem is that the adaptive determination of the bounds α and β is still a challenging task. Thus, in our implementation these values were adjusted to cover a relatively large region of the full range (α=0.4, β=0.8). Detection results are summarized in the following figure.

Figure 5.4: Results of colour based shadow detection: upper left-input image, upper right-motion-detection mask, lower left-foreground mask determined by using colour features and lower right-"worse-case" shadow mask (α=0.4, β=0.8) used for input to classification method. (The binary masks are without morphological post-processing.)

5.3.1. Outline of the iteration scheme

The aim of classification is to decide about every foreground pixel in the initial foreground-background mask (M) whether it is a foreground pixel or a shadow pixel.

DOI:10.15774/PPKE.ITK.2024.006

This two-classes problem is equivalent to find the probable class for an arbitrary pixel in the given scene setting (M and C). In the literature there are several different approaches to accomplish such classification tasks [73]. Nevertheless, in this section a simple Bayesian iteration will be introduced. We selected this probabilistic framework, because it is rather general and is suitable for further improvements. This method was used successfully for blind deconvolution in [75] and [74].

We define the unknown shadow mask as

$$H = \{h_i, i \in S\} \tag{5.11}$$

and foreground mask as

$$F = \{f_i, i \in S\} \tag{5.12}$$

Together with definition of the detected initial foreground-background mask M (5.9), using Bayes conditional probability formula we can get the probability of observing H and F with given M in the following form (the formulas for F are similar):

$$P(h_i|m_j) = \frac{P(m_j|h_i)P(h_i)}{\sum_k P(m_j|h_k)P(h_k)}, \quad i, j \in S \tag{5.13}$$

Substituting this equation into the conditional probability formula, we get:

$$\begin{aligned} &P(h_i) = \sum_j P(h_i m_j) = \sum_j P(h_i|m_j)P(m_j) = \\ &\sum_j \frac{P(m_j|h_i)P(h_i)P(m_j)}{\sum_k P(m_j|h_k)P(h_k)}, \quad i, j \in S \end{aligned} \tag{5.14}$$

Based on this formula the following iteration scheme can be written [12]:

$$\begin{aligned} &P_{k+1}(h_i) = \\ &P_k(h_i)\sum_j \frac{P(m_j|h_i)P(m_j)}{\sum_k P(m_j|h_k)P(h_k)}, \quad i, j, k \in S \end{aligned} \tag{5.15}$$

Where k is the iteration counter. We define the initial probabilities as follows,

$$P(m_i) = \begin{cases} 1, \text{ where } m_i = 1 \\ 0, \text{ otherwise} \end{cases}, i \in S \tag{5.16}$$

and

DOI:10.15774/PPKE.ITK.2024.006

$$P(h_i)=\begin{cases}1/2, \text{ where } c_i=1\\ 0, \text{ otherwise}\end{cases}, i\in S$$
$$P(f_i)=1-P(h_i) \tag{5.17}$$

During the iteration steps the values of $P(f_i)$ and $P(h_i)$ will change. These probabilities converge to stable values which describe the most probable shadow/foreground configuration in a given motion mask. Hereby the classification step is a simple decision to the most probable class:

$$\hat{f}_i=\begin{cases}1, \text{ if } P(h_i)<P(f_i)\\ 0, \text{ otherwise}\end{cases} \tag{5.18}$$

because,

$$P(h_i)+P(f_i)=1 \quad \forall i\in S \text{ where } m_i=1 \tag{5.19}$$

The key issue in the above introduced formula (5.15) is the determination of the conditional probability term ($P(m_j|h_i)$). This term enables the completion of probability model with additional knowledge about the problem. First, in case of shadow the indices in the summarizations can be reduced. In the above-defined form the summations are performed over the whole image. According to the geometrical model these 2D summations may be replaced with summations along a straight line (parameterised in 1D), direction of which is equal to $\vec{v}$. This procedure is demonstrated in Figure 5.5.

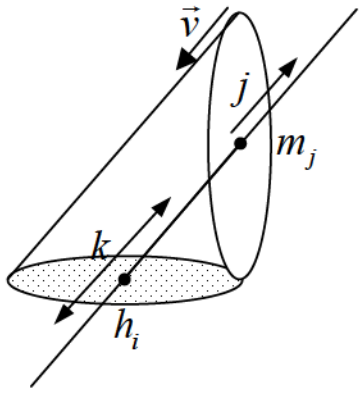


Figure 5.5: Simple geometrical constraint: including the collinearity to the conditional probabilities. The notations are introduced in the text. The indices *j* and *k* are related to the cyclical summarizations.

To allow this feature, a modified formula of (5.15) can be written:

DOI:10.15774/PPKE.ITK.2024.006

$$P_{k+1}(h_i) = P_k(h_i)\sum_j \frac{P\left(m_{r(j,i)}\middle|h_i\right)P\left(m_{r(j,i)}\right)}{\sum_k P\left(m_{r(j,i)}\middle|h_{r(k,i)}\right)P\left(h_{r(k,i)}\right)}, \quad i \in S \text{ and } j,k \in N \tag{5.20}$$

where the function $r(.)$ returns an image point, which is computed from an initial position (i) and a step counter (j) along the line:

$$r(j,i) = i + j\vec{v} \tag{5.21}$$

Based on this notation the expression of conditional probability may be rewritten using two step-counters along the line:

$$d_i(p,l) = P\left(m_{r(p,i)}\middle|h_{r(l,i)}\right),\ p,l \in N \tag{5.22}$$

For determination of this value, a simple formula is given:

$$d_i(p,l) = c_{r(p,i)}m_{r(p,i)}m_{r(l,i)}\left(1 - P\left(f_{r(l,i)}\right)\right) \tag{5.23}$$

This expression validates only the minimal conditions; motion must be present in both points and the shadow point must be in the colour mask (value of $c_{r(p,i)}$ is defined by (5.10)). The last component relates to the foreground mask, so, the probability that a given point belongs to shadow is equivalent to the probability that the point is not a foreground point. This extension makes connection between F and H during the iterations.

Since, the iteration formula contains the sum of these values (and because it is a 2D probabilistic distribution function), the normalization is necessary before use:

$$\sum_{p,l} d_i(p,l) = 1 \tag{5.24}$$

Till now, we used indices along the line without any upper and lower bounds. There is no need to compute the sums along the full line, because we can define the probable utmost distance between the original point and its corresponding shadow point. This distance is symbolized by parameter γ. Thus the ranges of the indices are

$$\begin{aligned} p &= -\gamma \ldots \gamma \\ l &= 0 \ldots \gamma \end{aligned} \tag{5.25}$$

DOI:10.15774/PPKE.ITK.2024.006

The values of $d_i(.)$ form a 2D pdf function. Its layout is visualized in matrix form in Figure 5.6.

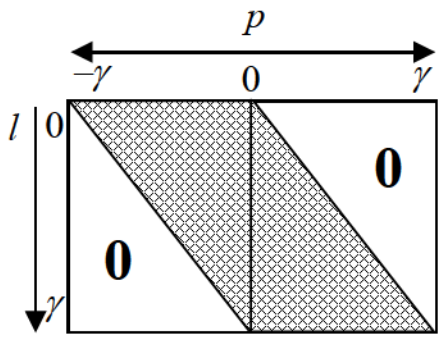


Figure 5.6: Layout of matrix $d_i(.)$. The filled region indicates the probable non-zero elements. This zone-structure is because the place of shadow is always relative to the original point.

After substituting (5.22) into (5.20) we get the final formula of the iteration step:

$$P_{k+1}(h_i) = P_k(h_i)\sum_j \frac{d_i(0,j)P\left(m_{r(j,i)}\right)}{\sum_k d_i(k-j,j)P\left(h_{r(k,i)}\right)}, \qquad (5.26)$$

$$i \in S \text{ and } j,k = 0...\gamma$$

The formulas for the foreground probabilities ($P(f_i)$) can be derived in the same way.

5.3.2. Experimental results

In the following, we present some samples of outdoor sequence. The first row contains the relevant part of input images. The second row displays the detected motion mask, see (5.9). The further rows demonstrate the foreground and shadow probabilities during the iteration steps.

DOI:10.15774/PPKE.ITK.2024.006

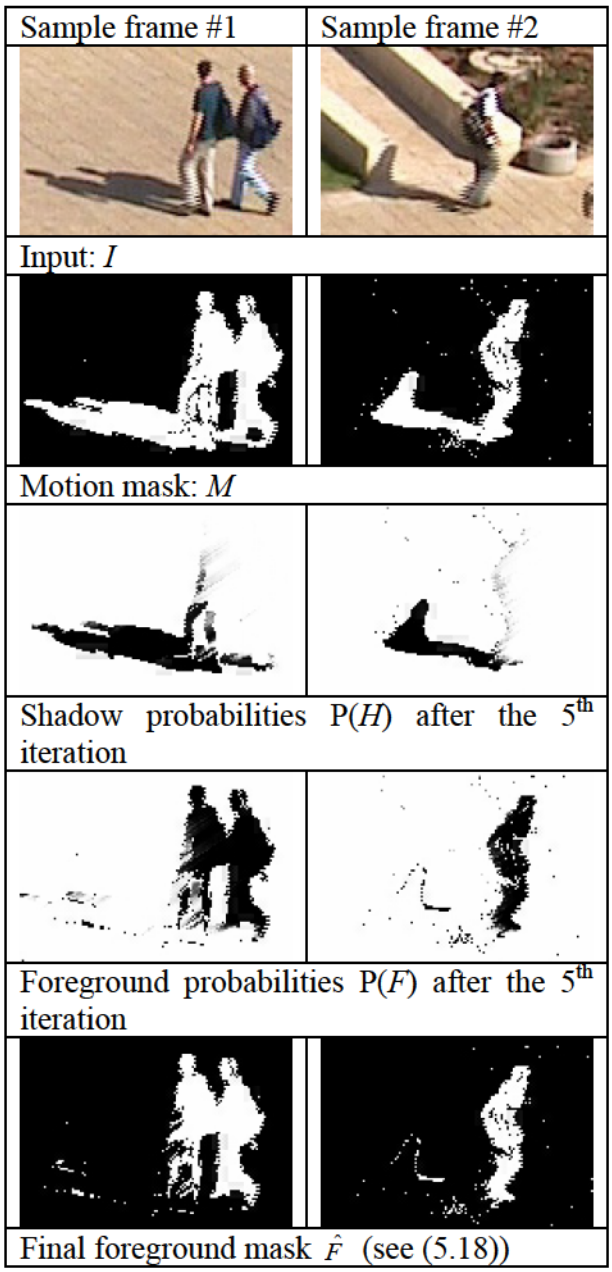


Figure 5.7: Experimental results on strong shadow. The final foreground mask is the output of the classifier, for details see text.

5.4. Conclusions

Based on the estimated geometrical model a simple method has been given for the improvement of the foreground segmentation step. This post-processing step is a possible way to remove reflections from the previously extracted foreground mask (determined by some arbitrary algorithm).

DOI:10.15774/PPKE.ITK.2024.006

We have presented an iterative Bayesian framework to determine the shadow and foreground masks taking into account both colour and geometrical information. The geometrical model of cast shadow is reduced to a simple direction (toward light vanishing point), which assists to implement an efficient localized variant of the iteration scheme.

DOI:10.15774/PPKE.ITK.2024.006

6.Summary

In Chapter 2 an overview about different geometrical scene characteristics is presented: point and line homography, skew symmetric fundamental matrix and horizontal vanishing line (horizon). The nature of parameter estimation tasks is also defined including the kind and amount of necessary information for computation. The implemented feature extraction methods were presented in Chapter 3. In detailed, the walk detection and gait feature identification were described in Section 3.1, while Section 3.2 summarizes the use of motion mask in a statistical framework for the detection of spatial correspondences. A discussion about the robustness is presented in Section 3.2.4. It was shown that the scene dynamic (moving objects) provides stable and useful features for further processing. Chapter 4 presents detailed descriptions about the parameter optimization approach, including outlier rejection and explanation of objective functions. Successful evaluations on both indoor and outdoor test videos prove the feasibility of the proposed methods.

DOI:10.15774/PPKE.ITK.2024.006

6.1. Methods used in the experiments

In the course of my work, theorems and assertions from the field of mathematical statistics, numerical geometry, optimization, reported results of image and video processing were explored.

The experiments for camera registration were performed by using the MDICam multi-camera software system that was designed in the Analogical and Neural Computing Laboratory. The second test environment is the PPKEyes which is a digital video surveillance system that was developed in the PPKE-ITK and it is operating in the university campus. For unique experiments I have also designed simulation systems in Matlab. Testing of the proposed algorithms was performed on various video sequences from personal experiments and from publicly available video databases. For the design and testing of algorithms I have used a software toolboxes provided by Intel.

6.2. New scientific results

The First Thesis summarizes the results that related to the use of detection of human activity (walking) for event detection and camera registration, the Second Thesis is about a statistical framework for geometrical scene analysis. The Third Thesis presents results that related to the foreground image mask segmentation problem.

6.2.1. First thesis

The high-level temporal descriptor of the structural changes of a moving non-rigid (human) object is feasible to detect the human activity (walking) and to determine information about the gait.

I give a new method to detect human activity and to determine a specific feature of walking in video sequence. I have introduced the eigenwalk space which is utilizable to detect the human walking. The proposed method is applicable to identify the leading leg, which is a descriptor of the gait.

DOI:10.15774/PPKE.ITK.2024.006

The detection of the human activity, namely the walking is possible by classifying the extracted temporal descriptors of object. The general criterion of walking is the moving legs, by detecting this motion the walking is perceptible and the leading leg can be identified from two successive steps.
This information about the scene is suitable not only for the event-level analysis of video sequences but for the registration of wide-baseline indoor camera configuration, which is a challenging task.

Published in [3][2][15][6][17][18][20]

1. **Detection of human walking using the spatio-temporal patterns of horizontal symmetries.**

   I worked out a method, which is able to compute the near horizontal symmetry axes. I have defined the symmetry levels, from which the third is characteristic for the presence of two pair near parallel edges (legs). From this fact, the input of the method is the binarized ridge of the edge map instead of the intensity map or object silhouette. I worked out a method for the temporal tracking and processing (time continuous interpolation and dimension reduction) of symmetry segments, which is the basis for the detection of pedestrians from walk patterns. The two classes (walk and non-walk) was separated with a non-linear hyper-plane. This classification step was carried out by using the Support Vector Machine (SVM) in the eigen space of walk patterns which is called eigenwalk space. I have managed to reduce the dimensionality of walk patterns through the linear dimension reduction technique (PCA).

2. **Identification of the leading leg.**

   I introduced a method for the identification of leading leg from one walk cycle (two successive steps). The non-rigid human body during a walking cycle has a useful property, which assists us in recognizing the leading leg. Depending on the 3D walk-direction, and on which is currently the leading leg, one leg or the other practically obscures the visible area between the legs. During a walk-

DOI:10.15774/PPKE.ITK.2024.006

cycle the ratio of the visible leg-opening areas, together with the 2D direction on the image-plane, can be used to identify which is the leading leg. I have shown experimentally that the method is reliable and accurate. I summarized the conditions necessary to the correct functioning.

I have listed the relationship between the leading leg and the ratio of surfaces from two successive patterns in table form. Furthermore the limitations of the method are also named. The leading leg as a discriminative feature is a novel description of the gait.

3. **Registration of partially overlapping and non-overlapping views by utilizing the detected walk patterns.**

I have shown experimentally that the registration of views can be done by using the spatial position of walk patterns. I have showed that the leading leg is a stronger discriminative feature and the spatial accuracy of walk detection is sufficient for the computation of homographies.

I used known optimization procedures and I have compared the model error of these methods. In case of non-overlapping views I utilized the line homography and corresponding line fragments instead of point-to-point correspondences.

6.2.2. Second thesis

The model-based, statistical description of the perceptible changes of scene and environment is applicable to determine the geometrical model of the plane-mirror, cast shadow and the horizon.

I have shown that the cumulative information which come from scene changing can be modeled with Gaussian mixture and can be used in geometrical model estimation problems. I worked out a framework for parametrical processing of motion statistics and for the use in different geometrical scene analysis computations.

DOI:10.15774/PPKE.ITK.2024.006

The detected changes in the camera plane reflect the changes of the dynamic scene and provide information about the position of camera and the geometrical properties of the scene. The vertical planar surface, or shadow casts on the ground-plane occur frequently in surveillance videos (both indoor and outdoor), and they inevitably cause problems in further image-processing steps and reduce the processing system's performance. A specific problem is the determination of horizontal vanishing line (horizon) which describes the relative camera orientation to the world coordinate system. These situations can be viewed as a geometrical optimization problem. To solve this optimization task I retrieved the set of measurements from the parametrical descriptors of motion statistics.

In summary, the investigated local co-motion statistics are feasible for the analysis of camera view in case of unknown environmental conditions.

Published in [1][9][11][5][8]

1. **The model based processing of motion statistics allows determination of spatial features to sub-pixel accuracy.**

   I have justified experimentally that, the parametrical descriptors of motion statistics can be used for the robust and accurate determination of 2D position information for parameter computation tasks.

   Briefly, the co-motion statistics are a numerical estimation of the concurrent motion probability (conditional probability) of different pixels in the camera plane. I investigated the theoretical background of the evaluation of such statistics and I have given the condition of the parametrical description. The analysis supports our empirical confidence in this statistical method.

   Both the empirical and the theoretical results confirm that the method is robust and is fairly insensitive to inaccuracy of the motion-mask. Based on our investigations, the length (frame count) of video sequence necessary for the robust extraction of correspondences may be estimated. The parameters in this formula are the estimated motion-intensity (which is a descriptor of the scene dynamics), and the detection error-rate of the motion-detector algorithm.

DOI:10.15774/PPKE.ITK.2024.006

2. **Using global optimum search method for the determination of geometrical model of camera-mirror scenes and cast shadow.**

   I proved that the geometrical model of planar mirror and cast shadow can be described with a skew-symmetric (auto epipole) fundamental matrix. This matrix determines a point-to-line transformation and it is formed from the position of vanishing point (2D mirror pole).
   I have defined an objective function from the geometrical features and statistical characteristics. The model parameters are the arguments of the objective function in its maximum, thus it leads to a global optimum search task. I have shown experimentally the robustness and accuracy of the proposed approach in both indoor and outdoor environmental conditions.
   I have justified experimentally that the sub-pixel accuracy can be achieved. The reduced spatial resolution of input data does not affect the precision of extracted features and the model parameters but enhances the running capabilities of the implementation.

3. **Determination of the horizontal vanishing line – which determines the orientation of camera – by using the height information of objects extracted from the motion statistics.**

   I have introduced a method which based on the statistical error propagation and the measurement transformation into the model parameter space. I showed that the computation of horizon can be originated in the same optimization problem such the previous section but it operates in the parameter space of lines namely in the Hough-space. It assumes that the data representation is continuous instead of the discrete accumulator array in the Hough-space, thus the formula of propagated error is expressed by Gaussian function. I have shown experimentally that the estimated height of objects can be used for horizon determination.

DOI:10.15774/PPKE.ITK.2024.006

6.2.3. Third thesis

Use of geometrical model for improved extraction of foreground image mask in case of reflection and cast shadow.

I worked out video segmentation methods with the integration of geometrical information into the decision process. I have shown experimentally that the resulted foreground image mask is more accurate than the mask without using the geometrical knowledge about the scene content.

During the processing of video sequences the basic feature extraction step is the perception of changes and foreground objects. Reflections and cast shadows in surveillance videos usually cause problems in image analysis. This is because they appear in the foreground mask extracted by using an adaptive background model. In turn, the inaccurate mask reduces the performance of the further image-processing steps. Consequently, techniques for the avoidance of such disturbances constitute an active current research area.
I introduced two different methods based on the estimated geometrical models; one for the removal of strong shadow pixels and an other for the removal of reflected pixels related to a valid foreground object, which can lead to better performance.

Published in [10][9][1]

1. **The removal of object's reflection from the foreground image mask based on Bayes decision rule.**

   I have introduced the integration of the geometrical model and statistics into a foreground-extraction method which is more reliable than previous approaches. Based on the model, only the fundamental constraint can be used, which is a point-line transformation. Because the co-motion statistics store information about the position of concurrent points, the inclusion of the appropriate component of the statistics into a class-conditional density

DOI:10.15774/PPKE.ITK.2024.006

function conveniently solves the identification of the reflection related to an arbitrary point.

I have shown experimentally – using both indoor and outdoor videos – that the detection error rate of foreground segmentation process can be reduced by taking into account the presence of reflective surface and by using the proposed post-processing method.

2. **The removal of moving cast (strong) shadow from foreground image mask.**

I give a novel approach for the identification of cast shadow related to foreground objects. The applied Bayesian iteration scheme is able to handle the *a prior* information about the object-shadow configuration. The proposed method is capable to remove the cast shadow regions from foreground image mask in case of strong shadow too.

In case of shadow the motion statistics can be used for the estimation of geometrical model only. This lack of spatial information is compensated by using the Bayesian iteration completed with the knowledge of the geometrical model. The proposed method uses the color based shadow segmentation in the initialization stage.

6.3. Examples for applications

All the developed algorithms and implementations offer solutions for real application problems.

The most important utilization of the methods is their integration into surveillance systems. These systems need algorithm with the capability of real-time functioning and robust operation.

The walk detection introduced in the first thesis is a useful procedure to scene analysis and event detection in both indoor and outdoor configurations. These tasks were implemented in the PPKEyes digital video surveillance system which is operating in the university campus.

DOI:10.15774/PPKE.ITK.2024.006

In the second thesis the presented geometrical model estimation provides the necessary information for improving the foreground image mask. It can be used to remove the pixels related to a reflective surface. This situation is often occurred in public places. This preprocessing step is important before using the foreground mask in some higher level processing (object detection, feature extraction etc.).

The approaches presented in the last thesis are the applications of the geometrical information determined in the previous thesis. Both classification methods are novel and do not use a prior assumptions.

DOI:10.15774/PPKE.ITK.2024.006

7.References

The author's journal publications

[1] **László Havasi** and T. Szirányi, "Estimation of Vanishing Point in Camera-Mirror Scenes Using Video", *Optics Letters,* vol. 31, No. 10, pp. 1411-1413, 2006.

[2] **László Havasi**, Z. Szlávik and T. Szirányi, "Higher order symmetry for non-linear classification of human walk detection", *Pattern Recognition Letters*, vol. 27, pp. 822-829, May 2006.

[3] **László Havasi**, Zoltán Szlávik and Tamás Szirányi: "Detection of gait characteristics for scene registration in video surveillance system", *IEEE Transactions on Image Processing*, 2006, in print

[4] Zoltán Szlávik, Tamás Szirányi and **László Havasi**: "Stochastic view registration of overlapping cameras based on arbitrary motion", *IEEE Transactions on Image Processing*, 2006, accepted

[5] Zoltán Szlávik, **László Havasi** and Tamás Szirányi: "Video camera registration using accumulated co-motion maps", *ISPRS Journal of Photogrammetry and Remote Sensing*, 2006, accepted

The author's conference publications

[6] **László Havasi**, Zoltán Szlávik, Tamás Szirányi: "Use of human motion biometrics for multiple-view registration", ACIVS 2005, ACIVS, LNCS, vol. 3708, pp. 35-44, 2005

[7] Zoltán Szlávik, **László Havasi**, Tamás Szirányi, „Estimation of common groundplane based on co-motion statistics", ICIAR'04, LNCS, vol. 3211, pp. 347-353, 2004

[8] **László Havasi** and Tamás Szirányi: „Extraction of horizontal vanishing line using shapes and statistical error propagation", PCV 2006, accepted

[9] **László Havasi** and Tamás Szirányi: „Use of motion statistics for vanishing point estimation in camera-mirror scenes", ICIP 2006, accepted

[10] **László Havasi**, Tamás Szirányi and Michael Rudzsky: „Adding geometrical terms to shadow detection process", EUSIPCO 2006, accepted

[11] Zoltán Szlávik, **László Havasi** and Tamás Szirányi: „Analysis of dynamic scenes by using co-motion statistics", IEEE International Workshop on Visual Surveillance, 2006

[12] Csaba Benedek, **László Havasi**, Tamás Szirányi, Zoltán Szlávik, "Motion-based Flexible Camera Registration", in Proc. of IEEE Advanced Video and Signal-Based Surveillance, AVSS'05, pp. 439-444, 2005

[13] **László Havasi**, Zoltán Szlávik, Csaba Benedek, Tamás Szirányi, "Learning human motion patterns from symmetries", ICML Workshop on Machine Learning for Multimedia, Bonn, 2005, (on CD-ROM)

[14] Zoltán Szlávik, **László Havasi**, Tamás Szirányi, Csaba Benedek, "Random motion for camera calibration", European Signal Processing Conference, EUSIPCO, Antalya, 2005

[15] **László Havasi**, Zoltán Szlávik, Tamás Szirányi: „Eigenwalks: walk detection and biometrics from symmetry patterns", ICIP, pp. 289-292, Genova, 2005

[16] Zoltán Szlávik, Tamás Szirányi, **László Havasi**, Csaba Benedek, "Optimizing of searching co-motion point-pairs for statistical camera calibration", IEEE International Conference on Image Processing, ICIP, pp. 1178-1181, Genova, 2005

[17] **László Havasi**, Zoltán Szlávik, Csaba Benedek, Tamás Szirányi, "Learning human motion patterns from symmetries", International Conference on Machine Learning Workshop on Machine Learning for Multimedia, Bonn, 2005, (on CD-ROM)

[18] **László Havasi**, Zoltán Szlávik, Tamás Szirányi: „Pedestrian detection using derived third-order symmetry of legs", ICCVG 2004, Warsaw, Poland, Kluwer, Computational imaging and vision

[19] **László Havasi**, Csaba Benedek, Zoltán Szlávik, Tamás Szirányi: „Extracting structural fragments of overlapping pedestrians", Proc. Of the 4th IASTED Int. Conference VIIP'04, pp. 943-948, Marbella, 2004

[20] **László Havasi**, Zoltán Szlávik, „Symmetry feature extraction and understanding", Proc. of CNNA'04, pp. 255-261, Budapest, 2004

[21] Zoltán Szlávik, **László Havasi**, Tamás Szirányi, „Image matching based on co-motion statistics", Proc. of 2nd Int. Sympozium on 3DPVT, Thessaloniki, 2004, (on CD-ROM)

[22] **László Havasi**, Tamás Szirányi: „Motion Tracking Through Grouped Feature Points", ACIVS, 2003

DOI:10.15774/PPKE.ITK.2024.006

Publications related to the dissertation

[23] H. Mitsumoto and S. Tamura, "3-D reconstruction using mirror images based on a plane symmetry recovering method", *IEEE Trans. on PAMI*, Vol. 14, pp. 941-946, 1992.

[24] R., Hartley and A., Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, Cambridge, 2003.

[25] R. Penne, "Mirror Symmetry in Perspective" in *Proc. of ACIVS, Lecture Notes on Computer Science*, 2005, pp. 634-642.

[26] R. J. Alexandre, G. G. Medioni and R. Waupotitsch, "Reconstructing mirror symmetric scenes from a single view using 2-view stereo geometry" in *Proc. of ICPR*, pp. 40012-40015, 2002.

[27] L. Lee, R. Romano and G. Stein, "Monitoring activities from multiple video streams: establishing a common coordinate frame," *IEEE Trans. on PAMI*, vol. 22, pp. 758-767, 2000.

[28] B. Hu, C. Brown and R. Nelson, "Multiple-view 3-D reconstruction using a mirror" Technical Report, University of Rochester Computer Science Department, 2005.

[29] R. Cucchiara, C. Grana, M. Piccardi and A. Prati, "Detecting Moving Objects, Ghosts and Shadows in Video Streams" *IEEE Trans. on PAMI*, vol. 25, pp. 1337-1342, 2003.

[30] C. Stauffer, W. Eric and L. Grimson, "Learning patterns of activity using real-time tracking", *IEEE Trans. on PAMI*, vol. 22, pp. 747-757, 2000.

[31] Cs. Benedek and T. Sziranyi, "Markovian Framework for Foreground-Background-Shadow Separation of Real World Video Scenes"' in *Proc. of ACCV, Lecture Notes on Computer Science*, 2006.

[32] G. Xu and Z. Zhang, *Epipolar Geometry in Stereo, Motion and Object Recognition*, Kluwer Academic Publisher, 1996.

[33] R. A. Johnson and D. W. Wichern, *Applied Multivariate Statistical Analysis*, Prentice Hall, 2002.

[34] F. Pernkopf, D. Bouchaffra, "Genetic-Based EM Algorithm for Learning Gaussian Mixture Models", *IEEE Trans. on PAMI*, vol. 27, pp. 1344-1348, 2005.

[35] W. Feller, "Laws of Large Numbers." , *An Introduction to Probability Theory and Its Applications*, Vol. 1, pp. 228-247, 1968.

[36] V. Nguyen, A. Martinelli, N. Tomatis and R. Siegwart, "A Comparison of Line Extraction Algorithms using 2D Laser Rangefinder for Indoor Mobile Robotics", in *Proc. of IEEE/RSJ Intenational Conference on Intelligent Robots and Systems*, 2005.

[37] H. Zabrodsky and D. Weinshall, "Utilizing symmetry in the reconstruction of 3-dimensional shape from noisy images", in *Proc. of ECCV*, pp. 403-410, 1994.

[38] J. C. Lagarias, J. A. Reeds, M. H. Wright and P. E. Wright, "Convergence properties of the Nelder-Mead simplex method in low dimensions", *SIAM Journal of Optimization*, Vol. 9, 1998.

[39] O. Kallenberg, *Foundations of Modern Probability*, New York: Springer-Verlag, 1997.

[40] Z. Szlavik, T. Sziranyi, "Bayesian Estimation of Common Areas in Multi-Camera Systems", in *Proc of ICIP,* 2006, accepted.

[41] J. Borenstein and Y. Koren, "The Vector Field Histogram–Fast Obstacle Avoidance for Mobile Robots," *IEEE Trans. on Robotics and Automation*, vol. 7, pp. 278-288, 1991.

[42] N. Kiryati and A. M. Bruckstein, "Heteroscedastic Hough Transform (HtHT): An Effective Method for Robust Line Fitting in the 'Errors in the Variables' Problem", *Computer Vision and Image Understanding*, vol. 78, pp. 69-83, 2000.

[43] A. Webb, *Statistical Pattern Analysis*, John Wiley & Sons, England, 2004.

[44] D.Hall, J. Nascimento, P. Ribeiro, E. Andrade, P. Moreno, S. Pesnel, T. List, R. Emonet,R.B. Fisher, J. Santos Victor and J.L. Crowley, "Comparison of target detection algorithms using adaptive background models", in Proc of PETS, pp. 113-120, 2005.

[45] Lu, F., Zhao, T. and Nevatia, R, 2000 Self-Calibration of a camera from video of a walking human. *in Proc. of ICPR*

[46] Criminisi, A., Reid, I. and Zisserman, A., 1999 Single view metrology. *in Proc. of ICCV*, pp. 434-442

[47] Ji, Q. and Xie, Y., 2003 Randomised hough transform with error propagation for line and circle detection. *Pattern Analysis and Applications*, vol. 6, pp. 55-64.

[48] Haralick, R.M., 1994 Propagating covariance in computer vision. *in Proc. of ICPR*, pp. 493-498.

[49] Kiryati, N. and Bruckstein, A. M., 2000 Heteroscedastic Hough Transform (HtHT): An Effective Method for Robust Line Fitting in the 'Errors in the Variables' Problem. *Computer Vision and Image Understanding*, vol. 78, pp. 69-83.

[50] Duda, R. O. and Hart, P. E., 1972 Use of the Hough transform to detect lines and curves in pictures. *Comm. ACM*, pp. 11-15.

[51] R. Cutler and T. Ellis, "Robust real-time periodic motion detection, analysis and applications" *IEEE Trans. PAMI*, vol. 22, pp. 781-796, August 2000.

[52] H. Murase and R. Sakai, "Moving object recognition in eigenspace representation: gait analysis and lip reading" *Pattern Recognition Letters*, vol. 17, pp. 155-162, February 1996.

[53] L. Wang, T. Tan, H. Ning and W. Hu, "Silhouette Analysis-Based Gait Recognition for Human Identification" *IEEE Trans. PAMI*, vol. 25, pp. 1505-1518, December 2003.

[54] J. Hayfron-Acquah, M. Nixon and J. Carter, "Automatic gait recognition by symmetry analysis" *Pattern Recognition Letters*, vol. 24, pp. 2175-2183, September 2003.

[55] M. Soriano, A. Araullo and C. Saloma, "Curve spreads: a biometric from front-view gait video" *Pattern Recognition Letters*, vol. 25, pp. 1595-1602, October 2004.

[56] C. Curio, J. Edelbrunner, T. Kalinke, C. Tzomakas and W. von Seelen, "Walking pedestrian recognition" *IEEE Trans. Int. Transport Systems*, pp. 155-163, September 2000.

[57] C. BenAbdelkader, R. Cutler, H. Nanda and L. Davis, "Eigengait: Motion-based Recognition of People Using Image Self-Similarity", in *Proc. Int. Conf. Audio- and Video-Based Biometric Person Authentication*, 2001, pp. 284-294.

[58] S. Chaudhuri and D.R. Taur, "High-resolution slow-motion sequencing: how to generate a slow-motion sequence from a bit stream", *IEEE Signal Processing Magazine*, vol. 22, pp. 16-24, March 2005.

[59] L. Lee, R. Romano and G. Stein, "Monitoring activities from multiple video streams: establishing a common coordinate frame," *IEEE Trans. PAMI*, vol. 22, pp. 758-767, August 2000.

[60] J. Kang, I. Cohen and G. Medioni, "Persistent objects tracking across multiple non overlapping cameras", in *Proc. of WACV/MOTION*, 2005, vol. 2, pp. 112-119.

[61] A. Rahimi, B. Dunagan, and T. Darrell, "Tracking People with a Sparse Network of Bearing Sensors", in *Proc. of ECCV, Lecture Notes in Computer Science*, 2004, pp. 507-518.

[62] Y. Caspi and M. Irani, "Alignment of Non-Overlapping Sequences", *International Journal of Computer Vision*, vol. 48, pp. 39-51, 2002.

[63] R. Cucchiara, C. Grana, M. Piccardi and A. Prati, "Statistical and knowledge-based moving object detection in traffic scene',' in *Proc. of IEEE Int'l Conf. On Intelligent Transportation Systems*, 2000, pp. 27–32.

[64] B. Jahne, *Digital image processing*, Springer, Berlin, 1991.

[65] C. de Boor, *A Practical Guide to Splines*, Springer Verlag, New York , 1978.

[66] P. Huang, C. Harris and M. Nixon, "Human Gait Recognition in Canonical Space Using Temporal Templates" *IEE Proc. Vision Image and Signal Processing Conf.*, 1999, pp. 93-100.

[67] K.-L. Müller, S. Mika, G. Ratsch, K. Tsuda and B. Schölkopf, "An Introduction to Kernel-Based Learning Algorithms", *IEEE Trans. Neural Network*, vol. 12, pp. 181-201, 2001.

[68] Z. Zalevsky, E. Rivlin, and M. Rudzsky, "Motion characterization from co-occurrence vector descriptor," *Pattern Recognition Letters*, vol. 26, pp. 533-543, 2005

[69] A. Prati, I. Mikic, M. M. Trivedi, and R. Cucchiara, "Detecting Moving Shadows: Algorithms and Evaluation," *IEEE Trans. PAMI, vol. 25, pp*. 918-923, 2003.

[70] A. Yoneyama, C. H. Yeh, and C. C. Jay Kuo, "Moving Cast Shadow Elimination for Robust Vehicle Extraction based on 2D Joint Vehicle/Shadow Models," in Proc. of IEEE Int'l Conf. on Advanced Video and Signal Based Surveillance, 2003.

[71] S. Nadimi, and B. Bhanu, "Physical Models for Moving Shadow and Object Detection in Video," *IEEE Trans. PAMI*, vol. 26, pp. 1079-1087, 2004.

[72] S. A. Shafer, "Shadows and Silhouettes in Computer Vision," *Kluwer Academic Publisher*, 1985.

[73] A. Webb, "Statistical Pattern Recognition," John Wiley & Sons, 2004.

[74] L. Kovács, and T. Szirányi, "Relative Focus Map Estimation Using Blind Deconvolution," *Optics Letters*, vol. 30, pp. 3021-3023, 2005.

[75] W. H. Richardson, "Bayesian-Based Iterative Method of Image Restoration," *JOSA* vol. 62, pp. 55-59, 1972.

[76] Abdelkader, C., Cutler, R., and Davis, L., 2002, Motion-based recognition of people in eigen-gait space, *Proc. of the 5th Int. Conf. on Automatic Face and Gesture Recognition*

[77] Baumberg, A., and Hogg, D., 1994, Learning Flexible Models from Image Sequences, *Proc. European Conf. on Computer Vision*, 299-308.

[78] Borgefors, G., Ramella, G., Sanniti di Baja, G., 2001, Hierarchical Decomposition of Multiscale Skeletons, *IEEE Trans. PAMI*, 23(11), 1296-1312.

[79] Canny, J., 1986, A computational approach to edge detection, *IEEE Trans. PAMI*, 8(6), 679-698.

[80] Giblin, P., and Kimia, B. B., 2004, A Formal Classification of 3D Medial Axis Points and Their Local Geometry, *IEEE Trans. PAMI*, 26(2), 238-251.

[81] Haritaoglu, I., Harwood, D., and Davis, L. S., 2000, W4: Real-Time Surveillance of People and Their Activities, *IEEE Trans. PAMI*, 22(8), 809-830.

[82] Hayfron, A. J., Nixon, M. S. and Carter, J. N., 2002, Human identification by spatio-temporal symmetry, *International Conference on Pattern Recognition*, pp. 632-635

[83] Kurita, T., Taguchi, T., 2002: A Modification of Kernel-Based Fisher Discriminant Analysis for Face Detection, *Int. Conference on Automatic Face and Gesture Recognition*, pp. 285-290

[84] Mika, S., Rätsch, G., Weston, J., Schölkopf, B., and Müller, K.-R., 1999, Fisher Discriminant Analysis With Kernels, *Neural Networks for Signal Processing IX*, pp. 41-48

[85] Moeslund, T., Granum, E., 2001 A Survey of Computer Vision-Based Human Motion Capture, *Computer Vision and Image Understanding*, 81, 231-268.

[86] Mohan, A., Papageorgiou, C., and Poggio, T., 2001, Example-based object detection in images by components, *IEEE Trans. PAMI*, 23(4), pp. 349-361

[87] Mokhtarian, F., and Mackworht, A. K., A, 1992, Theory of Multi-Scale, Curvature-Based Shape Representation for Planar Curves, *IEEE Trans. PAMI*, 14(8), 789-805.

[88] Nguyen, H. T., Worring, M., and Dev., A., 2000, Detection of moving objects in video using a robust motion similarity measure, *IEEE Trans. on Image Processing*, 9(1)

[89] Osuna E., Freund R. and Girosi F., 1997, Improved training algorithm for support vector machines. *NNSP'97*

[90] Reisfeld, D., Wolfson, H., and Yeshurun, Y., 1995, Context-free attentional operators - The Generalized Symmetry Transform, *Int. Journal of Computer Vision*, 17, 119-130.

[91] Sharvit, D., Chan J., Tek H. and Kimia B.B., 1988, Symmetry-based indexing of image databases, *J. Visual Comm. And Image Representation*, vol. 9 no. 4, pp. 366-380

[92] Song, Y., Goncalves, L., and Perona, P., 2003, Unsupervised learning of human motion, *IEEE Trans. PAMI*, Vol. 25, pp. 814-828

[93] Tax, D.M.J., Duin R.P.W., 2001, Uniform Object Generation for Optimizing One-class Classifier, *Journal of Machine Learning Research*, pp. 155-173

[94] Zhu, S. C., and Yuille, A. L., 1996, Forms: A Flexible Object Recognition and Modeling System, *Int. Journal of Computer Vision*, 20(3)

[95] Wren, C., Azarbayejani, A., Darrel, T., and Penltand, A., 1997, Pfinder: Real time Tracking of the Human Body, *IEEE Trans. PAMI*, 19(7), 780-885.

DOI:10.15774/PPKE.ITK.2024.006