

SphereCalib: LiDAR és kamera autokalibrációja gömbök segítségével ^{*}

Tóth Tekla¹, Pusztai Zoltán¹, Hajder Levente¹

¹ Algoritmusok és Alkalmazásai Tanszék, ELTE IK, Budapest
{teklatoth,puzsai,hajder}@inf.elte.hu

Kivonat Ez a cikk egy új, teljesen automatikus kalibrációs eljárást mutat be LiDAR-kamera rendszerekhez. A javasolt eljárás célobjektumokat használ, mégpedig gömböket, mivel azoknak felületi pontjai illetve kontúrgörbéi precízen detektálhatóak mind a pontfelhőkben, mind pedig a kameraképeken. Szintetikus és valós tesztheink alátámasztják, hogy a kifejlesztett automatikus algoritmusunk alkalmas a külső paraméterek gyors, robusztus és pontos becslésére.

1. Bevezetés

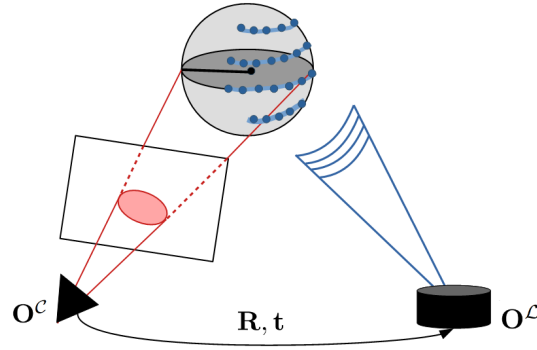
A szenzorfüzió komoly kihívást jelentő problémakör a számítógépes látás és a robotika területén. Mivel az eltérő modalitású eszközök különböző feladatokra alkalmasabbak, ezért a szenzorok kombinált használata lényegesen növelheti a környezetet észlelő szenzoros rendszer képességeit. Ezeknek az eszközöknek az egyik legfontosabb és legnépszerűbb alkalmazási területe az autonóm vezetéssel kapcsolatos rendszerek [13]. A járművek számára rendelkezésre álló szabad hely pontosan detektálható lézer alapú távérzékelőkkel, azaz LiDAR-ral. Mind-eközben gyalogosok detektálása kameraképek segítségével hatékony különböző deep learning algoritmusok használatával [21,33,37]. További jó példa, hogy egyrészt az út síkja és egyéb objektumok detektálása 3D-ben, *pl.* LiDAR adatokon, másrészt az alakzatok klasszifikálása 2D-ben, azaz kameraképeken végezhető el egyszerűbben. Az eszközök kombinált használata részletesebb vizualizációt tesz lehetővé: a 3D-s pontokat visszavetítve a képsíkra színinformáció rendelhető a térbeli koordinátához, azaz színezett pontfelhő állítható elő.

Kutatásunk során kétféle szenzor állt rendelkezésünkre: digitális kamera és LiDAR. A kamerák gyakran egyszerű, olcsó megoldást jelentenek, és nagy felbontású színes képeket állítanak elő aránylag nagy képfrissítési rátával. A kamera látószöge korlátozza, hogy a környezet mely része látható, ez több eszköz használatával kompenzálható. Nem alkalmazhatóak azonban alacsony megvilágítású,

^{*} Tóth, T., Pusztai Z., Hajder, L. „Automatic LiDAR-Camera Calibration of Extrinsic Parameters Using a Spherical Target” In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 2020, pp. 8580–8586. DOI: 10.1109/ICRA40945.2020.9197316

A fordítás néhány módosítást tartalmaz, de alapvető módszertana változatlan.

🔗 Forráskód és további eredmények: <https://github.com/tothtekla/SphereCalib>



1. ábra. Relatív pózbecslés digitális kamera és LiDAR között. A kamera és a LiDAR optikai középpontjai rendre O^C és O^L . t jelöli az eltolásvektort, R pedig az elforgatásvektort a két szenzor koordinátarendszere között.

árnyékos környezetben, valamint a takarás további kihívásokat jelent. Ugyanakkor a LiDAR aktív infravörös megvilágítási technikájának köszönhetően ezen esetekben is 3D-s pontfelhőket állít elő 360°-os körbefordulással, melyben ma-napság a pontthalmaz ritkassága és így pontossága 2 – 5 cm 15 Hz-es maximális frissítési ráta mellett. A LiDAR és a kamera ily módon képes kompenzálni egymás hibáit, így kombinált használatuk számos esetben előnyös lehet.

Ez a kutatás a LiDAR és kamera közti külső paraméterek becslésével foglalkozik. A kalibrációs eljárás a relatív merev transzformációt határozza meg a szenzorok között, ahogyan azt a sematikus illusztráció mutatja az 1. ábrán. Noha a módszert LiDAR-kamera kalibrációra teszteltük, más eszközökkel is kompatibilis lehet, pl. Time of Flight (ToF) kamerákkal, flash LiDAR-ral vagy Microsoft Kinect szenzoros adattal: a javasolt módszerek általánosak abban a tekintetben, hogy a LiDAR helyettesíthető bármilyen 3D szenzoros adattal, ha a mérésekből előállított pontfelhő legalább hasonló sűrűségű.

1.1. Irodalmi háttér

A szakirodalomban számos módszert publikáltak, ami kamera és LiDAR kalibrációval foglalkozik. Az *online kalibrációs eljárások* célja a külső paraméterek becslése az eszköz használata közben, míg az *offline kalibrációs eljárások* használatot megelőzően végzik el ezt a feladatot. Az online módszerek a környezet sajátosságait használják fel, pl. éleket és képi jellemzőket [7,18]. Az offline technikák általában kalibrációs objektumot használnak, melynek a geometriája ismert, és így pontosabb eredmény érhető el. Ez a dolgozat az utóbbi esettel foglalkozik.

Az eddigi alkalmazások különböző síkbeli vagy térbeli kalibrációs objektumokat használtak. A síklapok használata különösen elterjedt, pl. Geiger és mtsai. [14] saktáblákat használnak, míg Alismail és mtsai. [3] síkbeli alakzatokat fekete kör alakú régióval és külön jelölt középponttal. Park és mtsai. [22] fe-

hár síkbeli háromszög- vagy négyszöglapokat alkalmaznak. Vales és mtsai. [31] síkokat használnak, négy kör alakú lyuk látható egy homogén fehér háttér előtt.

Térbeli alakzatok szintén használhatóak LiDAR-kamera kalibrációhoz: Pusztai és mtsai. [24,23] valamint Gong és mtsai. [15] triéderekről készített felvételt dolgoznak fel. Az előbbi az élek metszéspontját becsüli meg LiDAR adatokon, és a doboz sarokpontjait határozza meg. Az utóbbi számos felvételt készít ugyanazokról a tárgyakról, ám ez sok emberi beavatkozást igényel. A sík alapú módszerek hátránya, hogy a síkidomot határoló élek nehezen detektálhatók változó sűrűségű LiDAR pontfelhőben, így a térbeli alakzatok használata célszerűbb. A triéder alapú megoldás hátránya, hogy a doboz sarkait manuálisan kell kiválasztani, amire még nem született robusztus, automatikus eljárás a szakirodalomban.

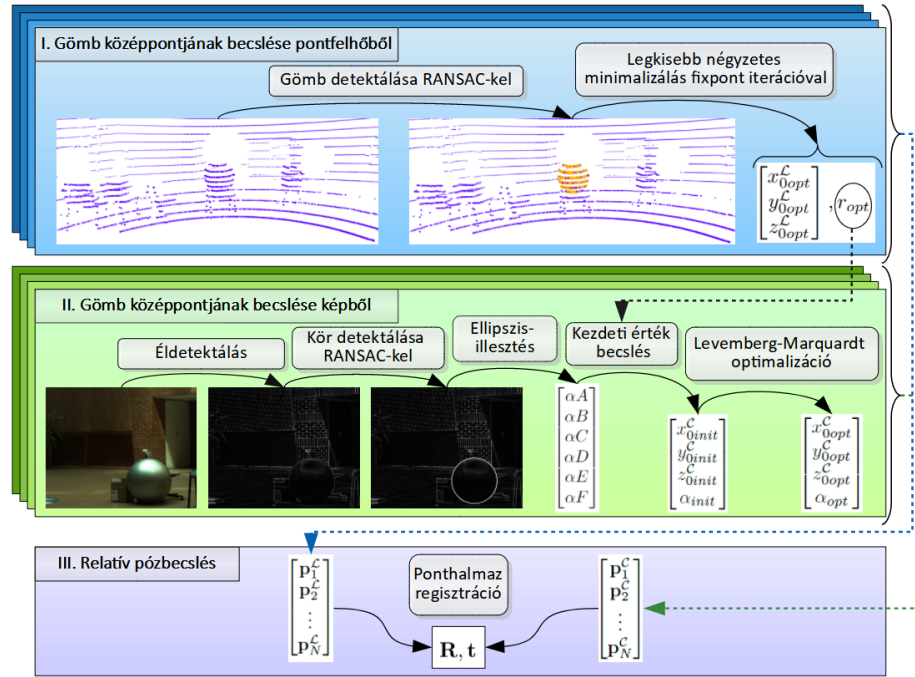
Gömböket már korábban is alkalmaztak kamerák kalibrációjára [2,8,36]. A gömb legnagyobb előnye a síkbeli alakzatokkal, pl. a sakktáblás eljárással szemben, hogy automatikusan detektálható, mivel a gömb kontúrvonala csak kis mértékben függ a kamera nézőpontjától. Ez nagyon előnyös tulajdonság többféle szenzor egyidejű alkalmazása esetén is. Gömbszerű geometriák felszíni mintapontjai LiDAR pontfelhőben jól detektálhatóak, ahogyan azt korábban megmutattuk [30].

Az előzőekben felsorolt érvek alapján választottuk a gömböt kalibrációs alakzatnak. A korszerű algoritmusok [5,19] nem használják ki minden előnyös tulajdonságát ennek a kalibrációs objektumnak. Sok munka foglalkozik RGB-D kamerák kalibrációjának javításával [5,26,29]. A gömb a képen egy kúpszeletként, konkrétan ellipszisként jelenik meg. Ezt a tulajdonságot nem használja ki pl. Kümmerle és mtsai. [19], ahol kör alapú és nem ellipszis alapú az alakzatdetektálás. Másik, ritkán felhasznált előnyös tulajdonság, hogy az ellipszis főtengelyére illeszkedő egyenes átmegy a kamerakép dőféspontján [28].

Néhány módszer első lépésként visszavetítést alkalmaz általános esetből abba a speciális esetbe, ahol a kúpmetaszt egy kör, aminek a középpontja a dőféspont, azaz a kamerapozíció és a gömb középpontja által meghatározott egyenes egybeesik az optikai tengellyel [5,27,29]. Ezek a módszerek bonyolultabb, számításigényesebb és pontatlanabb megoldást adnak az általunk javasolt módszerhez képest, mivel több művelet hajtanak végre.

Munkánk egyik fő célkitűzése az *automatikus gömb alapú offline kalibráció*. A korszerű eljárások már szintúgy manuális segítség és annotáció nélkül képesek kalibrálni, a szakirodalom szerint főként sakktáblával, 3D-s pontok, egyenesek és síkok közti megfeleltetést alkalmazó módszerek használatosak [19,32,38]. A fő kihívást a minél pontosabb objektumlokalizáció jelenti 2D-s és 3D-s adatokon.

Hozzájárulásaink. Egyrészt megmutatjuk, hogy a gömbök középpontjai precízen meghatározhatóak, ha az ellipszisek kontúrvonala detektálható egy kalibrált kamera képein, és ismert a gömb sugara. Másrészt egy új *automatikus* kalibrációs eljárást ismertetünk LiDAR-kamera rendszerekhez. A LiDAR pontfelhőből kinyerhető a gömb sugara, így az bemenetként szolgál az ellipszisdetektáláshoz a kameraképeken. A cikk két eszköz kalibrációjára fókuszál, de a javasolt algoritmus ugyanígy alkalmazható multiszenzoros esetben, így tetszőleges számú kamera és LiDAR kalibrálható össze.



2. ábra. A javasolt kalibrációs eljárás folyamata, amit a 2. fejezetben ismertettünk részletesen. Főbb lépések: (I) a gömb középpontjának becslése LiDAR pontfelhőben, (II) gömb középpontjának becslése kameraképekből, és (III) relatív pózbecslés. Mindkét középpontbecslés legfontosabb része a gömbprojekció detektálása: a gömb felszínén létrejövő ritkás 3D-s pontfelhő, illetve a 2D-s ellipszis vetület azonosítása. Kezdeti érték becslése és egy hatékony optimalizáló eljárás segíti a minél pontosabb eredmény elérését. A folyamat végén pontthalmaz regisztrációval kapjuk meg az eszközök közti egybevágósági transzformációs mátrixot, ha legalább négy különböző pozíció-pár rendelkezésre áll.

2. A javasolt kalibrációs eljárás

A javasolt kalibrációs módszer a 2. ábra segítségével tekinthető át. Az egész eljárás folyamata három főbb lépésből áll:

- I. a gömb középpontjának detektálása a pontfelhőben (2.1. fejezet)
- II. a gömb középpontjának becslése képekből (2.2. fejezet)
- III. relatív pózbecslés (2.3. fejezet)

A III. művelethez legalább négy gömb középpontja szükséges, hogy meg tudja a módszer becsülni a merev transzformációt a két szenzor között. Legfontosabb hozzájárulásunk a gömb középpontjának becslése a képeken és a pontfelhőből.

2.1. Gömb középpontjának becslése LiDAR pontfelhőben

Az egyik legfontosabb alapeladat a pontos és robusztus gömbdetekció pontfelhőkben, hogy aztán az alakzat középpontját és sugarát meg tudjuk határozni. A RANSAC algoritmus [11] használható a gömb pontjainak a detektálására. Az algoritmuson belül minimális illesztő módszert [34] használunk gömbillesztésre. Miután a legnagyobb konszenzushalmazt megbecsüljük, újraillesztést végzünk az Eberly és mtsai. [9] módszerének általunk továbbfejlesztett változatával.

Ez a módszer négyzetes hibaminimalizáláson alapul. Ha $\mathbf{p}_0 = [x_0 \ y_0 \ z_0]^T$ a térbeli gömb középpontjának koordinátái és r a gömb sugara, az Euklideszi távolság alapú hibafüggvény felírható a sugár és a becsült távolságok különbségeként. Utóbbi értékek a gömb egy felszíni pontja és a középpontja között mérhetőek. A pontokat először a baricentrumba toljuk. Ezután a probléma így formalizálható:

$$E(\mathbf{p}_0, r) = \sum_{i=1}^N (\|\mathbf{p}_0 - \mathbf{p}_i\| - r)^2, \quad (1)$$

ahol $\mathbf{p}_i = [x_i \ y_i \ z_i]^T$, $i \in \{1 \dots N\}$ a gömb i -edik felületi mintapontja. Optimális esetben E függvény parciális deriváltja x_0 , y_0 , z_0 és r mentén zérus a minimum helyen. Ez az alábbi egyenletekhez vezet:

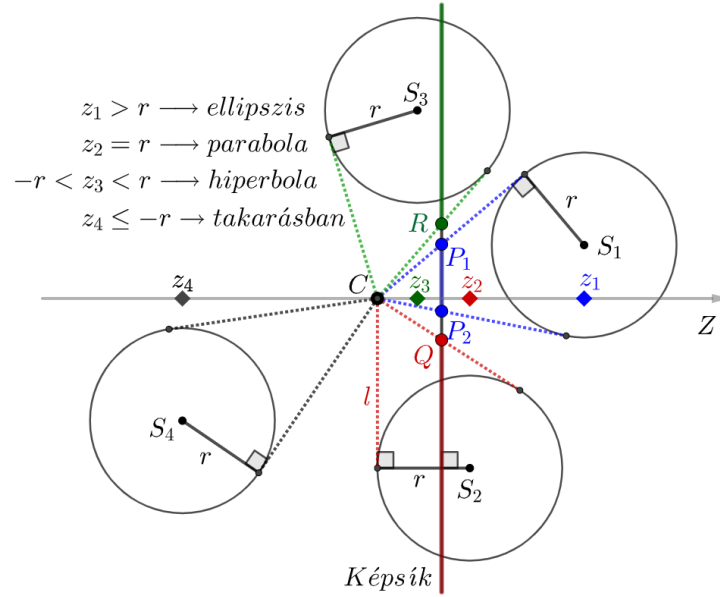
$$\mathbf{p}_0 = \frac{1}{N} \sum_{i=1}^N r \frac{\mathbf{p}_0 - \mathbf{p}_i}{\|\mathbf{p}_0 - \mathbf{p}_i\|} \quad r = \frac{1}{N} \sum_{i=1}^N \|\mathbf{p}_0 - \mathbf{p}_i\|. \quad (2)$$

Látható, hogy r és $\mathbf{p}_0$ értékei függenek egymástól. Ezek alapján egy fixponti-terációs algoritmus írható fel. Először megbecsüljük $\mathbf{p}_0$ kezdeti értékét a gömb felszíni pontjainak segítségével. Ezt követően a 2. *egyenletrendszer* segítségével az egyik lépésben kiszámoljuk r -t, majd a másik lépésben $\mathbf{p}_0$ értékét, és ezeket a lépéseket ismétljük addig, amíg a becslés nem konvergál. A konvergenciát követően transzformáljuk $\mathbf{p}_0$ -t a pontfelhő eredeti súlypontjának megfelelő eltolásvektorral.

Eberly és mtsai. [9] eredeti módszerében a középpont koordinátáinak kezdeti értéke a mintavételi pontok baricentruma. Azonban a módszer számtalan esetben pontatlan becslést eredményez LiDAR pontfelhők esetén, *pl.* az objektumok önárnyékolása miatt. Ezért a kezdeti érték a mi módszerünk esetén a négy véletlenszerűen választott mintavételi pont alapján becsülhető középponti koordináták mediánja, ami sokkal pontosabb és robusztusabb eredményt ad. Az optimalizációs lépést addig ismétljük, amíg a becslés nem konvergál. Az ennek során becsült r sugárhossz bemenete lesz a következő lépésnek, a képekből történő középpontbecslésnek, amit a következő fejezetben ismertetünk.

2.2. Gömb középpontjának becslése kameraképekből

A fejezet első része bemutatja a kameraparaméterek, a gömb paraméterei és a képsíkon megjelenő ellipszis implicit egyenletének együtthatói közt fennálló matematikai kapcsolatot. Ezt követően a gömb középpontját becselő módszert mutatjuk be. Az elméleti levezetés valamint a javasolt eljárás is egy korábbi saját elméleti munkánkon alapszik [17].



3. ábra. A gömb vetületének esetei a képsíkon a gömb Z koordinátája és r kapcsolatának függvényében. Az ábra a képsíkra merőleges síkmetszetet mutatja be, mintha $\mathbf{C}$ kamerát oldalnézetből szemlélnénk. A különböző pozícióban elhelyezkedő gömbök esetén a példák kimenete nem függ X vagy Y koordináták értékeitől, csak a Z mélységi koordinátától, és r értéke azonos mind a négy különböző esetben. S_1 gömb köré írt kúp ebben a nézőpontban P_1 és P_2 pontokban metszi a képsíkot, és ellipszisként látszik; S_2 gömbköz tartozó kúp l alkotója párhuzamos a képsíkkal, így parabolaként látszik; S_3 gömbnek egyik érintője metszi a képsíkot R -ben és hiperbolaként látszik; S_4 gömb takarásban van. A javasolt illesztő algoritmusunk az S_1 -hez hasonló, ellipszis vetületű eseteket értékeli ki.

A gömb vetületének elméleti hátttere. Tegyük fel, hogy a kamera kalibrált. Ennek megfelelően a belső paraméterek ismertek. Jelölje $\mathbf{K}$ a kameramátrixot. A koordinátarendszer jelen leírásunkban a kamerához igazított. A cél a gömb vetületének becslése a kameraképen. Jelölje a kamerakép pixeleit $[u \ v]^T$. A numerikus robusztusság érdekében kiszámítjuk az $[\hat{u} \ \hat{v}]^T$ normalizált képi koordinátákat a $\mathbf{K}^{-1}$ inverz kameramátrix segítségével az alábbi módon:

$$\begin{bmatrix} \hat{u} \\ \hat{v} \\ 1 \end{bmatrix} = \begin{bmatrix} f_u & 0 & u_0 \\ 0 & f_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}^{-1} \cdot \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{u-u_0}{f_u} \\ \frac{v-v_0}{f_v} \\ 1 \end{bmatrix}. \quad (3)$$

Egy pixel visszavetítése egy sugarat határoz meg a térben. A sugár koordinátái felírhatóak úgy, mint $[x \ y \ z]^T = z[\hat{u} \ \hat{v} \ 1]^T$. A gömb implicit reprezentációjával felírható a gömb-sugár metszéspont: $(z\hat{u} - x_0)^2 + (z\hat{v} - y_0)^2 + (z - z_0)^2 = r^2$.

Ez z változóra nézve egy másodfokú egyenlet:

$$(\hat{u}^2 + \hat{v}^2 + 1)z^2 - 2(\hat{u}x_0 + \hat{v}y_0 + z_0)z + x_0^2 + y_0^2 + z_0^2 - r^2 = 0. \quad (4)$$

Az egyenletnek pontosan egy gyöke van akkor, ha a gömböt a sugár egy pontban érinti. A képen a kontúrpointok ilyen pontok: a rajtuk átmenő vetítősugarak éppen érintik a gömböt. Ekkor a másodfokú egyenlet diszkriminánsa zérus:

$$(-2(\hat{u}x_0 + \hat{v}y_0 + z_0))^2 - 4(\hat{u}^2 + \hat{v}^2 + 1)(x_0^2 + y_0^2 + z_0^2 - r^2) = 0. \quad (5)$$

Ez az egyenlet átírható a kúpszelet implicit alakjának megfelelően

$$A\hat{u}^2 + B\hat{u}\hat{v} + C\hat{v}^2 + D\hat{u} + E\hat{v} + F = 0, \quad (6)$$

ahol az együtthatók írják le a kapcsolatot a gömb középpontjának és sugarának értékei, és az implicit ellipszisparaméterek között:

$$\begin{aligned} A &= y_0^2 + z_0^2 - r^2, & B &= -2x_0y_0, & C &= x_0^2 + z_0^2 - r^2, \\ D &= -2x_0z_0, & E &= -2y_0z_0, & F &= x_0^2 + y_0^2 - r^2. \end{aligned} \quad (7)$$

A kúpszelet egy ellipszis abban az esetben, ha a 6. *egyenlet* diszkriminánsa negatív: $\Delta = B^2 - 4AC < 0$. Ha behelyettesítjük és átrendezzük az együtthatókat a 7. egyenletből, akkor

$$\Delta = -4(z_0^2 - r^2)(x_0^2 + y_0^2 + z_0^2 - r^2). \quad (8)$$

Miután ezt a kifejezést behelyettesítjük a Δ egyenlőtlenségébe:

$$(z_0^2 - r^2)(x_0^2 + y_0^2 + z_0^2 - r^2) > 0. \quad (9)$$

Amennyiben a gömbvetület látható a képen, bevezethetünk z_0 -ra egy megkötést, ami a gömb teljes felületét szigorúan a kamera előtt definiálja: $r < z_0$. Egyéb esetben a gömb vagy a kamera mögött lenne, vagy további nem kívánatos esetet modellezne, ahogyan a 3. *ábrán* bemutatjuk. A megkötés miatt $0 < z_0^2 - r^2 \leq x_0^2 + y_0^2 + z_0^2 - r^2$, ezért a 9. egyenlőtlenség igaz, $\Delta < 0$, és a kúpszelet valóban ellipszis a képen.

Ellipszis detektálás a képen. Az első feladat a képen az élpontok meghatározása, majd ezekből az ellipszishez tartozó kontúrpointok detektálása. A gömbök használatának egyik legnagyobb előnye, hogy ez automatikusan elvégezhető a speciális tulajdonságoknak köszönhetően [35].

Az ellipszis kontúrpointjainak lehetséges halmazát a standard Canny éldetektáló algoritmus [6] eredményének küszöbölésével választjuk ki. Ezt követően a RANSAC algoritmust [11] használjuk arra, hogy kiszűrjük a valós kontúrpointokat. További megfigyelések az ellipszis együtthatókra vonatkozóan, amelyeket a 3.2. *fejezetben* a tesztek során igazolunk, hogy (i) $A \approx C$, valamint (ii) $B \approx 0$. Ezek alapján a kör egy jó közelítés a kontúrvonalak megtalálásához, mivel kör esetén a kúpszelet egyenletében $A = B$ és $B = 0$. Tehát az ellipszist megfelelő

Mintavételi pontok száma	Outlier arány				
	50 %	65 %	80 %	90 %	95 %
3	3.50×10^1	1.06×10^2	5.73×10^2	4.602×10^3	$\sim 10^4$
5	1.46×10^2	8.75×10^2	$\sim 10^4$	$\sim 10^5$	$\sim 10^7$

1. táblázat: A RANSAC algoritmus [11] átlagos szükséges iterációs számai ahhoz, hogy 99%-os valószínűséggel a keresett modellt találja meg. A különböző sorokban a minimális illesztő algoritmus 3 és 5 mintavételi pontot használ. Az oszlopok az 50%-tól 95%-ig változó outlier arányt jelölik. Látható, hogy a körrel, azaz 3 ponttal történő közelítés sokkal gyorsabb, mint az 5 pontot igénylő ellipszissel.

hibaküszöb mellett közelíthetjük egy körrel, ami azt jelenti, hogy csak három mintavételi pontra lesz szükség a minimális illesztésre azzal szemben, hogy egy ellipszis szabadsági foka öt. Ennek relevanciáját a az 1. táblázatban ismertetett átlagos iterációs számok nagyságrendi szemléltetik. Továbbá az éirányoknak a kör érintőjével egy irányba kell mutatniuk. Ezt a tulajdonságot is kihasználják a RANSAC iterációkon belül a modellillesztési eljárásunkban. Végül Fitzgibbon és mtsai. [12] direkt ellipszisillesztő módszere segítségével becsüljük meg a kúpszelet együtthatóit a kiválasztott ellipszis kontúrponatok alapján.

Mivel a kiértékelt együtthatók egy tetszőleges $\alpha > 0$ ismeretlen skaláris értékkel skálázhatóak a 6. egyenletben, jelölje a skálázott együtthatókat $\Phi = [\hat{A} \ \hat{B} \ \hat{C} \ \hat{D} \ \hat{E} \ \hat{F}]^T$. Tehát a kép alapján *pl.* közvetlenül A együttható helyett $\hat{A} = \alpha A$ határozható meg, így összességében csak Φ ismert, és α ismeretlen. Mivel a normalizált képi koordinátáknak köszönhetően Φ vektor elemei közt nincsen nagyságrendbeli különbség, Fitzgibbon és mtsai. [12] ellipszisillesztő algoritmusának utolsó lépéseként normalizáljuk ezt a vektort $\|\Phi\| = 1$.

Gömb középpontjának kezdeti becslése. Az ismeretlen skálázásnak köszönhetően a gömb középpontjának koordinátái mellett α -t is meg kell határozni. $\sqrt{\alpha}x_0$, $\sqrt{\alpha}y_0$, $\sqrt{\alpha}z_0$ és α kifejezések megbecsülhetők az imént eljelölt $\hat{A}, \dots, \hat{F}$ normalizált együtthatók és r ismeretében, a gömb sugara a LiDAR-os mérésekből kinyerhető. Egy kezdeti érték becsléséhez az $\hat{A}$, $\hat{C}$ és $\hat{F}$ együtthatókat, valamint az $\hat{B}\hat{D}/\hat{E}$ kifejezést használjuk fel, mivel az ezekhez tartozó egyenletek az alábbi mátrixos formában is felírhatóak:

$$\begin{bmatrix} 0 & 1 & 1 & -r^2 \\ 1 & 0 & 1 & -r^2 \\ 1 & 1 & 0 & -r^2 \\ -2 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \alpha x_0^2 \\ \alpha y_0^2 \\ \alpha z_0^2 \\ \alpha \end{bmatrix} = \begin{bmatrix} \hat{A} \\ \hat{C} \\ \hat{F} \\ \hat{B}\hat{D}/\hat{E} \end{bmatrix}. \quad (10)$$

Jelölje $\mathbf{M} \in \mathbb{R}^{4 \times 4}$ a bal oldali együtthatómátrixot. Az $\mathbf{M}^{-1}$ inverz mátrix segítségével felírhatók a keresett változók:

$$\begin{bmatrix} \alpha x_0^2 \\ \alpha y_0^2 \\ \alpha z_0^2 \\ \alpha \end{bmatrix} = \mathbf{M}^{-1} \begin{bmatrix} \hat{A} \\ \hat{C} \\ \hat{F} \\ \hat{B}\hat{D}/\hat{E} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & -1/2 \\ 1 & -1 & 0 & -1/2 \\ 1 & 0 & -1 & -1/2 \\ 1/r^2 & -1/r^2 & -1/r^2 & -1/r^2 \end{bmatrix} \begin{bmatrix} \hat{A} \\ \hat{C} \\ \hat{F} \\ \hat{B}\hat{D}/\hat{E} \end{bmatrix}. \quad (11)$$

Gömb középpontjának optimális becslése. A valós mérések során jelentkező zaj az együtthatókat pontatlanná teheti, ezért további optimalizáló lépés szükséges. Általánosságban az alábbi túlhatározott egyenletrendszer írható fel:

$$\begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 & -r^2 \\ 0 & 0 & 0 & -2 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & -r^2 \\ 0 & 0 & 0 & 0 & -2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -2 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & -r^2 \end{bmatrix} \begin{bmatrix} \alpha x_0^2 \\ \alpha y_0^2 \\ \alpha z_0^2 \\ \alpha x_0 y_0 \\ \alpha x_0 z_0 \\ \alpha y_0 z_0 \\ \alpha \end{bmatrix} = \begin{bmatrix} \hat{A} \\ \hat{B} \\ \hat{C} \\ \hat{D} \\ \hat{E} \\ \hat{F} \end{bmatrix}. \quad (12)$$

Általános alakban felírható úgy, mint $\mathbf{Cf} = \Phi$, ahol $\mathbf{C}$ az együtthatómátrix, $\mathbf{f}$ az x_0 , y_0 , z_0 és α változókat tartalmazó polinomiális függvények vektora. Az együtthatómátrix ismert, hiszen r a LiDAR mérések alapján megbecsülhető.

Ezek alapján egy nemlineáris legkisebb négyzetes közelítés problémáját kapjuk, ami Levenberg-Marquardt algoritmussal megoldható. A kezdeti értékeket megbecsüljük a 12. *egyenlettel*, majd az $[x_0 \ y_0 \ z_0]$ pontosabb közelítését az alábbi költségfüggvény minimalizációjaként kapjuk meg:

$$\arg \min_{x_0, y_0, z_0, \alpha} \|\Phi - \mathbf{Cf}\|^2. \quad (13)$$

2.3. Relatív pózbecslés

A gömb középpontjait megbecsüljük a kamera és a LiDAR koordinátarendszerében az előző alfejezetekben részletezett módon: a 2. *egyenlet* fixpontiterációjával, a 13. *egyenlet* megoldásával Levenberg-Marquardt optimalizációval. Ahogyan a 2. *fejezet* bevezetőjében taglaltuk, legalább négy pozíció-párra van szükségünk ahhoz hogy a koordinátarendszerek közti transzformációt fel tudjuk írni. Jelölje $\mathbf{p}_i^C$ és $\mathbf{p}_i^L$ rendre a kamera és a LiDAR adatokból becsült i -edik gömbközepet, $i \geq 4$. A minimalizálandó hibafüggvény:

$$\arg \min_{\mathbf{R}, \mathbf{t}} \sum_{i=1}^N \|\mathbf{p}_i^L - \mathbf{R}\mathbf{p}_i^C - \mathbf{t}\|_2^2, \quad (14)$$

ahol $\mathbf{R}^T \mathbf{R} = \mathbf{I}$, azaz a merev transzformációt $\mathbf{R}$ ortonormált forgatási mátrixszal, az eltolás vektort $\mathbf{t}$ -vel definiáljuk, ahogyan az 1. *ábrán* bemutattuk. $\|\mathbf{x}\|_2^2$ az

Módszer	Kalibrációs alakzat	Követelmények	Futási idő
KITTI Geiger és mtsai. [14]	≥ 4 db sakktábla mintás tábla	Teljesen automatikus, de több lehetséges felkínált megoldás közül kell választani	Néhány perc
Polygonal Park és mtsai. [22]	≥ 3 db fehér tábla	Részben automatikus, a táblákat manuálisan kell kiválasztani a 3D-s pontfelhőből	–
BoxCalibration Pusztai és mtsai. [24,23]	Egy doboz, melynek 3 merőleges oldala látható	Részben automatikus, a doboz sarkait manuálisan kell bejelölni a képen	–
SphereCalib javasolt módszer	≥ 4 db gömb	Teljesen automatikus	Néhány perc

2. táblázat: A tesztelt módszerek főbb tulajdonságai

$\mathbf{x}$ vektor L_2 normájának négyzetét jelöli. A pontregisztráció optimálisan megoldható, mi Arun és mtsai. [4] algoritmusát preferáljuk annak egyszerűsége miatt: (i) az eltolás a ponthalmaz súlyponjába történő mozgatásával kinyerhető, (ii) az eltolás korrigálása után a forgatás $\mathbf{H}$ mátrix SVD dekompozíciójával kiszámítható, ahol $\mathbf{H} = \sum_{i=1}^N \mathbf{p}_i^C (p_i^C)^T$. Ha az SVD felbontás $\mathbf{H} = \mathbf{U}\mathbf{S}\mathbf{V}^T$, az optimális forgatás $\mathbf{R} = \mathbf{V}\mathbf{U}^T$. Mivel egybevágósági transzformációt keresünk, az eredeti módszer skálázást meghatározó lépésére nincsen szükség.

Kiterjesztés multi-kamera-multi-LiDAR pózbecslésre. A pontregisztrációs algoritmus [4] nem limitált egyetlen kamera-LiDAR párra, így tetszőleges számú kamera és LiDAR közti transzformáció becslésére alkalmas.

3. Tesztek és eredmények

Ez a fejezet a konkurens módszereket mutatja be, és a szintetikus és valós tesztekkel az elért eredményeket ismerteti a javasolt módszerhez képest. A virtuális környezet, melyet Blensor [1] segítségével modelleztünk, kamerát, LiDAR-t és a szükséges kalibrációs alakzatokat tartalmazta. A szintetikus tesztek előnye, hogy a rivális módszerek kvantitatívan kiértékelhetők. A valós tesztek során a külső paraméterek minősége vizuális összehasonlítással történt, miután a LiDAR pontfelhő egy részét visszavetítettük az eredeti kameraképre.

3.1. Konkurens módszerek

Három módszerrel hasonlítottuk össze a javasolt eljárást, melyek főbb tulajdonságainak összevetése a 2. táblázatban olvasható.

A KITTI Kalibrációs Toolboxban Geiger és mtsai. [14] sakktáblák segítségével végzik el a külső kalibrációt. A sakktáblákat egyesével detektálják a képeken,

majd a kamera külső paramétereit határozzák meg, ezeket után a sakktáblák sarkait rekonstruálják. A sakktáblák síkjait úgy becsülik meg a LiDAR pontfelhőből, hogy meghatározzák minden háromdimenziós pont normálvektorát. Végezetül a képekből rekonstruált sakktáblasíkok és a LiDAR adatokon detektált síkok közti megfeleltetések adják az eszközök külső paramétereit. A síkmegfeleltetési eljárás azonban többértelmű lehet, és téves eredményt is adhat.

Park és mtsai. [22] *Polygonal* módszere fehér sokszögek használatán alapul. A legtöbb kalibrációs eljárás nehézsége az objektumok éldetektálása a LiDAR pontfelhőben. Park és mtsai. ezért egy olyan algoritmust javasolnak, amely virtuális pontokat vezet be két egymást követő háromdimenziós pont között, így téve sűrűbbé a pontfelhőt. A virtuális pontokra egyenest illesztenek, majd megbecsülik a kalibrációs objektumok éleit. Végül a sarkok vetületeit FAST [25] algoritmussal választják ki, és perspektív n -pont (PnP) problémaként határozták meg a 2D-3D megfeleltetéseket. A módszer legnagyobb hátránya, hogy a LiDAR pontok és az ezekhez tartozó FAST képi jellemző pontok kiválasztása manuálisan történik.

A Puztai és mtsai. [24,23] által javasolt *BoxCalibration* három merőleges sík, pl. egy doboz oldalainak segítségével oldja meg a kalibrációt. Ezek a síkok automatikusan detektálhatóak a LiDAR pontfelhőben szekvenciális RANSAC algoritmus segítségével, outlier szűréssel és regresszióval. Ezután a doboz sarkai az objektum ismert méretei alapján becsülhetőek meg, ám ezeket a vetületi pontokat manuálisan kell kiválasztani a kameraképről. Végül felírhatók a kamera-LiDAR pontmegfeleltetések, és a külső kalibráció egy PnP problémára egyszerűsödik, melyet E-PnP algoritmussal [20] oldanak meg. A doboz sarkainak precíz kiszámításának köszönhetően ez a módszer pontosabb eredményt ér el, mint a többi, viszont ezen pontokat a kameraképen kézzel kell kiválasztani és a doboz méretét is ismerni kell.

3.2. Szintetikus tesztek

Egy valóság-hű kalibrációs környezetet modelleztünk Blenderben Blensor LiDAR szimulációs programcsomag [16] segítségével. A képek felbontása 1920×1080 . A javasolt és a három konkurens módszert hasonlítottuk össze az ismert mérési alappal (ground truth), miközben növeltük a LiDAR pontok zajosságát. A szintér egy perspektív digitális kamerát és egy Velodyne HDL-64E2 szenzort tartalmaz.

A módszereket additív Gauss zajjal szemben teszteltük. Először a μ -t, a Gauss zaj átlagát változtattuk 0.00 és 0.08 között 0.02-es lépésközzel, miközben a szórása konstans $\sigma = 0.02$ volt. Ez azt szimulálja, mintha a gömböket egyre távolabb helyeznénk el a LiDAR-tól. Ezt követően nulla átlagú zajt adtunk a mérésekhez 0.00 és 0.08 között változó szórással 0.02-es lépésközzel. A tesztkörnyezet 5 sakktáblát, 5 fehér poligont, egy dobozt és 4 gömböt tartalmaz rendre a *KITTI*, a *Polygonal*, a *BoxCalibration* és a *SphereCalib* futtatásához.

Először a javasolt módszer gömbközéppont becslő módszerét hasonlítottuk a valós értékekhez. A 3. és 4. táblázatok rendre a LiDAR és a kamera alapú módszer eredményeit tartalmazzák. Mindegyik táblázatban a mérések átlagos hibáját és a hiba szórását ismertetjük. d_{init}^C , d_{opt}^C és d_{opt}^L a kamera kezdeti és

		Az optimális becslés hibája $d_{opt}^{\mathcal{C}}$ alapján változó μ függvényében					
μ		0.00	0.02	0.04	0.06	0.08	
avg. (m)		0.005	0.022	0.047	0.072	0.0961	
std. (m)		0.003	0.004	0.006	0.006	0.004	
		Az optimális becslés hibája $d_{opt}^{\mathcal{C}}$ alapján változó σ függvényében					
σ		0.00	0.02	0.04	0.06	0.08	0.1
avg. (m)		0.004	0.005	0.007	0.041	0.090	0.185
std. (m)		0.003	0.003	0.003	0.011	0.011	0.020

3. táblázat: Szintetikus teszteredmények: az optimális gömb középpont becslés átlagos hibája (std.) és a hiba szórása (std.) LiDAR adatokból, m -ben megadva. A felső és alsó táblázat a változó μ átlagú és σ szórású Gauss zaj függvényében mutatja a javasolt algoritmus által elért eredményeket.

		Kezdeti érték becslés hibája					Optimális becslés hibája			
		$x_{0init}^{\mathcal{C}}$	$y_{0init}^{\mathcal{C}}$	$z_{0init}^{\mathcal{C}}$	$d_{init}^{\mathcal{C}}$		$x_{0opt}^{\mathcal{C}}$	$y_{0opt}^{\mathcal{C}}$	$z_{0opt}^{\mathcal{C}}$	$d_{opt}^{\mathcal{C}}$
avg (m)		0.052	0.021	1.412	1.414		0.004	0.004	0.031	0.032
std (m)		0.086	0.033	2.208	2.210		0.003	0.001	0.010	0.011

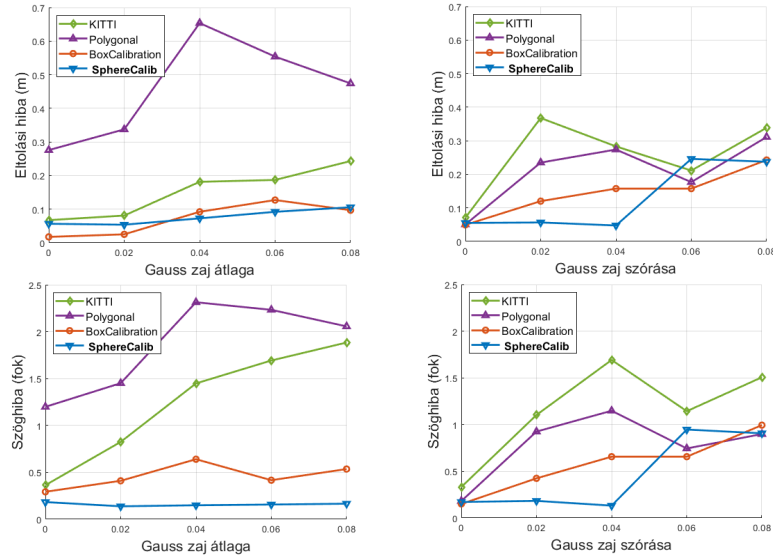
4. táblázat: Szintetikus teszteredmények: az optimális gömb középpont becslés átlagos hibája (std.) és a hiba szórása (std.) képekből, m -ben megadva. A jobb és bal táblázat a középpontbecslés kezdeti értékét és a konvergencia után kapott optimális eredményeket mutatja be az egyes változók mentén.

optimális, valamint a LiDAR gömbközepppont becslésének Euklideszi távolságát jelenti az ideális várt értékekhez képest. Kamera alapú becsléshez az egyes koordinátatengelyek menti eredményeket szintén feltüntettük. A legnagyobb hiba a Z tengely mentén figyelhető meg, ami a kamera síkjára merőleges tengely.

A szintetikus tesztek során a gömbök a szenzoroktól 15 – 20 m-es távolságban helyezkedtek el. A 4. táblázatból kiolvasható, hogy ha a kezdeti becslés 1 méteres hibát mutatott, az optimális értéket becslő algoritmus mindössze 1 – 3 cm-es hibahatár alá konvergált. Összességében elmondható, hogy a Z tengely befolyásolja legnagyobb mértékben az eredményekben észlelhető távolsághibát.

A szintetikus tesztek igazolták a feltételezéseinket az ellipszis együtthatókkal kapcsolatosan, amit a 2.2. fejezetben taglaltunk, így az ellipszist valóban egy körrel közelíthetjük. Az A/C együtthatóarány átlagosan 0.9758 volt 0.0262 szórással, továbbá B átlagos értéke 4.2952×10^{-5} volt 0.011 szórással.

Megvizsgáltuk a kalibráció külső paramétereiben jelentkező hibát is. Az eltolásvektor hibáját az ideális és mért vektorok Euklideszi távolságnormájában mértük. A forgatási hiba kiszámításához jelölje $\mathbf{R}_{\mathcal{GT}}$ és $\mathbf{R}$ az ideális és mért adatokból becsült forgatási mátrixot. Ebből kinyerhető a forgatási hiba szöge az



4. ábra. A szintetikus tesztek eredményei: az eltolás vektor távolsághibája méterben (fent) és a forgatási mátrix szöghibája fokban megadva (lent) változó Gauss zaj mellett. A jobb oldali grafikonok a változó átlagú, rögzített $\sigma = 0.02$ szórású Gauss zaj függvényében mért hibát ábrázolják. A bal oldalon a változó szórású, rögzített $\mu = 0.00$ átlagú Gauss zaj függvényében mért hiba látható.

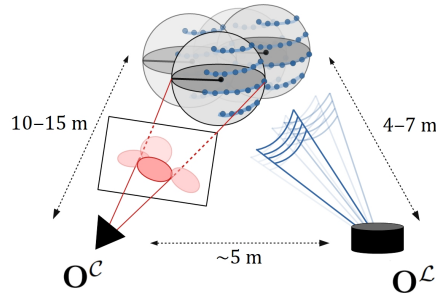
alábbi ismert képlet segítségével [10], ahol tr egy négyzetes mátrix nyomát jelöli:

$$\epsilon = \arccos \frac{\text{tr}(\mathbf{R}_{\mathcal{GT}}^T \mathbf{R}) - 1}{2}. \quad (15)$$

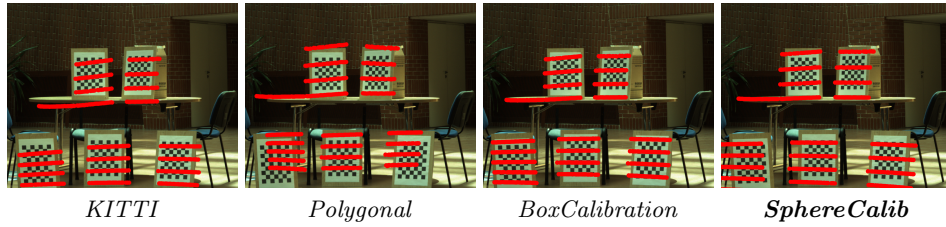
A 4. ábra a szintetikus tesztek eredményét mutatja be. A bal és jobb oszlop az eltolás és a forgatás hibáját mutatja be ebben a sorrendben. A felső sor a változó átlagú Gauss zaj, az alsó sor a változó szórású Gauss zaj esetén kapott hibát jeleníti meg. A különböző módszerekkel szemben a javasolt algoritmus kifejezetten jobban teljesít, ha a zaj átlaga változik, versenyképes abban az esetben is, ha a szórás változik. Összességében 6 cm-nál kisebb szórás mellett a javasolt módszer körülbelül 5 cm-es eltolási és 0.3° -os forgatási hibát eredményez.

3.3. Valós tesztek

Az első valós kvalitatív tesztekhez az 5. ábrán látható alapbeállításokat alkalmaztuk. A két eszközt körülbelül 5 m-es távolságban helyeztük le egymáshoz képest. A gömbök a kamerától 10 – 15 m-re, a LiDAR-tól 4 – 7 m-re helyezkedtek el. Így a LiDAR nagyjából 4 – 6 lézernyalábbal találta el a gömböt a 16 vertikálisan kibocsájtott sugárból, ami ideálisnak mondható. Tesztjeink alapján egy mérés 150-200 gömbfelszíni ponttal megbízhatónak tekinthető, és ennek elérése



5. ábra. Kvalitatív valós tesztek konfigurációja. Az O^C kamerapozíció és az O^L LiDAR pozíció, valamint a gömbök középpontjai közötti eltolás közelítő értékei.



6. ábra. Kvalitatív valós tesztek eredménye. A képek a külső paraméterek segítségével a képsíkra visszavetített LiDAR pontokat jelenítik meg az eredeti felvételen. A visszavetítéshez az asztal élét és a sakktáblák oldalait használtuk fel. Az egyes módszereknél látható kisebb elcsúszásokat a kevésbé pontos külső kalibráció okozza. Ezek alapján a *Polygonal* módszer teljesít a legrosszabbul, a *BoxCalibration* és a javasolt *SphereCalib* a legpontosabban.

érdekében a javasolt módszerhez használt objektum egy 30.00 cm sugarú gömb volt. A valós tesztek igazolták a gömb sugarának pontfelhő alapú becslésének pontosságát, mivel az átlagos eredmény 30.53 cm volt 1.45 cm-es szórással.

Kvantitatív kiértékelés itt nem volt lehetséges, mivel a valós eltolás és forgatás a szenzorok között nem ismert. Ezért az összehasonlítást a LiDAR pontfelhő egy részének visszavetítésével végeztük a kameraképekre. Ha a megbecsült külső paraméterek pontosak, a megfelelő pixelre történik visszavetítés, ha nem, akkor eltolási hibaként érzékeljük azt. Minél több ilyen eltolási hiba észlelhető, annál pontatlanabb a külső kalibráció.

A 6. ábra az ezen módszerrel vizsgált valós tesztek eredményét mutatja be. A piros körök a visszavetített LiDAR pontjai a *KITTI* kalibrációs környezet egy részének: a sakktáblák illetve az asztallap pontjait képeztük le a kameraképre. Ez azért lehetséges, mert a kamera és a LiDAR a négy különböző objektumot alkalmazó teszteset közt nem lett elmozdítva, így pontosan ugyanazt a külső kalibrációt kell megbecsülni mindegyik módszerrel. Így egy adott módszer bármely

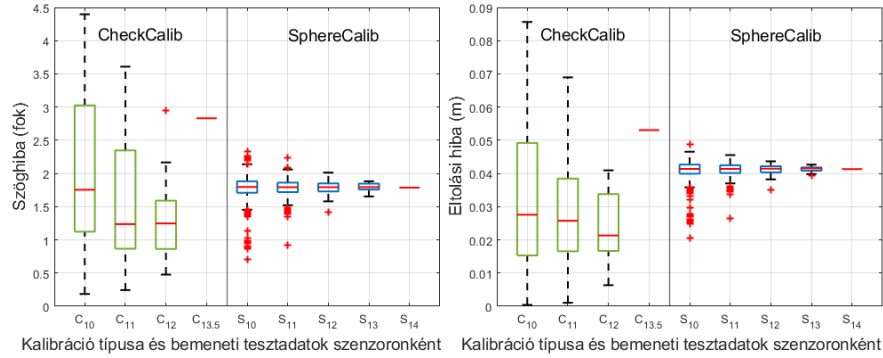
más konkurens módszer 3D-s LiDAR pontjait képes a hozzátartozó kameraképre vetíteni a megbecsült külső kalibráció segítségével. Azért a sakktáblás példát választottuk referenciaként, mert a táblák élei és sarkai környékén jól szemléltethetők az eltérések.

A *Polygonal* és a *KITTI* módszerek esetében nagyobb hiba észlelhető, mint a *BoxCalibration* és a javasolt *SphereCalib* képein. A *BoxCalibration* pontosabbnak tűnik, vélhetően a kézzel kiválasztott LiDAR síkok detektálása pontosabb az adott valós példa esetén. Ezt leszámítva a javasolt módszer a többit felülmúlja, így összességében elmondható, hogy a mi eljárásunk hasonlóan jó eredményeket produkál, mint a jelenleg a szakirodalomban fellelhető legjobb módszerek, miközben módszerünk teljesen automatizált.

Valós kvantitatív teszteket futtatunk egy új, automatizált módszerrel [39] Matlab Toolbox segítségével, ami sakktáblákat használ. Saját algoritmusunkat is továbbhangoltuk főként a képi ellipszisdetektálást illetően. A tesztek során két Hikvision MV-CA020-20GC típusú kamera Fujinon SV-0614H normál lencsékkel és 6.1 mm-es fókusz távolsággal, valamint egy Velodyne VLP-16 LiDAR állt rendelkezésünkre. A kamerákat egy 3D nyomtatott állvánnyal rögzítettük egymáshoz, ezáltal ismert a két kamera közti külső kalibráció: a forgatás 0 fok, az eltolás 0.21 m. A LiDAR 1–2 m-re helyezkedett el a kameráktól. Ezt követően az első kamerát a LiDAR-hoz, majd a LiDAR-t a második kamerához kalibráltuk. Ha az algoritmusok helyesen működnek, két lépésben ugyanazokat a külső kalibrációs paramétereket kell visszakapni, amit a 3D nyomtatott állvány garantál. Ugyanakkor fontos megjegyezni, hogy mindkét módszer pontosabb eredményt adhat két szenzor közvetlen összehangolása esetén.

Kvantitatív eredményeinket a 7. ábrán foglaltuk össze, ahol Zhou és *mt-sai*. [39] módszerét *CheckCalib* névvel tüntettük fel. Egy 9×10 mezőből álló, négyzetenként 81 mm-es oldalhosszúságú sakktáblát használtunk, ahogyan az eredeti módszerben is szerepelt. 59 elkészült tesztesetből összesen 13 illetve 14 volt a sakktáblás módszerhez megfelelő, a többit az algoritmus kiszűrte valamilyen okból. Tapasztalataink alapján az automatikus síkdetektálás LiDAR pontfelhőben nehézkesen működik. A gömb alapú tesztesetek számát ehhez igazítottuk, 30 felvételtől indultunk ki, és ez esetben ismertnek tekintettük a gömb sugarát, ami 0.25 m volt. A LiDAR pontfelhőkből 1 cm-es hibahatárral válogattuk ki a legpontosabb bemeneteket, ebből összesen 14 db volt. A vizsgált tesztesetek száma az összes lehetséges ismétlés nélküli kombináció volt a rendelkezésre álló adatokból 10–14 bemenetre, mivel mindkét módszerhez legalább 10 teszt javasolt a lehető legpontosabb eredmények eléréséhez. A *SphereCalib* szórása kisebb, szélsőértékei kevésbé kiugróak, így stabilabb eredményt ad különböző bemenetek esetén is, viszont a *CheckCalib* átlagos eredménye összességében sokszor jobb, néhány szerencsés esetben nagyon pontos eredményt produkál. Minkét módszer átlagos szöghibája 1–3 fok között mozog, az átlagos eltolási hiba 2–4 cm. Az összes tesztesetet felhasználva egy futtatást bemutató $C_{13.5}$ és S_{14} esetén a *SphereCalib* ad pontosabb eredményt.

A javasolt eljárás futási idejét illetően a szűk keresztmetszetet a RANSAC alapú gömb- és ellipszisdetektálás jeletni: az időigénye nagy mértékben függ at-



7. ábra. Kvantitatív valós tesztek dobozábrája automatikus módszerekkel. C a sakktáblás, S a gömb alapú kalibráció, az indexek a szenzorokénti bemeneti képek illetve pontfelhők számát jelentik. A $C_{13.5}$ esetben az egyik kamera és a LiDAR közt 13, a másik kamera és a LiDAR közt 14 megfeleltetés van. A kiugró adatokat piros kereszttel jelöltük. Az ábra lényegi részére történő nagyítás miatt kimaradt kiugróan rossz kalibrációs eredmények darabszáma a szöghiba esetén C_{10} : 56, C_{11} : 9, C_{12} : 1, az eltolási hiba esetén C_{10} : 45, C_{11} : 5, C_{12} : 1.

tól, hány iterációval futtatjuk a RANSAC algoritmust. A MATLAB implementáció futtatása néhány perc egy Intel Core i7-es processzoron, ami hasonló az automatikus konkurens módszerekhez képest. A *SphereCalib* egyéb lépései csak alapvető, gyors algoritmusokat használ, ezek néhány másodperc alatt lefutnak.

4. Összefoglalás

Cikkünk egy új, gömb alapú kalibrációs módszert ismertetett, amelyhez levezettük a szükséges elméleti összefüggéseket, hogy az eljárást 2D-s és 3D-s adatokon kombináltan használhassuk. A javasolt kalibráció előnye, hogy teljesen automatizált, könnyen állítható elő hozzá pontosan működő bemenet, és képes percek alatt kiértékelni az adatokat. Szintetikus és valós tesztek alátámasztják, hogy az új algoritmus versenyképes a vizsgált módszerekkel szemben, és számos esetben robusztusabb más automatikus eljáráshoz képest is.

Köszönetnyilvánítás

Nagyon köszönjük az anonim bírálók hasznos észrevételeit. EFOP-3.6.3-VEKOP-16-2017-00001 - Tehetséggondozás és kutatói utánpótlás fejlesztése autonóm járműirányítási technológiák területén. A projekt a Magyar Állam és az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg. Az Alkalmazásiterület-specifikus nagy megbízhatóságú informatikai megoldások című projekt a Nemzeti Kutatási Fejlesztési és Innovációs Alapból biztosított támogatással, a Tématerületi kiválósági program (TKP2020-NKA-06, Nemzeti Kihívások Alprogram) finanszírozásában valósult meg.

Hivatkozások

1. Blender sensor simulation. <http://www.blensor.org>, (accessed September 15, 2019).
2. M. Agrawal and L. S. Davis. Camera calibration using spheres: A semi-definite programming approach. In *9th IEEE International Conference on Computer Vision (ICCV 2003), 14-17 October 2003, Nice, France*, pages 782–791, 2003.
3. H. Alismail, L. D. Baker, and B. Browning. Automatic calibration of a range sensor and camera system. In *3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT)*, pages 286–292. IEEE, 2012.
4. K. S. Arun, T. S. Huang, and S. D. Blostein. Least-squares fitting of two 3-D point sets. *PAMI*, 9(5):698–700, 1987.
5. D. J. T. Boas, S. Poltaretskyi, J.-Y. Ramel, Y. Chaoui, J. Berhouet, and M. Slimane. Relative pose improvement of sphere based rgb-d calibration. In *Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 4: VISAPP*, pages 91–98. INSTICC, SciTePress, 2019.
6. J. Canny. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(6):679–698, June 1986.
7. J. Castorena, U. S. Kamilov, and P. T. Boufounos. Autocalibration of lidar and optical cameras via edge alignment. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2862–2866, March 2016.
8. N. Daucher, M. Dhome, and J.-T. Lapresté. Camera calibration from spheres images. In *Computer Vision - ECCV'94, Third European Conference on Computer Vision, Stockholm, Sweden, May 2-6, 1994, Proceedings, Volume I*, pages 449–454, 1994.
9. D. Eberly. Least squares fitting of data. *Magic Software*, 2000.
10. D. Eberly. Rotation representations and performance issues. *Magic Software*, 2002.
11. M. A. Fischler and R. C. Bolles. RANdom SAMpling Consensus: a paradigm for model fitting with application to image analysis and automated cartography. *Commun. Assoc. Comp. Mach.*, 24:358–367, 1981.
12. A. W. Fitzgibbon, M. Pilu, and R. B. Fisher. Direct least square fitting of ellipses. *IEEE Trans. Pattern Anal. Mach. Intell.*, 21(5):476–480, 1999.
13. A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.
14. A. Geiger, F. Moosmann, O. Car, and B. Schuster. Automatic camera and range sensor calibration using a single shot. In *IEEE International Conference on Robotics and Automation, ICRA 2012, 14-18 May, 2012, St. Paul, Minnesota, USA*, pages 3936–3943, 2012.
15. X. Gong, Y. Lin, and J. Liu. 3d lidar-camera extrinsic calibration using an arbitrary trihedron. *Sensors*, 13(2):1902, 2013.
16. M. Gschwandtner, R. Kwitt, A. Uhl, and W. Pree. Blensor: Blender sensor simulation toolbox. In *Advances in Visual Computing: 7th International Symposium, ISVC 2011, Las Vegas, NV, USA, September 26-28, 2011. Proceedings, Part II*, volume 6939, pages 199–208, 09 2011.
17. L. Hajder, T. Toth, and Z. Pusztai. Automatic estimation of sphere centers from images of calibrated cameras. In *Proceedings of the 15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, Imaging and Computer Graphics Theory and Applications - Volume 4: VISAPP*, pages 4490–497. INSTICC, SciTePress, 2020.

18. R. Ishikawa, T. Oishi, and K. Ikeuchi. Lidar and camera calibration using motions estimated by sensor fusion odometry. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7342–7349, Oct 2018.
19. J. Kümmerle, T. Kühner, and M. Lauer. Automatic calibration of multiple cameras and depth sensors with a spherical target. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1–8, Oct 2018.
20. V. Lepetit, F. Moreno-Noguer, and P. Fua. Epnnp: An accurate $\mathcal{O}(n)$ solution to the pnp problem. *Int. J. Comput. Vision*, 81:155–166, 2009.
21. W. Liu, S. Liao, W. Ren, W. Hu, and Y. Yu. High-level semantic feature detection: A new perspective for pedestrian detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.
22. Y. Park, S. Yun, C. S. Won, K. Cho, K. Um, and S. Sim. Calibration between color camera and 3d lidar instruments with a polygonal planar board. *Sensors*, 14(3):5333–5353, 2014.
23. Z. Pustai, I. Eichhardt, and L. Hajder. Accurate calibration of multi-lidar-multi-camera systems. *Sensors*, 18(7), 2018.
24. Z. Pustai and L. Hajder. Accurate calibration of lidar-camera systems using ordinary boxes. In *The IEEE International Conference on Computer Vision (ICCV) Workshops*, Oct 2017.
25. E. Rosten, R. Porter, and T. Drummond. Faster and better: A machine learning approach to corner detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(1):105–119, 2010.
26. M. Ruan and D. Huber. Extrinsic calibration of 3d sensors using a spherical target. In *International Conference on 3D Vision*, 2014.
27. D. Schnieders and K.-Y. K. Wong. Camera and light calibration from reflections on a sphere. *Comput. Vis. Image Underst.*, 117:1536–1547, 2013.
28. G. Shivaram and G. Seetharaman. A new technique for finding the optical center of cameras. In *Proceedings 1998 International Conference on Image Processing. ICIP98 (Cat. No.98CB36269)*, volume 2, pages 167–171 vol.2, Oct 1998.
29. A. Staranowicz, G. R. Brown, F. Morbidi, and G. L. Mariottini. Easy-to-use and accurate calibration of rgb-d cameras from spheres. In Reinhard Klette, Mariano Rivera, and Shin’ichi Satoh, editors, *Image and Video Technology*, pages 265–278. Springer Berlin Heidelberg, 2014.
30. T. Toth and L. Hajder. Robust fitting of geometric primitives on lidar data. In *Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 5: VISAPP.*, pages 622–629. INSTICC, SciTePress, 2019.
31. M. Velás, M. Španěl, Z. Materna, and A. Herout. Calibration of rgb camera with velodyne lidar. In *WSCG 2014 Communication Papers Proceedings*, volume 2014, pages 135–144. Union Agency, 2014.
32. S. Verma, J. S. Berrio, S. Worrall, and E. Nebot. Automatic extrinsic calibration between a camera and a 3d lidar using 3d point and plane correspondences. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, page 3906–3912. IEEE Press, 2019.
33. X. Wang, T. Xiao, Y. Jiang, S. Shao, J. Sun, and C. Shen. Repulsion loss: Detecting pedestrians in a crowd. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7774–7783, June 2018.
34. S. Yesudasan. Fast geometric fit algorithm for sphere using exact solution. *CoRR*, abs/1506.02776, 2015.
35. X. Ying and Z. Hu. Catadioptric camera calibration using geometric invariants. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(10):1260–1271, 2004.

36. H. Zhang, K.-Y. K. Wong, and G. Zhang. Camera calibration from images of spheres. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(3):499–502, 2007.
37. S. Zhang, L. Wen, X. Bian, Z. Lei, and S. Z. Li. Occlusion-aware r-cnn: Detecting pedestrians in a crowd. In V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, editors, *Computer Vision – ECCV 2018*, pages 657–674, Cham, 2018. Springer International Publishing.
38. L. Zhou, Z. Li, and M. Kaess. Automatic extrinsic calibration of a camera and a 3d lidar using line and plane correspondences. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2018, Madrid, Spain, October 1-5, 2018*, pages 5562–5569. IEEE, 2018.
39. L. Zhou, Z. Li, and M. Kaess. Automatic extrinsic calibration of a camera and a 3d lidar using line and plane correspondences. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5562–5569, 2018.