

Kisméretű Pilóta Nélküli Repülők Képi Alapú Ütközés Elkerülése



A disszertáció tézisei

Zsedrovits Tamás

Témavezető:
Zarándy Ákos, D.Sc.

Konzulens:
† Roska Tamás, D.Sc.

Pázmány Péter Katolikus Egyetem
Információs Technológiai és Bionikai Kar
Multidiszciplináris Műszaki és Természettudományi Doktori Iskola

Budapest, 2014

1 Bevezetés

Az elmúlt évtizedben a pilóta nélküli repülőgépek (UAV) és az ezeket támogató technológia nagymértékben fejlődött. A katonai alkalmazások mellett most már adott a lehetőség, hogy civil feladatokat is ellássunk ezek segítségével. Egyre több cég kapcsolódik be a UAV alapú alkalmazások fejlesztésébe. Több, a légiközlekedés területén dolgozó szakértő véleménye szerint, a közeljövőben a pilóta nélküli repülőgépek forradalmasítják a légi közlekedést. Ahogy az IEEE Spectrum 2011 decemberi számának vezető cikkében olvasható: „El fog jönni a pilóta nélküli utasszállító repülő kora, a kérdés csak az, hogy pontosan mikor.” – nyilatkozta James Albaugh a Boeing Commercial Airlines elnök és vezérigazgatója.

Ahhoz, hogy a pilóta nélküli repülőgépeket használni tudjuk a civil alkalmazásokban, növelnünk kell a megbízhatóságukat, tovább kell bővítenünk a képességeik körét, egyszerűbbé kell tennünk a használatukat, valamint csökkentenünk kell az árukat. Ezzel egy időben meg kell alkotni a UAV-k nemzeti és nemzetközi légtérbe való integrálásának szabályozását is. Az egyik legfontosabb lépés az ütközés elkerülés biztosítása. Az ütközés elkerülésnek a repülőgép fedélzetén kell futnia, hogy a repülő, egy esetleges kommunikációs, vagy szenzor probléma esetén is képes legyen a biztonságos repülésre.

Abban az esetben, ha a repülőgép fedélzetén a hely, és az elérhető elektromos energia korlátozott, a kamera alapú ütközés elkerülés költséghatékonyabb, mint például a nagyobb repülőkön

használatos TCAS rendszer, és még a tömege is kisebb. A kamera alapú megoldás a repülőterek környezetében szintén előnyös, mivel ott egyszerűen sok repülő van jelen, és a felhasználható frekvenciatartomány is limitált.

A mai sokprocesszoros csipek segítségével kis fogyasztással komplex algoritmusokat valósíthatunk meg, ami lehetővé teszi a képfeldolgozást. Az itt bemutatott munkámat az Office of Naval Research (ONR) és ONR Global szervezetek N62909-10-1-7081 számú kutatási projekt keretében végeztem. A kutatás fő célja, egy olyan közepes méretű, fix szárnyú, autonóm, pilóta nélküli repülőgép kifejlesztése, amely biztonságosan használható civil alkalmazásokban.

A kutatási projektben párhuzamosan történt a használni kívánt hardver elemek fejlesztése a szoftveres keretrendszerrel, valamint a látás alapú ütközés elkerülés (SAA) megalkotásával. A kutatás idején nem állt rendelkezésre teljes látás alapú SAA rendszer közepes méretű UAV-k részére, valamint nem voltak ismertek a vizuális SAA rendszerek tulajdonságai.

A célom az volt, hogy új kamera alapú algoritmusokat készítssek az UAV SAA-hoz, valamint hogy meglévő algoritmusokat vizsgáljak, amik használhatók a probléma megoldására. Pontosabban, arra a kérdésre keresem a választ, hogy milyen képi információ nyerhető ki a képfolyamból abban az esetben, ha az észlelt repülő már elég nagy a képsíkon, valamint hogy a saját repülőnk állásszög meghatározásához milyen rekonstrukciós algoritmusok használhatók.

2 A vizsgálat módszerei

A munkám során a képfeldolgozás, valamint a többszörös nézetű geometria módszereit használtam. Új algoritmust mutattam be az érzékelt repülőgép relatív állásszögének becslésére, melyben az adott geometriai elrendezést és a projektív kamera modellt használja. Az eljárások vizsgálatánál végzett kísérletekben a multi-foveális képfeldolgozást használtam celluláris operátorokkal a szegmentációhoz.

A kísérleteket egyrészt az általunk készített hardver-in-the-loop(HIL) szimulátoron futtattam, másrészt a készülő UAV-t használtam a repülési tesztekhez. A HIL szimulátorban a repülésirányítás a repülőn is használt mikrokontrolleren fut, míg a repülők fizikai modelljét MATLAB Simulinkben futtattuk. A UAV szimulálására egy nagy pontosságú fizikai modellt használtunk, míg a másik repülő modellje egy egyszerű dupla integrátor. A saját géphez tervezett pályakövető rendszer egy MPC555-ös mikroprocesszoron fut. A repülési adatokat számító PC vezetékes hálózaton küldi és fogadja az adatokat a képeket előállító és feldolgozó számítógéppel. A rendereléshez a FlightGear szimulátor programot használtuk. A képfeldolgozás egy módosított FlightGear-ben fut. A képfeldolgozás végeredménye, ami az érzékelt repülőgép képsíkon mért szögeit és méretét jelenti, USB csatolón keresztül jut el a mozgásbecslést végző FPGA-hoz. Az FPGA egy EKF-et futtat a másik repülő relatív 3D-s pozíciójának becslésére. A jelenlegi rendszerben a becslés egy Spartan 3 FPGA-n fut. A becslés eredménye az irányítási rendszerhez kerül vissza, ahol a kockázatbecslés és a pályagenerálás is történik.

A relatív állásszög becslő algoritmust a HIL szimulátorral készített képeken, valamint egy modell repülőről felvett videókon teszteltem. A repülőgépek szárnyvégpontjait először egy szakértő, majd egy algoritmus jelölte ki a tesztekhez. A hiba meghatározásánál a számított és a beállított szöget kivontam egymásból, hogy megmaradjon az előjel információ.

A kamera forgatás számításánál négy kiválasztott algoritmust vizsgáltam: (i) Egy homográfia alapú megoldást, ami kis számítási igényvel rendelkezik, de kevésbé pontos. (ii) A nyolc pont algoritmust, ami az epipoláris geometriát használja, és egy alap algoritmus. (iii) Az öt pont algoritmust, amely az egyik legjobb, várhatóan a legstabilabb a különböző helyzetekben, de nagy számítási igényű. Végül, (iv) az MLESAC, egy iteratív, sztochasztikus megoldás.

A vizsgálatokhoz a MATLAB EGT eszközkészlettel generáltam jellegpontokat. Két valós, repült pályát választottam: (i) egy szinuszos pályát és (ii) egy cikk-cakk pályát. Mindkettő esetben 70 másodperc volt a repülési idő, és 100 Hz volt az alap mintavételi frekvencia, ami 7000 mintavételi pontot jelent. A jellegpontokat véletlenszerűen helyeztem el a térben a földhöz közel. A jellegpont koordinátákat a repülőn használt kamera kalibrációs mátrixán alapuló kamera modellekkel képeztem le 2D-ba. A különböző térbeli felbontásokhoz az eredeti kalibrációs mátrix átskálázott változatát használtam. Az eljárások teljesítményét a számított három Euler szög abszolút hibáját vizsgálva határoztam meg. Emellett a hiba átlagát, középértékét és szórását is vizsgáltam.

3 Új tudományos eredmények

1 Tézis: *Relatív állásszög becslő algoritmus autonóm pilóta nélküli repülőgépek látás alapú ütközés elkerülő rendszeréhez:*

Új eljárást mutattam be a relatív állásszög becsléshez, és meghatároztam a különböző helyzetekben elérhető pontosságot. Tegyük fel, hogy a két egymás felé haladó repülőgép egyenes pályát követ, és hogy a kamera kalibráció ismert. Egy egyszerű eljárást adtam a repülőgép szárnyvégpontjainak meghatározására a képsíkon. Az állásszög becslő algoritmus pontosságát mértem tisztán szimulációs esetben, renderelt képeken és videó felvételeken is. A pontosságot vizsgáltam szakértő által bejelölt szárnyvégpont koordinátákon, és az egyszerű algoritmus által meghatározott koordinátákon is.

1.1 Egy új eljárást mutattam relatív állásszög becslésre autonóm UAV-k vizuális alapú SAA rendszeréhez. Az algoritmus feltételezi, hogy mindkét repülő egyenesen halad. Megmutattam, hogy szimuláció esetén, amikor a koordináta értékeket nem terheli zaj, és nincsenek kerekítve, az eljárás pontossága a numerikus pontossággal összemérhető.

A relatív állásszög a következőképpen számítható:

$$\cos \alpha = \frac{\langle p_{p_3} - p_4; P_1 - P_2 \rangle}{\|p_{p_3} - p_4\| \|P_1 - P_2\|} \quad (1)$$

ahol p_{p_3} és p_4 a képsíkon mért koordináták, P_1 és P_2 a kamera mátrix alapján becsült mennyiségek, és igaz a két repülő pályájára tett feltevés. Ebben a modellben a 180° -kal elforgatott értékek egyenlők, és az $\alpha = \cos^{-1} X$ függvény az $\alpha = [0^\circ; 180^\circ]$ intervallumon ad eredményt. A relatív szög, α a $[-90^\circ; 90^\circ]$ tartományban értelmezett, így a következő szabály szerint kell számítani a végeredményt. Ha $\alpha > 90^\circ$, akkor $\alpha = 180^\circ - \alpha$, ha $\alpha < -90^\circ$, akkor $\alpha = -180^\circ - \alpha$. Így minden esetben számíthatók a helyes értékek.

Ha a másik repülő az xy horizontális síkon helyezkedik el, p_{p_3} egyenlő p_4 és α nem számítható ezzel a módszerrel. Ebben az esetben a UAV repülési magasságát változtatni kell.

A közeledő repülő centroid pontját a képsíkról különböző távolságokra vetítjük vissza a lyukkamera modell szerint. A repülő szárnyfesztszáma 11m, ami megegyezik egy Cessna 172p repülővel. A Cessna 172p egy elterjedt kisrepülő, amivel a UAV egy légtérben fog repülni. A szárnyat egy 11m-es szakasz modellezi, amit az előzőleg számított pontban forgattam el. A látószög, a kamera felbontás, valamint az x tengely mentén vett távolság szükséges a számításhoz. A repülőgép törzsét elhanyagoltam. A számítások segítségével közelítettem a hiba alsó korlátját.

1.2 Szimulációk segítségével vizsgáltam a pontosság változását a közeledő repülőgép pozíciójának függvényében. Kísérletekkel megmutattam, hogy minél közelebb van a repülő a vízszintes, (y) tengelyhez, annál nagyobb az α szög hibája. És ehhez hasonlóan, minél nagyobb az x tengely mentén vett távolság, azaz minél kisebb a repülő a képen, a térbeli diszkretizáció annál nagyobb hibát generál.

A méréseket ismét lyukkamera modellel végeztem, és az első esetben használt repülő modellekkel. A különbség az volt, hogy most a koordináta értékek kerekítettek.

Az x tengely mentén vett távolság 1 km, a felbontás 1920x1080 pixel, a látószög vízszintes irányban 50° és a képpontok négyzet alakúak. A repülő szárnyfeszítávolsága 11m, és a képsíkon 15 és 20 pixel közötti méretűnek látszik, a pozíciótól és a forgatástól függően.

Megmutattam, hogy az azimut szög változása gyakorlatilag nincs hatással a pontosságra, de a relatív távolság és a relatív eleváció változás hatással van a pontosságra. A változás egyik oka, hogy a nagyobb relatív távolság esetén a repülő kisebbnek látszik, a másik ok pedig, hogy nagyobb magasságbeli különbség esetén jobban rálátunk a szárnyra.

1.3 Vizsgáltam az elérhető pontosságot renderelt képeken és valós videókon. Kísérletileg megmutattam, hogy mind az emberi szakértő által, mind az általam bemutatott egyszerű algoritmussal megközelíthető az elméleti pontosság. Azt is megmutattam, hogy valós a videókon fellépő zaj időbeli átlagolással kiszűrhető.

A képeket a szimulációs környezetben készítettem el. Ezeken egy szakértő jelölte be a szárnyvégpontokat. A közeledő repülő pozícióját az előzőekhez hasonlóan visszavetítéssel számoltam, és az xy síkban vett szögekkel elforgattam. A felbontás 1920×1080 képpont, a vízszintes látószög 50° és a pixelek négyzet alakúak. A mérések megmutatták, hogy jól megválasztott szárnyvégpont koordináták esetén a hiba az elméleti minimumhoz közelít.

A szárnyvégpontokat egy egyszerű algoritmussal is számoltam, ami a szegmentált képet veszi alapul. Az y és z koordináták szélső értékeit megfelelő sorrendben számolva kaptam a koordinátákat. Ebben az esetben, amikor a repülő 80° és -80° szögekkel volt elforgatva, nagyobb hibát kaptam, mivel az egyszerű algoritmus nem tud különbséget tenni a szárnyvégpontok és a farok végpont között. Ezzel szemben a közepes tartományban az igen egyszerű algoritmus hatékonysága megközelíti a szakértőét (közel van az elméleti határhoz).

2 Tézis: *Négy, a pilóta nélküli repülőök ütközés elkerülésénél a forgatás számítására használt, kamera állásszög becslő algoritmus hibaanalízise:*

Négy relatív állásszög becslő algoritmust választottam ki: (i) Egy homográfia alapú megoldást, ami kis számítási igényvel rendelkezik, de kevésbé pontos. (ii) A nyolc pont algoritmust, ami az epipoláris geometriát használja, és egy alap algoritmus. (iii) Az öt pont algoritmust, amely az egyik legjobb, várhatóan a legstabilabb a különböző helyzetekben, de nagy számítási igényű. Végül, (iv) az MLESAC, egy iteratív, sztochasztikus megoldás. A cél ezen algoritmusok előnyeinek és hátrányainak bemutatása a vizuális alapon működő repülőgép állásszög becslési feladatban.

2.1 Szimulációk segítségével megvizsgáltam a négy kiválasztott algoritmus teljesítményét. A szimulációkban kétféle valós repült pályát használtam, és szintetikus képeket készítettem véletlenszerűen elhelyezett jellegpontokról, a repülőgépen található kamera alapján számított kamera modellel, többféle felbontással. Kísérletileg megmutattam, hogy pontos jellegpont koordinátákkal számolva az öt pont algoritmus a legjobb, és a hiba a numerikus hibával összemérhető. A másik két, epipoláris geometriát használó algoritmus hibája kevesebb, mint 0,1 pixel, és a homográfia hibája is 1 pixel alatt van.

A tesztekhez 350 jellegpontot helyeztem el véletlenszerűen, egyenletes eloszlással egy 2km széles, 3km hosszú, és 30 m magas hasámban. A koodrináták -1 és 1km között vannak az Y irányban, 0 és 3km között az X irányban, és a legnagyobb magasság 23m, a legkisebb pedig 3m-rel a föld alatt van, a kisebb lyukak szimulálására.

A kamera leképezéshez az egyik fedélzeten használt mini kamera kalibrációs mátrixát használtam. A belső kamera mátrixot különböző felbontású kameráknak megfelelően skáláztam.

Az első tesztek abszolút pontos jellegpont koordinátákkal futtattam. Ebben az esetben a lehető legjobb megoldást kaptam, mivel gyakorlatilag nincsen térbeli diszkretizáció, így az időbeli felbontás egyedül vizsgálható

2.2 Vizsgáltam a transláció hatását a kiválasztott eljárások teljesítményére. Kísérletekkel megmutattam, hogy a hiba nagyobb időlépés esetén nagyobb, az öt pont algoritmus néhány esetét kivéve.

A pitch szög eredményeit mutattam meg, amelyikre leginkább hat a translációból eredő hiba. Elméletileg, a nagyobb alapvonal távolság miatt, a két kép között eltelt több idő előnyös az epipoláris kényszerrel teljesítő algoritmusok esetén (öt pont, nyolc pont, MLESAC). A mérésekből az derült ki, hogy ez a gyakorlatban nincs így. A hiba nagyobb, a nagyobb időlépés esetén, kivéve az öt pont algoritmus néhány esetét. Egy lehetséges magyarázat, hogy a mindkét egymás utáni képen jelen levő jellegpontok száma lecsökken. Emellett viszont az egész repült pályára számított integrális hiba kisebb.

2.3 A szubpixeles felbontású jellegpont meghatározó algoritmusok esetleges használatának lehetőségét vizsgáltam. Kísérletekkel megmutattam, hogy az öt pont algoritmus kivételével, az állásszög becslés ki tudja használni a szubpixeles jellegpont meghatározás által adott plusz információt.

A szubpixeles felbontást az abszolút pontos koordinátákhoz adott, normál eloszlású (0 átlag, 0,5 pixel szórású) zaj hozzáadásával szimuláltam.

Meglepő módon az öt pont algoritmus rosszul teljesített ezekben a tesztekben a többiekhez viszonyítva. A nyolc pont algoritmus, és az MLESAC átlagos hibája kisebb volt ebben az esetben.

	5 pont	8 pont	Homo- gráfia	MLESAC
Absolute precision	$3.2 \cdot 10^{-11}$	$2.1 \cdot 10^{-3}$	$5.1 \cdot 10^{-2}$	$1.3 \cdot 10^{-3}$
Subpixel	$1.2 \cdot 10^{-1}$	$1.1 \cdot 10^{-2}$	$7.2 \cdot 10^{-2}$	$2.0 \cdot 10^{-3}$
Pixelized	$9.4 \cdot 10^{-2}$	$5.5 \cdot 10^{-1}$	$1.2 \cdot 10^{-1}$	$3.2 \cdot 10^{-1}$

Table 1 A négy algoritmus roll hibájának változása a különböző jellegpont koordináta pontosságokra, CPAR=0.093°/px kamera esetén

2.4 Vizsgáltam az eljárások teljesítményét általánosabb esetben is, amikor a jellegpont koordináták kerekítettek, vagy ezen felül zajjal terheltek. Kísérletekkel megmutattam, hogy az öt pont algoritmus teljesít a legjobban 1 pixel körüli átlagos hiba értékkel. Kísérletekkel megmutattam, hogy a homográfia algoritmus csaknem olyan jól teljesített, mint az öt pont, 1,5 pixel körüli átlagos hibával. A homográfia számítási igénye 2 nagyságrenddel kisebb, mint az öt pont algoritmusé, a szorzások számát tekintve. Kísérletekkel megmutattam, hogy a kerekítés sokkal kevésbé van hatással a homográfiára, mint a többi algoritmusra. Állítható, hogy a homográfia használható azon esetekben, amikor a számítási kapacitás korlátozott, mivel minimális a számítási igénye.

4 Az eredmények felhasználása

Amint a bevezetőben olvasható, a kutatási projekt célja egy autonóm UAV fejlesztése, amely képes az ütközés elkerülésre. A készülő pilóta nélküli repülőgépes rendszer az eredmények fő felhasználási területe.

Az EKF alapú mozgás becslés profitálhat a relatív állásszög becslő algoritmus eredményeiből. Amikor a szárnyvégpontok meghatározhatók, a szög mérési adatként használható az EKF-ben. Egy másik lehetőség a kiszámított α szög használata az EKF inicializálásánál.

A hiba analízis eredményét a saját mozgás becslésénél használjuk, ami képes a konvencionális INS/GPS adatok, a képi adatok alapján elvégzett számítások eredményeivel való fuzionálására. Első lépésként az öt pont és a homográfia becsléseit használtuk a fuzionált becslés tesztelésére. A tesztek megerősítették, hogy a homográfia valóban kevésbé érzékeny a zajra.

Ezen túlmenően a felépített környezet és az eredmények segíthetnek egy adott szituációhoz megfelelő algoritmus kiválasztásában. Ha a kívánt pontosság és például a számítási kapacitás adott, meghatározható, hogy milyen kamerát, és melyik algoritmust használjuk. A kamerát a CPAR határozza meg, az algoritmust pedig a pontosság és a tér-időbeli korlátozások. Továbbá más állásszög becslő algoritmusok is tesztelhetők a szimulációs környezetben.

5 Köszönetnyilvánítás

Elsősorban témavezetőmnek, Zarándy Ákosnak szeretném megköszönni a támogatását. Sokat segített nekem, hogy megtaláljam az utam a PhD tanulmányaim elején, tanácsaival segítette a munkámat, és motivációt nyújtott lelkesedésével. Roska Tamásnak is köszönöm a nagyvonalú támogatását, és az atyai jó tanácsait a tanulmányaim során.

Hálás vagyok Szolgay Péternek, Nyékyné Gaizler Juditnak, Kovács Ferencnek és Bércesné Novák Ágnesnek a támogatásukért, és bátorításukért az évek folyamán.

Ugyancsak szeretném megköszönni legközelebbi munkatársaim segítségét, és mind a remek beszélgetéseket: Vanek Bálintnak, Nagy Zoltánnak, Kiss Andrásnak, Bauer Péternek, Péni Tamásnak, Pencz Borbálának, Németh Máténak és Leitli Arnoldnak.

Igen hálás vagyok doktorandusz társaim segítségével, különösképpen, akikkel egy irodában dolgoztunk. Ők: Koller Miklós, Tornai Gábor, Radványi Mihály, Fülöp Tamás, Rák Ádám, Horváth András, Stubendek Attila, Gergelyi Domonkos, Józsa Csaba, Füredi László, Tornai Kálmán, Borbély Bence, Reguly István, Nemes Csaba, Laki László, Balogh Ádám, Sárkány Norbert, Tuza Zoltán, Rudan János, Varga Balázs, Tisza Dávid, Pilissy Tamás, Kárász Zoltán, Kozák László, Kovács Andrea, Szolgay Dániel, Fekete Ádám, Gelencsér András, Bihary Dóra, Gelencsér Zsolt, Hiba Antal, László Endre, Siklósi Borbála, Rác András, Oláh Balázs, Horváth Anna, Laki András, Juhász Imre és Jani Mátyás.

Szintén hálával tartozom a posztdoktori kollégáknak, Oláh Andrásnak, Cserey Györgynek, Karacs Kristófnak, Iván Kristófnak, Gáspári Zoltánnak és Gyöngy Miklósnak az ihletett munkájukért, és minden bátorító szóért.

Köszönöm a Tanulmányi Osztály, személy szerint Vida Tivadarné Katinka, a Dékáni Hivatal, a Gazdasági Osztály és az Informatika készséges segítségét. Sifter Viktória könyvtárosnak is hálás vagyok a segítségével.

Különösen hálás vagyok feleségemnek, Krisztinek, aki bátorított, és hitt bennem egész idő alatt. Nagyon hálás vagyok édesanyámnak, édesapámnak, húgomnak és az egész családomnak, akik nem késlekedtek segíteni, és támogattak a munkám során. Nagyon hálás vagyok kislányomnak, Julcsinak, aki engedett dolgozni a határidők előtti napokban, órákban is.

6 Publikációk

6.1 A szerző saját folyóirat publikációi

- [1] T. Zsedrovits, A. Zarandy, B. Vanek, T. Peni, J. Bokor, and T. Roska, “Estimation of Relative Direction Angle of Distant, Approaching Airplane in Sense-and-Avoid,” *J. Intell. Robot. Syst.*, vol. 69, no. 1–4, pp. 407–415, Jan. 2013.
- [2] A. Zarandy, M. Nemeth, Z. Nagy, A. Kiss, L. Santha, and T. Zsedrovits, “A real-time multi-camera vision system for UAV collision warning and navigation,” *J. Real-Time Image Process.*, Sep. 2014.

6.2 A szerző saját nemzetközi konferencia publikációi

- [3] T. Zsedrovits, A. Zarandy, B. Vanek, T. Peni, J. Bokor, and T. Roska, “Collision avoidance for UAV using visual detection,” in *Proc. of 2011 IEEE Int. Sym. of Circuits and Systems (ISCAS)*, 2011, pp. 2173–2176.

- [4] T. Zsedrovits, A. Zarandy, B. Vanek, T. Peni, J. Bokor, and T. Roska, “Visual Detection and Implementation Aspects of a UAV See and Avoid System,” in *Proc. of 2011 20th European Conference on Circuit Theory and Design (ECCTD)*, 2011, pp. 472–475.
- [5] T. Zsedrovits, A. Zarandy, B. Vanek, T. Peni, J. Bokor, and T. Roska, “Estimation of Relative Direction Angle of Distant, Colliding Airplane in Sense-and-avoid and Tracking,” Presented at the *International Conference on Unmanned Aircraft Systems*, Philadelphia, Pennsylvania, 2012.
- [6] T. Zsedrovits, A. Zarandy, B. Vanek, T. Peni, J. Bokor, and T. Roska, “Azimuth estimation of distant, approaching airplane in See-and-avoid Systems,” in *Proc. of 2012 13th International Workshop on Cellular Nanoscale Networks and their Applications*, Turin, Italy, 2012, pp. 1–6.
- [7] T. Zsedrovits, P. Bauer, A. Zarandy, B. Vanek, J. Bokor, and T. Roska, “Towards Real-Time Visual and IMU Data Fusion,” Presented at the *AIAA Guidance, Navigation, and Control Conference*, Reston, Virginia, 2014.

- [8] T. Zsedrovits, P. Bauer, A. Zarandy, B. Vanek, J. Bokor, and T. Roska, “Error Analysis of Algorithms for Camera Rotation Calculation in GPS/IMU/Camera Fusion for UAV Sense and Avoid Systems,” Presented at the *International Conference on Unmanned Aircraft Systems*, Orlando, Florida, 2014.

6.3 A szerző egyéb folyóirat és konferencia publikációi

- [9] B. Vanek, T. Peni, J. Bokor, T. Zsedrovits, A. Zarandy, and T. Roska, “Performance analysis of a vision only Sense and Avoid system for small UAVs,” Presented at the *AIAA Guidance, Navigation, and Control Conference*, Reston, Virginia, 2011.
- [10] B. Vanek, T. Péni, Á. Zarándy, J. Bokor, T. Zsedrovits, and T. Roska, “Performance Characteristics of a Complete Vision Only Sense and Avoid System,” Presented at the *AIAA Guidance, Navigation, and Control Conference*, 2012.
- [11] Z. Nagy, A. Kiss, A. Zarandy, T. Zsedrovits, B. Vanek, T. Peni, J. Bokor, and T. Roska, “Volume and power optimized high-performance system for UAV collision avoidance,” in *Proc. of the 2012 IEEE Int. Symp. on Circuits and Systems*, 2012, pp. 189–192.

- [12] A. Zarandy, T. Zsedrovits, Z. Nagy, A. Kiss, and T. Roska, “Visual sense-and-avoid system for UAVs,” in *Proc. of 2012 13th International Workshop on Cellular Nanoscale Networks and their Applications*, 2012, pp. 1–5.
- [13] A. Zarandy, T. Zsedrovits, Z. Nagy, A. Kiss, P. Szolgay, and T. Roska, “Cellular processor array based UAV safety system,” in *Proc. of 2012 13th International Workshop on Cellular Nanoscale Networks and their Applications*, 2012, pp. 1–2.
- [14] A. Zarandy, Z. Nagy, B. Vanek, and T. Zsedrovits, “A five-camera vision system for UAV visual attitude calculation and collision warning,” *Comput. Vis. Syst. Lect. Notes Comput. Sci.*, vol. 7963, pp. 11–20, 2013.