Univerza v Ljubljani

Fakulteta za elektrotehniko

Rok Mandeljc

Kalibracija sistema za sledenje v realnem času Ubisense s pomočjo metod za kalibracijo kamer

Diplomsko delo univerzitetnega študija

Mentor: prof. dr. Stanislav Kovačič

Ljubljana, 2010

Zahvala

Rad bi se zahvalil vsem, ki so kakorkoli prispevali k nastanku tega dela; v prvi vrsti mentorju prof. dr. Stanislavu Kovačiču za stalno razpoložljivost, koristne nasvete in strokovno vodstvo ter ostalim članom Laboratorija za strojni vid, še posebej dr. Janezu Peršu in dr. Mateju Kristanu, za pomoč, potrpežljivost in podporo.

Posebna zahvala gre tudi mojim staršem za pomoč in podporo v času študija (pa tudi sicer); hvala, ker sta mi omogočila, da se ukvarjam s tem, kar me zanima in veseli ter tako med drugim omogočila tudi nastanek tega dela.

In ne nazadnje — hvala bogu, da je diplomska naloga napisana...

Rok Mandeljc

V Ljubljani, 6. september 2010

Povzetek

Diplomska naloga je posvečena iskanju postopka kalibracije sistema UWB za radijsko lociranje in sledenje v realnem času, ki bi s strani uporabnika zahteval čim manj napora, predvsem v smislu meritev. Pri tem smo se osredotočili na sistem UWB, ki ga ponuja podjetje Ubisense. Vgrajeni postopek kalibracije, ki ga je razvil Ubisense, namreč zahteva poznavanje položajev senzorjev in uporabo drage merilne opreme za določanje le-teh, kar močno omejuje možnosti za uporabo sistema.

V predlaganem alternativnem postopku senzorje obravnavamo kot navidezne kamere in jih kalibriramo z metodami za kalibracijo kamer. To storimo s pomočjo zajema meritev kotov, pod katerimi senzorji sprejemajo signal z značk. Izmerjene kote prihoda z ustreznim modelom kamere projeciramo v navidezne slikovne ravnine in tako dobljene slike uporabimo za kalibracijo zunanjih parametrov navideznih kamer. Kalibracijo smo najprej poskusili izvesti z uporabo avtokalibracijske metode T. Svobode, nato pa še z Zhangovo metodo in z metodo DLT — slednji spadata med klasične kalibracijske metode.

Ideja alternativnega postopka je predstavljena v kontekstu uporabe sistema Ubisense za sledenje igralcem v športu. Predlagali smo novo praktično metodo za izvedbo kalibracije v športni dvorani in jo podrobno analizirali. Ob predstavitvi rezultatov smo opisali in analizirali tudi težave, ki izhajajo predvsem iz lastnosti in načina delovanja sistema Ubisense.

Ključne besede: *ultraširoki spekter, UWB, Ubisense, sistem za sledenje v realnem času, kalibracija, senzor, navidezna kamera, koti prihoda, navidezna slikovna ravnina*

Abstract

This work is dedicated to calibration of UWB Real-Time Location Systems and development of a calibration method that would require minimal user effort, especially with regard to measurements. Specifically, the Real-Time Location System from Ubisense was used, as the built-in calibration method, developed by Ubisense, requires manual measurements of sensors' positions and use of expensive measuring equipment, thus severely limiting system's deployment possibilities.

In the considered alternative approach, sensors are treated as virtual cameras and are calibrated using camera calibration methods. This is done by capturing the raw angle of arrival measurements for each sensor; using appropriate camera model, the angles of arrival are then projected into virtual image planes and the resulting images are used to calibrate virtual cameras' external parameters. The calibration was first attempted using the autocalibration method developed by T. Svoboda and afterwards using classic calibration methods — the DLT method and the method developed by Zhang.

The idea of this alternative calibration approach is presented in the context of rapid deployment of Ubisense system for tracking in sports. We propose and thoroughly analyze a new practical method for calibration of a system deployed in a sports hall. Along with the results we also detail the problems arising from the properties and modus operandi of the Ubisense system.

Key words: *ultra-wideband, UWB, Ubisense, Real-Time Location System, calibration, sensor, virtual camera, angles of arrival, virtual image plane*

Vsebina

1 Uvod **1**

1.1 Zgradba diplomske naloge 4

2 UWB radijsko lociranje in sistem Ubisense **5**

2.1 UWB — tehnologija ultraširokega spektra 5

2.2 Osnovni principi radijskega lociranja 7

2.2.1 Lociranje na osnovi izmerjene moči signala 9

2.2.2 Lociranje na osnovi časov prihoda 10

2.2.3 Lociranje na osnovi razlik v časih prihoda 10

2.2.4 Lociranje na osnovi kotov prihoda 11

2.3 Sistem za sledenje v realnem času Ubisense 11

2.3.1 Arhitektura sistema Ubisense 13

2.3.2 Celice, senzorji in značke 14

2.3.3 Časovna okna in frekvenca javljanja značk 17

2.4 Vgrajeni postopek kalibracije sistema Ubisense 17

2.4.1 Namestitev senzorjev v prostoru in določitev njihovega položaja 19

2.4.2 Kalibracija senzorjev 19

2.5 Odboji in problem več poti signala 21

2.6 Dostop do podatkov s sistema 22

2.7 Alternativna metoda kalibracije 24

3 Metode za kalibracijo kamer **27**

3.1 Model kamere 28

3.2 Direktna linearna transformacija — DLT 30

3.2.1 Parametri DLT 33

3.2.2 Distorzija zaradi leče 34

3.2.3 Iskanje vrednosti parametrov DLT 35

3.2.4 Določitev parametrov kamere 36

3.2.5 Rekonstrukcija s pomočjo DLT 37

3.2.6 Napaka reprojekcije 38

3.2.7 Napaka rekonstrukcije 38

3.2.8 Modificirana metoda DLT 38

3.2.9 Pomanjkljivosti metode 39

3.3 Methoda Zhengyou Zhanga 40

3.3.1 Osnovne enačbe 40

3.3.2 Kalibracija kamere 42

3.3.3 Implementacija v programskem okolju Matlab 45

3.3.4 Pomanjkljivosti metode 48

3.4 Metoda Tomáša Svobode 49

3.4.1 Teoretično jedro metode 50

3.4.2 Implementacija metode v programskem okolju Matlab . . . 52

3.4.3 Parametri kamer in poravnava koordinatnega sistema . . . 55

3.4.4 Pomanjkljivosti metode 56

4 Kalibracija sistema Ubisense z metodami za kalibracijo kamer **57**

4.1 Motivacija za iskanje alternativnega postopka 58

4.2 Senzorji kot navidezne kamere 59

4.2.1 Notranji in zunanji parametri navideznih kamer 60

4.3 Avtokalibracija s pomočjo surovih podatkov 62

4.3.1 Eksperiment in rezultati 62
4.3.2 Teoretični koti prihoda 64
4.3.3 Eksperiment z manjšim številom meritev 64
4.4 Avtokalibracija s pomočjo filtriranih podatkov 67
4.4.1 Eksperiment in rezultati 67
4.4.2 Uporaba mediane 69
4.5 Primerjava treh metod 70
4.5.1 Testno okolje 71
4.5.2 Rezultati 71

5 Diskusija in sklep **79**
5.1 Predlagani praktični postopek kalibracije 80
5.2 Nadaljnje delo 82

Literatura **83**

Dodatek A **87**

Poglavje 1

Uvod

Na področju lociranja s pomočjo radijskih valov se je v zadnjem času precej uveljavila relativno nova tehnologija ultraširokega spektra (*angl. ultra-wideband — UWB*). Zaradi svojega načina delovanja je tehnologija primerna tudi za uporabo v zaprtih prostorih, kjer zaradi odbojev in problema zakrivanja konvencionalne radijske metode lociranja in sledenja odpovedo.

Danes je komercialno dostopnih več sistemov za sledenje osebam in predmetom [1, 2, 3]. V večini primerov gre za mrežo časovno sinhroniziranih UWB sprejemnikov, ki v prostoru sledijo velikemu število majhnih, cenenih in baterijsko napajanih UWB oddajnikov. Navedena natančnost tovrstnih sistemov je običajno dva do tri decimetre.

UWB sisteme za sledenje danes lahko srečamo na različnih področjih: proizvodnja, vojska, prevoz, varstvo otrok, sledenje živini, sledenje igralcem v športu, itd.

Eden od komercialno dosegljivih sistemov je tudi sistem podjetja Ubisense [3]. Ubisense med drugim ponuja tudi t. i. raziskovalni paket, ki vsebuje štiri senzorje, deset značk ter pripadajočo programsko opremo in je v prvi vrsti namenjen akademskim ustanovam, ki se ukvarjajo z raziskavami in razvojem na področju lociranja in sledenja v realnem času.

Sledenje igralcem v športu je eno od raziskovalnih področij Laboratorija za strojni vid na Fakulteti za elektrotehniko v Ljubljani, kjer so se letos odločili za nakup zgoraj omenjenega raziskovalnega paketa z namenom dopolnitve obstoječih premičnih video sistemov za sledenje igralcem. To bi storili tako, da bi Ubisense senzorje namestili skupaj z video kamerami na stojala in jih prinesli na prizorišče

športnega dogodka (npr. v športno dvorano). Primer namestitve senzorjev na stojala je prikazan na Sliki 1.1.

Slika 1.1: *Primer namestitve senzorjev na stojala za kamere*

Izkazalo se je, da je pri takšni premični postavitvi sistema Ubisense poleg same postavitve največji izziv njegova kalibracija. Praktično vse danes komercialno dosegljive UWB sisteme za sledenje kalibriramo s postopki, ki so vgrajeni v pripadajočo programsko opremo in jih je razvil proizvajalec sistema; edino vidnejše delo s področja iskanja alternativnih postopkov za kalibracijo je članek [13], ki ga bomo podrobneje analizirali v Podpoglavju 2.7.

V pripadajočo programsko opremo vgrajeni postopek za kalibracijo sistema Ubisense ima s stališča premične postavitve veliko slabost — za izvedbo postopka kalibracije moramo poznati položaje senzorjev. Za sistem Ubisense je namreč predvidena fiksna namestitev. Senzorje montiramo na mesta, za katera smo v postopku načrtovanja ugotovili, da zagotavljajo optimalno pokritost prostora. Po

namestitvi se položaji senzorjev ne spreminjajo, zato je potrebno sistem kalibrirati samo enkrat, ob namestitvi. Ker gre za načrtovano postavitev, je domneva, da so položaji senzorjev znani, upravičena — bodisi jih lahko preberemo z načrta postavitve, bodisi jih moramo izmeriti, da zagotovimo skladnost z načrtom. Zato je upravičena tudi domneva, da imamo na voljo ustrezno opremo za določanje položajev senzorjev — lasersko merilno postajo (teodolit).

Pri premični postavitvi moramo kalibracijo sistema izvesti vsakič, ko stojala s senzorji prinesemo in namestimo v športni dvorani. Senzorjev običajno ne moremo namestiti tik ob igrišču (na katerega je pripet koordinatni sistem), pač pa jih zaradi varnosti in boljše pokritosti namestimo na tribune. Zaradi tega težko izmerimo položaj senzorjev, še posebej, če nimamo laserske merilne postaje.

Po drugi strani pa kamere, ki so nameščene na stojala skupaj s senzorji, kalibriramo z metodami, pri katerih poznavanje položaja kamer ni potrebno — te določimo skupaj z orientacijami tekom postopka kalibracije. Običajno za kalibracijo kamer potrebujemo nekaj kontrolnih točk, v zadnjem času pa je bilo napravljenih precej raziskav na področju t. i. avtokalibracije, ki poskušajo odpraviti tudi to zahtevo.

Cilj diplomske naloge je zato raziskati možnost alternativnega postopka kalibracije sistema Ubisense, s katerim bi se predvsem izognili merjenju položajev senzorjev. Ker diplomska naloga nastaja pod okriljem Laboratorija za strojni vid in v okviru integracije sistema Ubisense v video sistem za sledenje igralcem, se zdi smiselno, da raziščemo možnost uporabe metod za kalibracijo kamer tudi za kalibracijo sistema Ubisense.

Prispevek diplomske naloge je večplasten. Glavni prispevek je predlagani alternativni postopek za kalibracijo sistema Ubisense, po drugi strani pa smo se poglobili tudi v samo delovanje sistema Ubisense in težave, ki izhajajo iz njegovih lastnosti. Poleg tega smo pri kalibraciji sistema Ubisense z metodami za kalibracijo kamer analizirali tako kvaliteto surovih podatkov, ki jih dobimo s senzorjev, kot tudi obnašanje uporabljenih kalibracijskih metod, zlasti s stališča robustnosti.

1.1 Zgradba diplomske naloge

Diplomsko nalogo poleg pričujočega uvodnega poglavja sestavljajo še štiri poglavja.

V DRUGEM POGLAVJU bomo predstavili sistem Ubisense, skupaj s tehnologijo ultraširokega spektra (UWB) in osnovnimi koncepti radijskega lociranja. Pri tem se bomo osredotočili predvsem na značilnosti in koncepte, ki so neposredno povezani s kalibracijo sistema in pridejo do izraza pri iskanju alternativnega postopka kalibracije.

Podobno bomo v TRETJEM POGLAVJU podali osnove kalibracije kamer; predstavili bomo centralno projekcijski model kamere in podrobneje opisali tri metode za kalibracijo kamer, nato pa si bomo ogledali še dve implementaciji v programskem okolju Matlab [23].

Vsebino, predstavljeno v drugem in tretjem poglavju, bomo nato združili v ČETRTEM POGLAVJU, kjer bomo predstavili idejo o kalibraciji Ubisense senzorjev z metodami za kalibracijo kamer, skupaj z eksperimenti, s katerimi smo analizirali lastnosti predlagane metode.

Na koncu, v PETEM POGLAVJU poglavju, so povzete ugotovitve iz četrtega poglavja; podana je tudi končna različica predlaganega alternativnega postopka za kalibracijo sistema Ubisense, pri kateri senzorje obravnavamo kot navidezne kamere, ki jih kalibriramo z metodami iz drugega poglavja.

Poglavje 2

UWB radijsko lociranje in sistem Ubisense

V tem poglavju bomo predstavili sistem za radijsko lociranje Ubisense. Slednji temelji na tehnologiji ultraširokega spektra (UWB), ki je na kratko predstavljena v Podpoglavju 2.1. Sledi Podpoglavje 2.2, v katerem so opisane osnovne metode radijskega lociranja glede na različne lastnosti sprejetega signala.

Nato sledi predstavitev sistema Ubisense (Podpoglavje 2.3); na kratko so predstavljene arhitektura sistema, njegovo delovanje in posamezne komponente (senzorji in značke) ter njihove glavne značilnosti.

V Podpoglavju 2.4 je predstavljen vgrajeni postopek kalibracije sistema Ubisense, v Podpoglavju 2.5 se dotaknemo problema odbojev in več poti, v Podpoglavju 2.6 so opisane možnosti dostopa do (predvsem surovih) podatkov, na koncu, v Podpoglavju 2.7, pa je na kratko predstavljen še edini vidnejši alternativni postopek za kalibracijo UWB sistemov za lociranje, ki pa, kot se bo izkazalo, ne zadošča za kalibracijo sistema Ubisense.

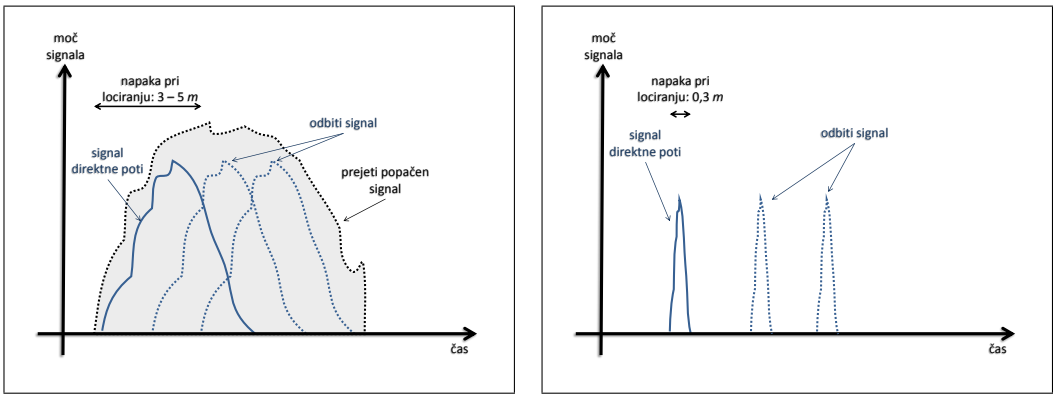
2.1 UWB — tehnologija ultraširokega spektra

Tehnologija ultraširokega spektra (*angl. Ultra-wideband — UWB*) je radijska tehnologija, katere razvoj v vojaške namene poteka že od štiridesetih let prejšnjega stoletja. V primerjavi s tem je uporaba v civilne namene relativno nova; ameriški regulatorni organ za telekomunikacije (*Federal Communications Comission — FCC*) je leta 2002 izdal pravilnik, s katerim dovoljuje civilno uporabo UWB

tehnologije v omejenem frekvenčnem pasu od $3,1$ do $10,6$ GHz pri zelo majhni moči.

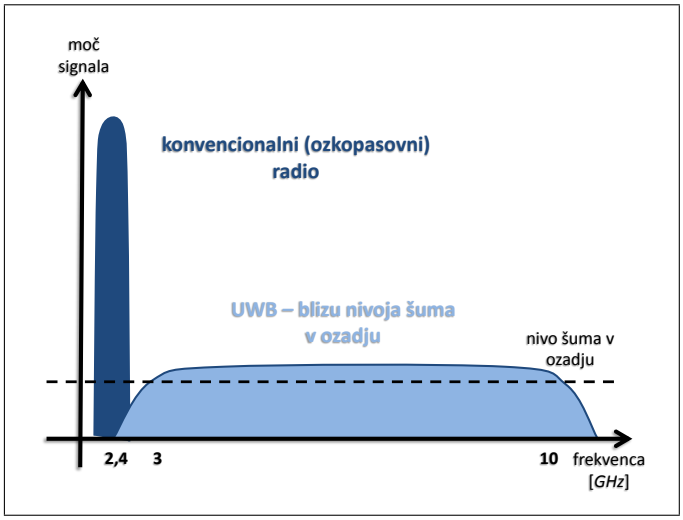
Po definiciji FCC je UWB tehnologija "vsakršna radijska tehnologija, katere signali istočasno in trenutno zasedajo frekvenčni pas, katerega širina je večja bodisi od 20% centralne operativne frekvence bodisi od absolutne pasovne širine 500 MHz" [15, 33]. UWB poznamo tudi pod imenom impulzna radijska tehnologija oziroma radijska tehnologija brez nosilca, saj omogoča radijsko komunikacijo brez uporabe radijskega nosilca — namesto tega uporablja modulirane pulze energije, ki trajajo manj kot 1 pikosekundo, njihova energija pa se razprostira čez zelo širok pas frekvenčnega spektra.

Energija vsakega pulza je skoraj enakomerno razporejena po vsem UWB frekvenčnem pasu (običajna širina slednjega je od 1 do 2 GHz in je pogojena s časom trajanja pulzov). Sprejemnik zazna prisotnost pulza v času neodvisno od frekvence, zato je sistem precej odporen na motnje na posameznih frekvencah (npr. zaradi delovanja drugih radijskih naprav, ali zaradi prisotnosti ovire, ki te frekvence ne prepušča). Ker imajo UWB pulzi izredno kratko valovno dolžino, v primeru odbojev le redko pride do prekrivanja odbitih signalov s pravim signalom, kar je sicer pogosta težava pri običajnih ozkopasovnih radijskih sistemih (Slika 2.1). Poleg tega je porazdeljena energija pulza pri katerikoli frekvenci blizu nivoja šuma, kar pomeni, da UWB signali praktično ne motijo običajnih radijskih naprav, ki uporabljajo RF-nosilce (Slika 2.2).



(a) (b)

Slika 2.1: *Odboji v primeru konvencionalnega ozkopasovnega radijski signala (a) in v primeru UWB pulzov (b)*



Slika 2.2: *Spekter UWB signala in konvencionalnega ozkopasovnega radijskega signala*

Zaradi izredno majhne predpisane moči imajo UWB sistemi praviloma kratek doseg, po drugi strani pa zaradi kratkega trajanja pulzov omogočajo veliko hitrost prenosa podatkov. Eno od področij uporabe UWB tehnologije je tako brezžična povezava med napravami — primer so brezžična osebna omrežja PAN in brezžični USB.

Drugo področje, na katerem se tehnologija UWB uveljavlja v zadnjem času, pa sta radijsko lociranje in sledenje v realnem času. Zasluga za to gre po eni strani predvsem neobčutljivosti na odboje in zmožnosti natančnega določanja časov prihoda pulzov ter posledično relativno visoki prostorski ločljivosti, po drugi strani pa so tovrstni sistemi zaradi majhne moči primerni tudi za okolja, ki so občutljiva na radijske signale, na primer bolnišnice. Poleg tega kratki časi trajanja UWB pulzov omogočajo uporabo relativno velikega števila oddajnih značk v prostoru. Tako je danes komercialno dosegljivih več različnih sistemov za radijsko sledenje osebam in predmetom, ki izkoriščajo zgoraj naštete prednosti tehnologije UWB.

2.2 Osnovni principi radijskega lociranja

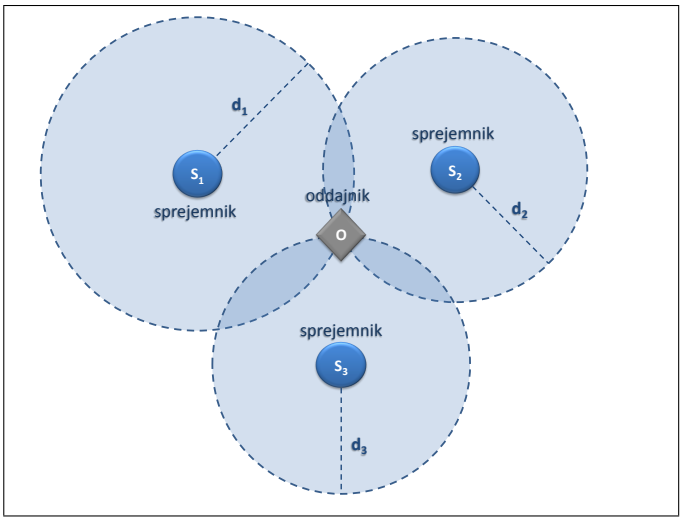
Problem radijskega lociranja lahko povzamemo kot problem določitve položaja oddajnika radijskega signala na osnovi poznavanja položaja (in po potrebi

orientacije) sprejemnikov in ene ali več lastnosti sprejetega signala na mestu vsakega sprejemnika:

- moč signala
- čas prihoda signala oziroma razlika v časih prihoda
- kot prihoda signala

Obstajata dva splošno razširjena in matematično dodelana postopka določitve položaja oddajnika — trilateracija in triangulacija. Trilateracija je metoda iskanja presečišča treh sfer s središči v položajih sprejemnikov in polmeri, ki ustrezajo ocenjeni razdalji med oddajnikom in sprejemnikom (Slika 2.3). Razdaljo lahko določimo na primer iz izmerjene moči ali časa prihoda signala.

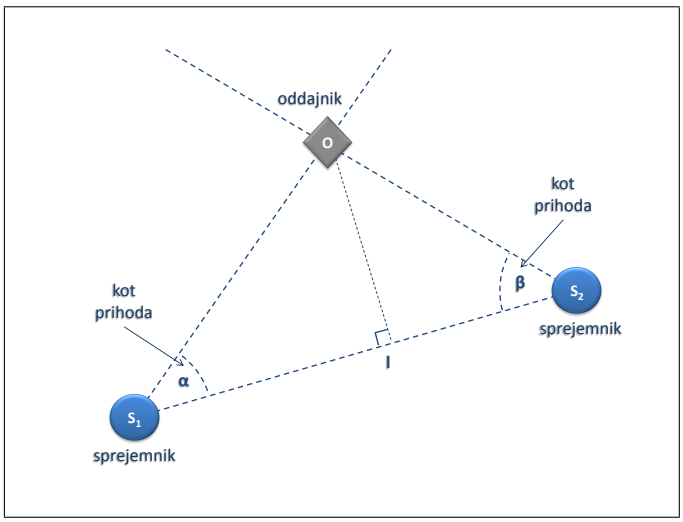
Postopek triangulacije pa temelji na poznavanju položajev (vsaj) dveh sprejemnikov ter kotov, pod katerimi sprejemnika sprejemata signale z oddajnika (Slika 2.4).



Slika 2.3: *Trilateracija*

Glede na informacijo o signalu, ki jo uporabljajo, lahko sisteme za radijsko lociranje v grobem razdelimo na sisteme, ki delujejo na osnovi izmerjene moči signala, sisteme, ki delujejo na osnovi časovnih meritev (bodisi čas prihoda bodisi razlika v časih prihoda) in sisteme, ki delujejo na osnovi kotov prihoda signala.

Večina danes komercialno dosegljivih sistemov (npr. sistema podjetij SandLinks [1] in TimeDomain [2]) deluje na osnovi časovnih meritev, predvsem



Slika 2.4: *Triangulacija*

na osnovi časa prihoda signalov. Sistem, ki ga ponuja podjetje Ubisense [3], pa deluje na osnovi kombinacije razlik v časih prihoda in informacije o kotih prihoda signalov.

Lociranje na osnovi moči signala v UWB sistemih ni preveč razširjeno, predvsem zaradi v nadaljevanju navedenih slabosti, se pa uporablja npr. za preprostejše določanje prisotnosti v prostoru (*angl. presence detection*).

2.2.1 Lociranje na osnovi izmerjene moči signala

Pri tej metodi ocenimo razdaljo med oddajnikom in sprejemnikom na osnovi izmerjene moči signala (*angl. Received Signal Strength Indicator — RSSI*) na sprejemni strani. Poleg izmerjene moči signala na sprejemni strani moramo poznati še oddajno moč signala in model slabljenja signala na prenosni poti.

Ko s pomočjo modela iz moči sprejetega signala ocenimo razdaljo med oddajnikom in sprejemnikom, lahko določimo položaj oddajnika z metodo trilateracije, za kar potrebujemo meritve z vsaj treh sprejemnikov.

Prednost sistema, ki bi temeljil na uporabi te metode, je predvsem nezahtevnost v smislu časovne sinhronizacije in preciznosti sistemske ure. Po drugi strani pa metoda zahteva natančen model prenosne poti in je zato zelo občutljiva na spremembe obnašanja prenosne poti (npr. prisotnost ovir). Prav

tako metoda ne izkorišča visoke prostorske ločljivosti, ki jo omogoča UWB tehnologija.

2.2.2 Lociranje na osnovi časov prihoda

Najbolj razširjena metoda lociranja v UWB sistemih temelji na merjenju časa, ki ga signal potrebuje za pot od oddajnika do sprejemnika — t. i. čas prihoda (*angl. Time of Arrival — ToA*). Sprejemniki morajo biti časovno sinhronizirani, kot referenčno točko za določitev časa prihoda signala pa običajno vzamemo začetek časovnega okna (določen z uro sprejemnika, s katerim se usklajujejo ostali sprejemniki).

Podobno kot pri metodi lociranja na osnovi izmerjene moči signala tudi pri tej metodi ocenimo razdaljo med oddajnikom in sprejemnikom — tokrat iz časa prihoda t in znane hitrosti širjenja radijskega valovanja v prenosnem mediju c:

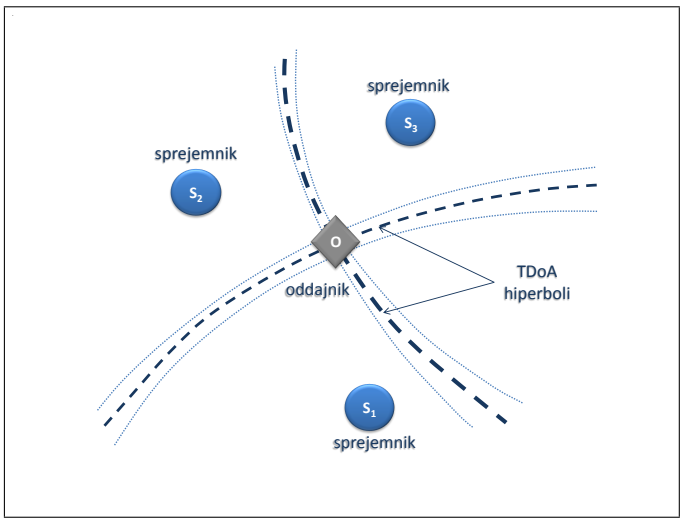
$$d = c \cdot t \tag{2.1}$$

Z izračunanimi razdaljami med sprejemniki in oddajnikom lahko določimo položaj slednjega z metodo trilateracije, za kar potrebujemo meritve z vsaj treh sprejemnikov.

2.2.3 Lociranje na osnovi razlik v časih prihoda

Metoda lociranja na osnovi razlik v časih prihoda (*angl. Time Difference of Arrival — TDoA*) je precej podobna metodi lociranja na osnovi časa prihoda, le da namesto absolutnih časov prihoda signala uporabimo razlike v časih prihodov med pari sprejemnikov.

To metodo poznamo tudi pod imenom multilateracija ali hiperbolično lociranje, saj od vsakega para sprejemnikov dobimo hiperboloid z gorišči v položajih sprejemnikov (Slika 2.5). Položaj oddajnika določimo kot presečišče vsaj dveh hiperboloidov, kar pomeni, da za to metodo potrebujemo vsaj tri sprejemnike (dva para).



Slika 2.5: *Lociranje na osnovi razlik v časih prihoda*

2.2.4 Lociranje na osnovi kotov prihoda

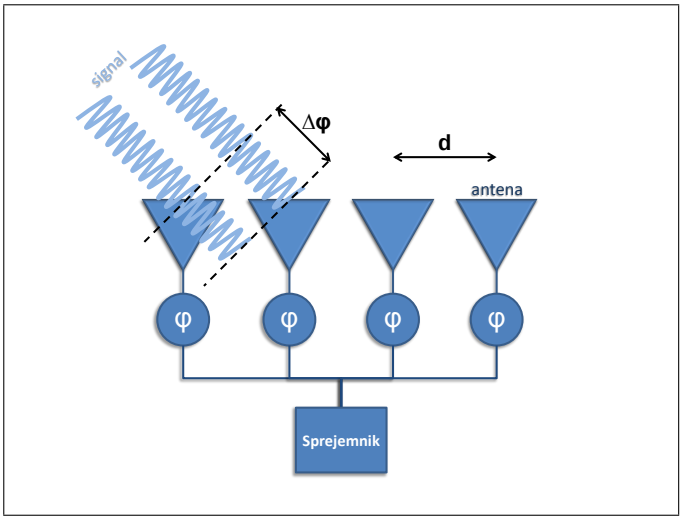
Metoda lociranja na osnovi kotov prihoda temelji na meritvi kota, pod katerim signal z oddajnika prispe v sprejemnik — t. i. kot prihoda (*angl. Angle of Arrival — AoA*). V primeru 3-D lociranja lahko kot prihoda razdelimo na horizontalno komponento (azimut) in vertikalno komponento (elevacija).

Z izmerjenimi koti prihoda lahko položaj oddajnika določimo s pomočjo triangulacije; za lociranje zadostujeta že dva sprejemnika.

Za merjenje kotov prihoda potrebujemo sprejemnik z usmerjeno anteno (Slika 2.6). Ta je sestavljena iz niza anten, ki sprejemajo vpadni signal; merimo časovne razlike med prihodi signala na različne elemente niza oziroma fazno razliko med sprejetim signalom na posameznih elementih. Iz informacije o fazni razliki in ob poznavanju razdalje med antenami lahko določimo vpadni kot signala.

2.3 Sistem za sledenje v realnem času Ubisense

Angleško podjetje Ubisense [3] s sedežem v Cambridgeu je eden od vodilnih proizvajalcev sistemov za lociranje in sledenje v realnem času (*angl. Real-time Location System — RTLS*), ki delujejo na osnovi tehnologije UWB. Podjetje je bilo ustanovljeno leta 2003, leto pozneje pa je njihov sistem za lociranje s strani FCC prejel dovoljenje za uporabo v Združenih državah. Večji razcvet



Slika 2.6: *Zgradba in delovanje usmerjene antene*

je podjetje doživelo leta 2007, ko so pridobili precej pomembnejših strank in razvojnih partnerjev. Istega leta je Evropska unija pripravila svoj pravilnik o uporabi UWB tehnologije in podjetju Ubisense izdala dovoljenje za splošno uporabo njihovega sistema v državah članicah (do tedaj je bil sistem namreč na voljo samo v raziskovalne in preizkusne namene) [6].

Sistem za lociranje Ubisense se je do sedaj uveljavil na številnih in precej raznolikih področjih; najpogostejša je uporaba v proizvodnih obratih in logističnih podjetjih, predvsem za sledenje opremi in vozilom, pa tudi za izboljšanje poteka proizvodnih procesov in zagotavljanje varnosti delavcev v nevarnih okoljih. Sistem omogoča lociranje in sledenje v realnem času z natančnostjo vsaj 30 cm tudi na razdaljah, večjih od 100 m; pomembna lastnost UWB tehnologije je sposobnost razlikovanja med signali, sprejetimi po direktni poti in signali, sprejetimi prek odbojev s sten ali kovinskih površin, zaradi česar je sistem primeren tudi za uporabo v zaprtih prostorih. Nizka moč signalov Ubisense sistema zagotavlja, da sistem ne moti ostalih brezžičnih in radijskih sistemov in je primeren tudi za okolja, ki so občutljiva na radijske signale.

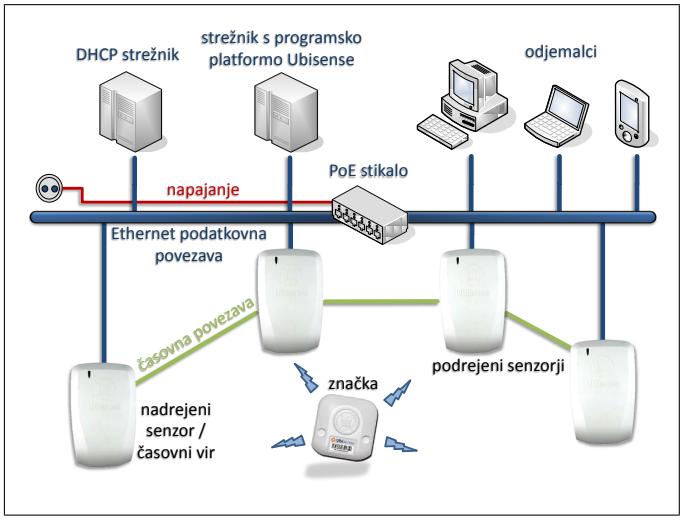
Sistem Ubisense tako lahko najdemo v logističnih podjetjih, v avtomobilski in letalski industriji, v različnih proizvodnih obratih in nevarnih okoljih, na razstavah, na področju navidezne resničnosti in iger, v vojski, v skladiščih in prodaji. Sistem se uveljavlja tudi v zdravstvu, npr. v bolnišnicah, ter na področju nege in varovanja starejših oseb.

V našem primeru gre za uporabo sistema Ubisense za sledenje igralcem v športu, in sicer bodisi kot samostojen sistem bodisi kot dopolnitev obstoječih video sistemov za sledenje, ki temeljijo npr. na metodah [11, 18, 19, 24].

2.3.1 Arhitektura sistema Ubisense

Arhitektura sistema je prikazana na Sliki 2.7. Podobno kot ostale tovrstne sisteme za lociranje in sledenje v realnem času, tudi sistem Ubisense v grobem sestavljajo tri komponente:

- Senzorska strojna oprema: UWB senzorji in značke.
- Ubisense programska platforma za lociranje in sledenje, ki teče na osebnem računalniku.
- Različni že razviti odjemalci ter skupek razvojnih orodij, ki omogočajo integracijo prostorskega modela in podatkov o lokacijah v poljubne aplikacije.



Slika 2.7: *Arhitektura sistema Ubisense*

Vsak od senzorjev je standardna Ethernet naprava, ki je prek Ethernet omrežja povezana z osebnim računalnikom, na katerem teče programska platforma Ubisense. Komunikacija med senzorji in storitvami znotraj programske platforme Ubisense poteka po TCP/IP protokolu, in sicer z UDP paketki in načinom oddajanja več prejemnikom hkrati (*angl. multicast*). Omrežje,

v katerega so povezani senzorji, mora zato podpirati tovrstno komunikacijo, senzorjem in računalniku s programsko platformo pa morajo biti dodeljeni ustrezni naslovi IP; to najlažje dosežemo z namenskim DHCP strežnikom.

Na vsakem od senzorjev teče program, ki ga senzor ob zagonu v obliki zagonske slike naloži z računalnika, na katerem teče programska platforma. Program poskrbi za krmiljenje senzorja in obdelavo prejetih signalov ter posredovanje meritev programski platformi.

Programska platforma prejete meritve obdela in izračuna položaj značke; podatki o položajih značk so na voljo odjemalcem, ki so prek Ethernet omrežja povezani s programsko platformo.

Poleg podatkovne Ethernet povezave so senzorji med seboj povezani še z Ethernet kablom za časovno sinhronizacijo. Gre za običajen Ethernet kabel, po katerem se senzorji časovno usklajujejo prek lastniškega "signala za časovno sinhronizacijo". Vsak senzor ima en Ethernet priključek, ki služi kot vhod za časovno sinhronizacijo in pet priključkov, ki služijo kot časovni vir za ostale senzorje, s čimer lahko dosežemo najugodnejši način povezave glede na dano postavitev; hrbtna stran senzorja je prikazana na Sliki 2.8.

Senzorji podpirajo tudi napajanje prek Etherneta (*angl. Power over Etherner — PoE*), če jih seveda priključimo na stikalo, ki to omogoča. Pri tem moramo upoštevati, da vsak senzor med obratovanjem potrebuje približno 10 W moči.

2.3.2 Celice, senzorji in značke

Osnovna enota sistema Ubisense je celica (*angl. cell*). Celice običajno ustrezajo posameznim prostorom v zgradbi, čeprav sistem omogoča tudi kompleksnejše postavitve, ki vključujejo več celic v enem prostoru ali več manjših prostorov v eni celici. V celici so fiksno nameščeni senzorji — UWB sprejemniki, ki sprejemajo UWB pulze v frekvenčnem pasu $6 - 8\ GHz$. Ti prihajajo z oddajnikov — značk, ki se trenutno nahajajo v celici in so pripete oziroma pritrjene na osebe in predmete, ki jim želimo slediti.

V vsaki celici en senzor služi kot časovni vir (*angl. time source*) za časovno sinhronizacijo, en senzor (lahko isti, ni pa nujno) pa je nadrejen (*angl. master*). Naloga slednjega je dvosmerna komunikacija z značkami, in sicer prek klasičnega $2,4\ GHz$ telemetričnega radijskega kanala. Značka je aktivna samo v primeru

Slika 2.8: *Ubisense Series 7000 senzor; na hrbtni strani se nahaja šest RJ45 priključkov za časovno sinhronizacijo in en RJ45 priključek za podatkovni kabel*

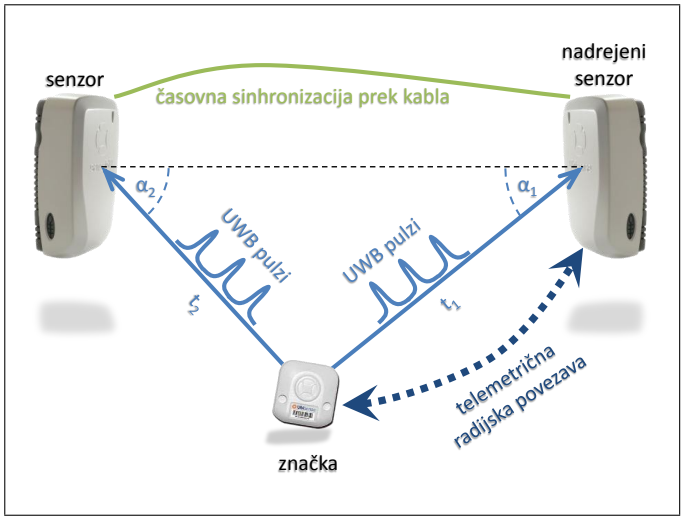
vzpostavljene telemetrične povezave, zato mora nadrejeni senzor pokrivati celotno celico.

Sistem Ubisense omogoča vzpostavitev telemetrične povezave prek enega od osmih različnih radio kanalov. S tem po eni strani omogoča nemoteno delovanje več celic v istem prostoru, po drugi strani pa kanale lahko izberemo tako, da ne motimo ostalih naprav, ki delujejo na $2,4\ GHz$ frekvenčnem območju (npr. brezžična omrežja).

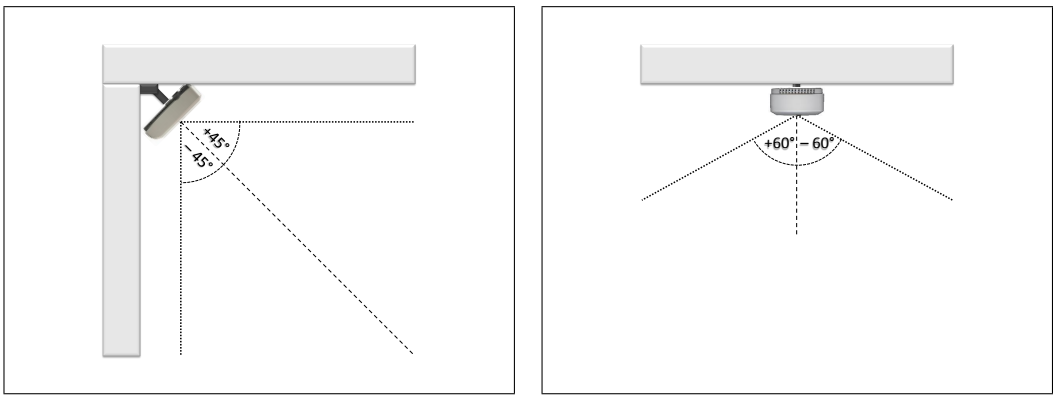
Poleg prenosa informacij z značke (npr. pritisk gumba, stanje baterije) in nastavljanja parametrov značke (npr. frekvenca javljanja) je telemetrična povezava namenjena tudi proženju značk; na začetku časovnega okna, ki znački pripada, nadrejeni senzor namreč pošlje znački ukaz za oddajo UWB pulza.

Senzorji, ki so usmerjene antene, določijo kot prihoda signala ter razliko v časih prihoda glede na časovni vir (Slika 2.9). Prek podatkovne Ethernet povezave meritve posredujejo nadrejenemu senzorju, ki določi položaj značke in ga posreduje programski platformi; prek le-te so podatki o položajih značk na voljo odjemalcem.

Nazivno "vidno polje" oziroma kot, ki ga vsak senzor pokriva, je približno 90° v vertikalni in 120° v horizontalni smeri. Točnost meritev kotov prihoda pada z naraščanjem azimuta in elevacije, medtem ko točnost meritev razlik v časih prihoda ostaja nespremenjena, tudi pri kotih, ki presegajo nazivne. Potrebno je



Slika 2.9: *Delovanje senzorjev in značk*



(*a*) (*b*)

Slika 2.10: *Vidno polje senzorja — pogled s strani (a) in pogled od zgoraj (b)*

poudariti, da pri nazivnih kotih ne gre za ostre meje, temveč bolj za smernice pri postavitvi senzorjev (Slika 2.10).

Sistem Ubisense vključuje dve vrsti značk (oddajnikov UWB pulzov), ki sta prikazani na Sliki 2.11. Prva so t. i. "vitke značke" (*angl. Slim Tag*) dimenzij $83 \times 42 \times 11$ mm; gre za usmerjene oddajnike, ki oddajajo signal samo v eni smeri in so zato posebej primerni za nošenje. Druga vrsta so t. i. "kompaktne značke" (*angl. Compact Tag*), ki so, kot pove že ime, nekoliko kompaktnejše; gre za vsesmerni oddajnik v ohišju dimenzij $38 \times 39 \times 16,5$ mm. Kompaktne značke so namenjene bodisi nošenju bodisi pritrditvi na opremo. V obeh primerih gre za UWB oddajnik, ki ga napaja baterija; ta naj bi zaradi nizke porabe moči

(a) (b)

Slika 2.11: *Znački — Ubisense Series 7000 Slim Tag (a) in Compact Tag (b)*

zadostovala za štiri leta neprekinjenega delovanja (javljanje vsake tri sekunde). Nazivni operativni doseg značk je do 160 m (v odprtih prostorih brez ovir), z dosegljivo natančnostjo lociranja pod 30 cm.

2.3.3 Časovna okna in frekvenca javljanja značk

Ker je za lociranje na voljo samo en UWB kanal, je v vsaki celici v danem trenutku lahko aktivna samo ena značka. Sistem zato razdeli razpoložljivi čas v celici v časovna okna in vsaki znački dodeli število časovnih oken glede na želeno frekvenco javljanja posamezne značke. Usklajevanje aktivnosti značk poteka prek telemetričnega radijskega kanala; vsaka značka je tako aktivna samo znotraj svojih časovnih oken, preostali čas pa je v stanju pripravljenosti.

Trajanje časovnega okna je odvisno od frekvence delovanja sistema. Ta je lahko bodisi 40 Hz bodisi 160 Hz. V prvem primeru časovno okno traja natanko $27,023$ ms, v drugem pa $6,5$ ms. V Tabeli 2.1 so prikazane periode javljanja značk v odvisnosti od števila časovnih oken; število uporabljenih časovnih oken je odvisno od števila značk v celici.

2.4 Vgrajeni postopek kalibracije sistema Ubisense

Kot smo omenili že pri predstavitvi trilateracije in triangulacije, moramo za določitev položaja oddajnika poznati položaje sprejemnikov. V primeru sistema Ubisense moramo poznati tudi orientacije senzorjev, saj senzorji določijo kot

Št. časovnih oken	**Perioda javljanja** $[s]$	
	40 Hz	160 Hz
4	0,109	0,026
8	0,216	0,052
16	0,432	0,104
32	0,865	0,208
64	1,73	0,416
128	3,46	0.832
256	6,91	1,66
512	13,8	3,32
1024	27,7	6,66
2048	55,3	13,3
4096	111	26,6
8192	221	53,2
16384	221	106
32768	221	106

Tabela 2.1: *Perioda javljanja značk v odvisnosti od števila časovnih oken*

prihoda signala v svojem lastnem koordinatnem sistemu (relativno na svojo orientacijo). Zaradi časovne sinhronizacije senzorjev prek kabla je pri določanju razlik v časih prihoda potrebno upoštevati tudi zakasnitev, ki je posledica končne hitrosti širjenja signala po kablu.

Vse zgoraj naštete lastnosti sistema določimo v postopku kalibracije. Za kalibracijo UWB sistemov za lociranje uporabimo postopek, ki ga je razvil proizvajalec in je vgrajen v pripadajočo programsko platformo. Postopek namestitve, kalibracije in zagona sistema Ubisense je podan v priročniku *Ubisense Location Engine Config Manual* [22] in obsega naslednje korake:

1. Namestitev senzorjev v prostoru.
2. Določitev položajev senzorjev.
3. Namestitev in zagon programske platforme.
4. Dodajanje senzorjev v celico.
5. Načrtovanje celice.
6. Nastavitev razpona značk.
7. Zagon senzorjev.

8. Določitev nivoja šuma v ozadju.
9. Aktivacija značk.
10. Kalibracija orientacije senzorjev in zakasnitev zaradi kablov.
11. Preverjanje pravilnosti delovanja.

Ker je za sistem Ubisense predvidena fiksna namestitev in se položaj senzorjev po namestitvi načeloma ne spreminja, je treba postopek kalibracije izvesti samo enkrat, ob namestitvi.

Posamezni koraki postopka so podrobno opisani v [22], v nadaljevanju pa se bomo dotaknili samo tistih sklopov, ki so neposredno povezani s temo diplomske naloge.

2.4.1 Namestitev senzorjev v prostoru in določitev njihovega položaja

Za sistem Ubisense je predvidena fiksna namestitev; nosilci za senzorje so namenjeni fiksni montaži na steno ali strop. Senzorje običajno namestimo v vogale prostora in čim višje (običajno pod strop), da dosežemo čim boljšo pokritost prostora. Obrnemo jih tako, da so usmerjeni proti tlom in v sredino prostora. Poleg tega zagotovimo, da je nagib senzorjev (zasuk okoli vzdolžne osi) enak nič, kar preverimo s pomočjo vodne tehtnice.

Za določitev položaja senzorjev podjetje Ubisense priporoča uporabo laserske merilne postaje (teodolit), s katero izmerimo položaj senzorjev glede na izhodišče izbranega koordinatnega sistema. Pri tem si pomagamo z oznako, ki se nahaja pod odstranljivim pokrovom na čelni strani senzorja.

Položaje senzorjev lahko določimo tudi kako drugače, vendar je za optimalno delovanje sistema priporočljivo, da so položaji izmerjeni na vsaj pet centimetrov natančno.

Določitev položaja senzorjev je nujna za nadaljnjo kalibracijo s postopkom, vgrajenim v programsko platformo Ubisense.

2.4.2 Kalibracija senzorjev

Po namestitvi in priključitvi senzorjev, ter nato namestitvi in zagonu jedra programske platforme, v programu *Location Engine Config* ustvarimo celico.

Vanjo dodamo senzorje, določimo nadrejeni senzor in časovni vir, vnesemo približne orientacije senzorjev in določimo nivo šuma v ozadju vsakega senzorja. Na koncu aktiviramo značke, ki jim želimo slediti, in njihove oznake dodamo v celico.

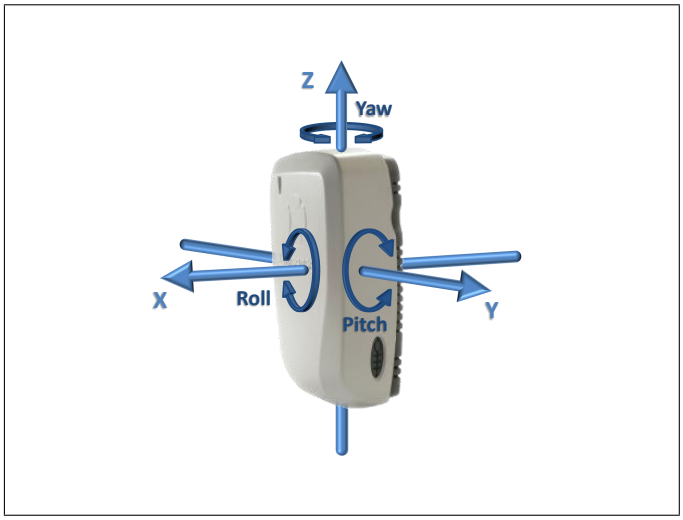
S tem dobimo celico, ki je skoraj pripravljena za uporabo. Treba je določiti samo še dejanske orientacije senzorjev in dolžine (oziroma časovni zamik) kablov — torej korak, ki (poleg določanja položaja senzorjev) predstavlja dejansko kalibracijo sistema.

Orodje *Location Engine Config* omogoča tri načine kalibracije. Najpreprostejša je t. i. polna kalibracija (*angl. full calibration*), pri kateri značko postavimo na pet različnih točk v prostoru, in sicer tako, da je višina vedno enaka, točke pa morajo biti med seboj oddaljene vsaj dva metra. Značko na vsaki točki pustimo približno deset sekund, tako da sistem zbere dovolj meritev. Sistem nato na osnovi znane višine (slednjo lahko izmerimo z metrom) in meritev s senzorjev določi orientacijo vseh senzorjev ter dolžine (časovne zamike) kablov.

Postopek polne kalibracije ne uspe vedno, zlasti v prostorih z velikim številom odbojnih površin. V tem primeru vsak senzor kalibriramo s postopkom parne kalibracije (*angl. dual calibration*). Kot pove že ime, izvajamo parno kalibracijo med pari senzorjev — običajno med nadrejenim in enim od podrejenih senzorjev. Postopek zahteva, da značko postavimo v točko, katere vse tri koordinate poznamo; najprimernejše so točke, ki se nahajajo pred obema senzorjema. Sistem nato na osnovi več meritev v izbrani točki ter poznanih koordinat točke določi orientacijo in dolžine kablov za oba senzorja.

Parni kalibraciji je ekvivalentna uporaba naprej postopka kalibracije orientacije (*angl. orientation calibration*), ki določi orientacijo senzorja, in kalibracije kabla (*angl. cable calibration*), ki določi dolžino kabla. Oba postopka izvedemo za vsak senzor posebej, podobno kot parna kalibracija pa vključujeta zajem več meritev v točki z znanimi koordinatami.

Za opis orientacije senzorjev sistem Ubisense uporablja RPY kote, zapis, ki se uporablja predvsem za opis orientacije letal in ladij. Kot je prikazano na Sliki 2.12, orientacijo senzorja opišemo s tremi koti: nagibom (zasuk okoli osi X — *angl. roll*), naklonom (zasuk okoli osi Y — *angl. pitch*) in odklonom (zasuk okoli osi Z — *angl. yaw*).



Slika 2.12: *Orientacija senzorja in RPY koti*

Priročnik *Ubisense Location Engine Config Manual* zahteva, da ob postavitvi senzorji nimajo nagiba, zato vgrajeni postopki kalibracije tega kota ne določajo in predpostavljajo, da je enak nič stopinj.

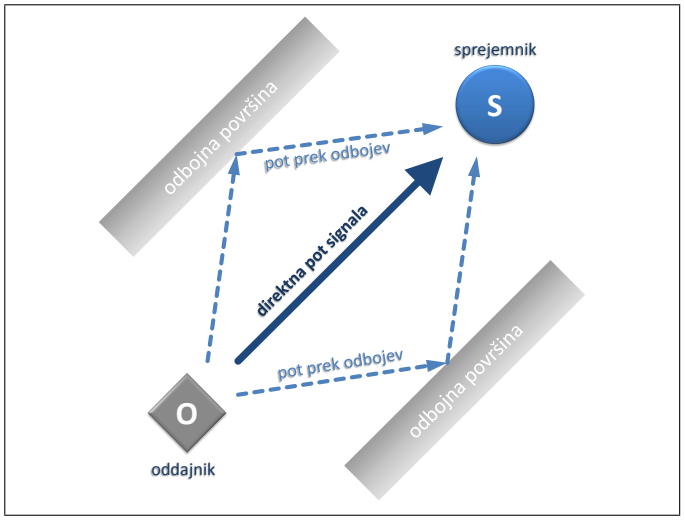
2.5 Odboji in problem več poti signala

Enega od večjih problemov pri uporabi sistemov za radijsko lociranje predstavljajo odbojne površine v prostoru, med katerimi so najočitnejši:

- Večji kovinski objekti, npr. kovinski tramovi, rezervoarji in cevi.
- S kovino prevlečene steklene površine.
- Pregradni zidovi z vgrajeno kovinsko mrežo.
- Žične ograje z luknjami, manjšimi od 5 *cm*.

Del signala z oddajnika namreč do sprejemnika pride po direktni poti, del pa se odbije od kovinskih površin v prostoru in prispe do sprejemnika po daljši poti prek odbojev (Slika 2.13). S stališča sprejemnika deli signala prispejo pod različnimi koti, obenem pa so zaradi različnih dolžin poti tudi fazno zamaknjeni.

Pri uporabi konvencionalnega radia že zaradi tega nastopijo težave, saj zaradi interference med fazno zamaknjenimi deli signala pride do njegove popačitve



Slika 2.13: *Odboji*

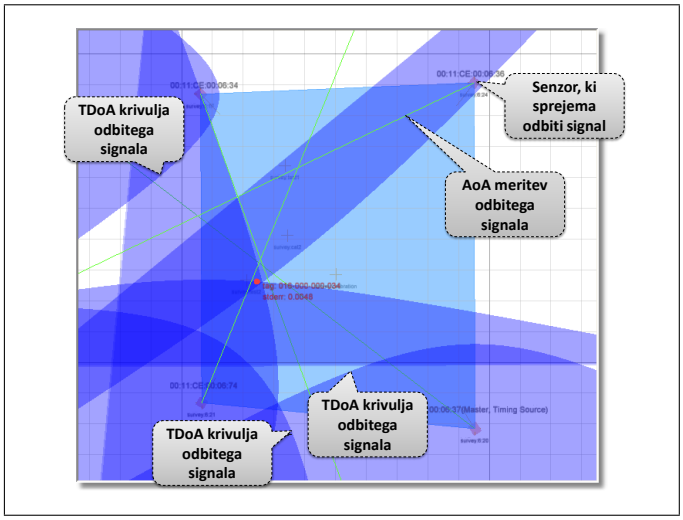
(glej Sliko 2.1 v Podpoglavju 2.1). Uporaba UWB sistemov ta problem odpravi. V primeru odbojev zaradi izredno kratkega časa trajanja pulzov med fazno zamaknjenimi deli signala ne pride do interference in sistem prejme niz nepopačenih pulzov. Ker je direktna pot najkrajša, kot in čas prihoda prvega pulza ustrezata dejanski lokaciji oddajnika, vsi nadaljnji pulzi znotraj časovnega okna pa so posledica odbojev.

Problem nastopi v primeru ovirane direktne poti; v tem primeru sprejemnik določi kot prihoda, ki ustreza lokaciji zadnjega odboja v najkrajši poti prek odbojev ter čas prihoda, ki ustreza dolžini najkrajše poti prek odbojev.

Slika 2.14 prikazuje primer, ko eden od senzorjev zaradi ovire med njim in značko sprejme signal z značke prek odbojev. Senzor posledično napačno določi kot prihoda (zelena črta), prav tako pa so narobe določene tri razlike v časih prihodov (temno modre parabole). Sistem Ubisense je na osnovi meritev z ostalih senzorjev ter s pomočjo Kalmanovega filtra [14] in modela gibanja značke vseeno sposoben izločiti napačne meritve in pravilno določiti položaj značke (rdeča pika).

2.6 Dostop do podatkov s sistema

Podatki s sistema Ubisense, ki jih odjemalci potrebujejo za sledenje značkam v realnem času, so na voljo prek Ubisensovih .NET knjižnic; te omogočajo



Slika 2.14: *Primer odboja in problema več poti*

gradnjo grafičnih aplikacij s podporo shranjevanja lokacij, obveščanja o dogodkih v prostoru, vnos modela zgradbe, 2D in 3D vizualizacijo, itd. Do istih podatkov lahko dostopamo tudi brez uporabe Ubisensovih knjižnic, in sicer tako, da v programu *Location Engine Config* nastavimo poljuben ponor (*angl. sink*) — IP naslov in vrata, na katera naj programska platforma v realnem času pošilja UDP paketke z obdelanimi podatki. Paketke nato lahko preberemo s svojim programom, ki posluša na danem naslovu in vratih.

Ta Ubisensov "On-The-Wire" protokol [5] obsega sporočila o vstopu značke v celico, izstopu iz celice, pritisku na gumb na znački in lociranju značke. Vsa sporočila vsebujejo oznako značke in zaporedno številko časovnega okna, sporočila o lociranju značke pa vsebujejo še lokacijo značke (x, y in z) ter standardno napako izmerjene lokacije. Ti podatki zadoščajo za implementacijo programa za sledenje značkam v realnem času.

Možen je tudi dostop do surovih podatkov s senzorjev (koti prihoda, razlike v časih prihoda, itd.), vendar na žalost ne v realnem času (razen morda za razvojne partnerje podjetja Ubisense) [4]. Program *Location Engine Config* namreč beleži dogodke (*angl. events*), ki nastopijo v vsakem časovnem oknu, v katerem senzor poskuša locirati eno od značk.

Vsak dogodek vsebuje oznako značke in zaporedno številko časovnega okna, obdelane podatke (lokacijo značke in standardno napako) ter podatke s senzorjev

(MAC naslov senzorja, položaj in orientacijo senzorja, dolžino kabla, kot prihoda, podatek o časovni razliki prihoda, moč prejetega signala).

Program *Location Engine Config* omogoča izvoz največ 10000 preteklih dogodkov v obliki datoteke .xcm, v kateri so dogodki zapisani v obliki XML. Podatki o kotih prihoda so shranjeni v značkah *azimuth_* in *elevation_* ter predstavljajo vrednost ustreznega kota v radianih. Podatki o razlikah v časih prihoda so po nekaterih informacijah shranjeni v značkah *event1_* in *event2_*, vendar njihov točen pomen ni nikjer razložen. DODATEK A vsebuje primer .xcm datoteke z nekoliko podrobnejšim opisom vsebine.

Poudariti je treba, da lahko s programom *Location Engine Config* zajemamo dogodke tudi v primeru, ko sistem Ubisense ni kalibriran. V tem primeru so obdelani podatki, ki so del dogodka, sicer brezpredmetni; enako najverjetneje velja za razlike v časih prihoda, ki so vezane na dolžine kablov, ki jih je potrebno kalibrirati. Kljub temu pa lahko iz dogodkov razberemo informacijo o kotih prihoda, ki so izraženi v koordinatnih sistemih posameznih senzorjev.

Zato lahko zaključimo, da so nam od surovih podatkov v praksi dosegljive samo meritve kotov prihoda.

2.7 Alternativna metoda kalibracije

Postopek kalibracije sistema Ubisense je precej zamuden, zlasti zato, ker zahteva natančne meritve položaja senzorjev in značk. Primeren je sicer za stalne (fiksne) postavitve sistema, kjer kalibracijo izvedemo samo enkrat (ob namestitvi), kar pa precej omejuje možnosti uporabe sistema.

Zamuden postopek kalibracije, ki zahteva poznavanje položajev senzorjev in značk, je značilen za večino komercialno dostopnih UWB sistemov za lociranje. Edino vidnejše delo s področja iskanja alternativnih postopkov kalibracije je članek, v katerem avtorji Hol, Schön in Gustafsson [13] predlagajo algoritem za hitro in enostavno kalibracijo UWB sistemov na osnovi meritev časov prihoda. Avtorji so predlagani algoritem, ki ne potrebuje zamudnega merjenja položaja senzorjev in značk, uspešno preizkusili na UWB sistemu podjetja TimeDomain [2].

Pri poskusu uporabe predlaganega algoritma za kalibracijo sistema Ubisense naletimo na nekaj ovir, ki izhajajo predvsem iz razlik v delovanju sistemov

TimeDomain in Ubisense; za razliko od sistema Ubisense, ki za lociranje uporablja kote prihoda in razlike v časih prihoda, sistem TimeDomain uporablja čase prihoda. To pomeni:

1. Ker sistem TimeDomain ne meri kotov prihoda, antene njegovih senzorjev niso usmerjene. Posledično predlagani alternativni algoritem ne ugotavlja orientacije senzorjev, ampak samo njihov položaj in razlike v sistemskih urah (*angl. clock offset*).

2. Algoritem temelji na meritvah časov prihodov, ki v sistemu Ubisense niso na voljo. Algoritem je sicer možno prirediti tudi za uporabo razlik v časih prihodov, vendar tudi ti podatki v praksi niso na voljo (glej Podpoglavje 2.6).

3. Parameter razlike v sistemskih urah senzorjev, ki ga določi algoritem, je sicer povezan s časovno sinhronizacijo, vendar njegova zveza z dolžinami (časovnimi zamiki) kablov v primeru sistema Ubisense ni povsem očitna.

Cilj diplomske naloge je zato poiskati alternativen postopek kalibracije sistema Ubisense, katerega rezultat bi bili parametri senzorjev v obliki, ki bi omogočala direkten vnos v program *Location Engine Config* (bodisi kot vnos vrednosti v polja dialoga s parametri senzorjev bodisi v obliki .xsc datoteke, ki opisuje celico). Postopek bi se izognil meritvam položajev senzorjev, obenem pa bi od surovih podatkov potreboval samo kote prihoda, ki jih lahko dobimo z izvozom dogodkov v .xcm datoteko.

Poglavje 3

Metode za kalibracijo kamer

Medtem ko moramo pri vgrajenem postopku kalibracije sistema Ubisense izmeriti položaje senzorjev, metode za kalibracijo kamer, razvite na področju strojnega vida, običajno ne zahtevajo poznavanja položajev kamer, temveč te skupaj z orientacijami kamer ocenijo tekom postopka kalibracije.

Za kalibracijo kamer običajno potrebujemo nekaj kontrolnih točk (oziroma kalibracijski vzorec ali kalibracijski predmet), katerih koordinate v prostoru izmerimo, koordinate v slikovni ravnini pa določimo s posnetkov. V zadnjem času se uveljavljajo tudi metode avtokalibracije, ki za samo kalibracijo načeloma ne potrebujejo poznavanja koordinat točk v prostoru, temveč samo njihove koordinate v slikovni ravnini.

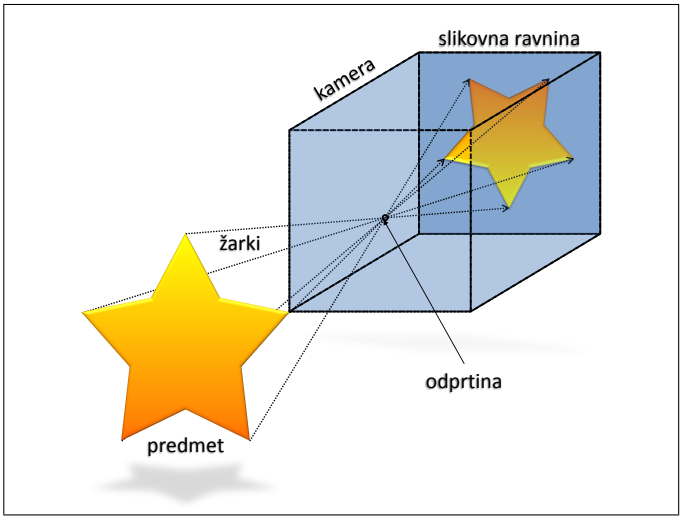
V nadaljevanju si bomo ogledali tri metode za kalibracijo kamer: metodo direktne linearne transformacije [20], Zhangovo metodo [34] ter avtokalibracijsko metodo Tomáša Svobode [26]. V Podpoglavju 3.1 bomo najprej predstavili centralno projekcijski model kamere skupaj s parametri, ki jih določimo tekom postopka kalibracije. Sledijo posamezne metode: metoda direktne linearne transformacije (DLT) v Podpoglavju 3.2, Zhangova metoda v Podpoglavju 3.3 in metoda Tomáša Svobode v Podpoglavju 3.4.

Skupaj z metodo DLT bomo predstavili tudi postopek rekonstrukcije ter koncept napak reprojekcije in rekonstrukcije. Metoda DLT in Zhangova metoda sta predstavljeni nekoliko podrobneje, medtem ko matematično ozadje metode Tomáša Svobode presega področje diplomske naloge in bomo zato predstavili samo njeno bistvo. Poleg samih metod sta predstavljeni tudi dve implementaciji v programskem okolju Matlab, ki ju bomo uporabili v naslednjem poglavju.

3.1 Model kamere

Če želimo ugotavljati položaj predmetov v prostoru s pomočjo njihovih slik, moramo poznati preslikavo med koordinatnim sistemom slike in zunanjim koordinatnim sistemom oziroma koordinatnim sistemom sveta, ki ga slika predstavlja; postopek ugotavljanja te preslikave imenujemo kalibracija.

Ker je preslikava med zunanjim koordinatnim sistemom in slikovno ravnino lastnost kamere, običajno govorimo o kalibraciji kamere, preslikavo pa opišemo z modelom kamere, čigar parametre določimo v postopku kalibracije.

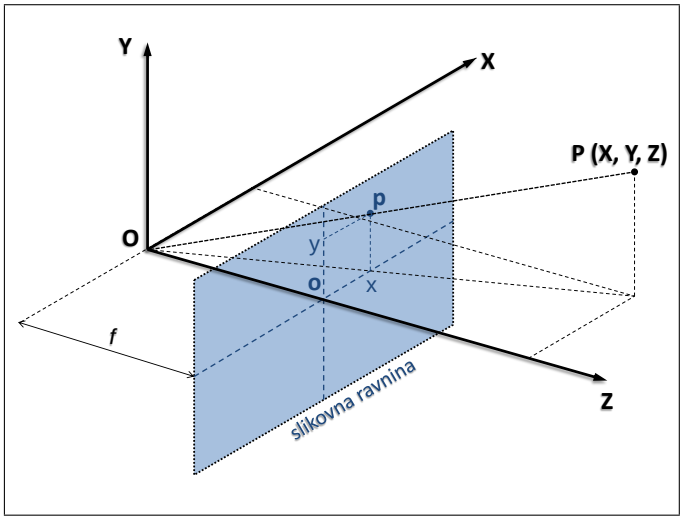


Slika 3.1: *Nastanek slike po centralno projekcijskem modelu*

Najbolj razširjen model kamere je t. i. centralno projekcijski model (*angl. pinhole camera*). Nastanek slike po tem modelu prikazuje Slika 3.1: slika objekta nastane tako, da žarki z objekta vstopajo v kamero skozi odprtino in v slikovni ravnini, ki se nahaja za odprtino, ustvarijo obrnjeno sliko. Če bi bila odprtina, skozi katero vstopajo žarki, neskončno majhna, bi skozi vsako točko v slikovni ravnini potekal natanko en žarek. V praksi je odprtina končne velikosti in skozi vsako točko v slikovni ravnini poteka snop žarkov, kar povzroča zamegljenost slike. Zaradi tega v odprtino vstavimo lečo, ki z zbiranjem žarkov zmanjša efektivno velikost odprtine. Oddaljenost slikovne ravnine od leče oziroma t. i. goriščna razdalja določa širino vidnega polja kamere.

Pri teoretični obravnavi centralno projekcijskega modela si pogosto pomagamo z navidezno slikovno ravnino; to dobimo tako, da dejansko slikovno ravnino

prestavimo med opazovani objekt in lečo kamere, pri čemer ohranimo goriščno razdaljo.



Slika 3.2: *Centralno projekcijski model kamere*

Preslikava med zunanjim koordinatnim sistemom in slikovno ravnino je pravzaprav sestavljena iz dveh ločenih preslikav; iz preslikave iz zunanjega koordinatnega sistema v koordinatni sistem kamere, ki jo opišemo s t. i. zunanjimi (*angl. extrinsic*) parametri, in preslikave iz koordinatnega sistema kamere v slikovno ravnino; to opišemo s t. i. notranjimi (*angl. intrinsic*) parametri.

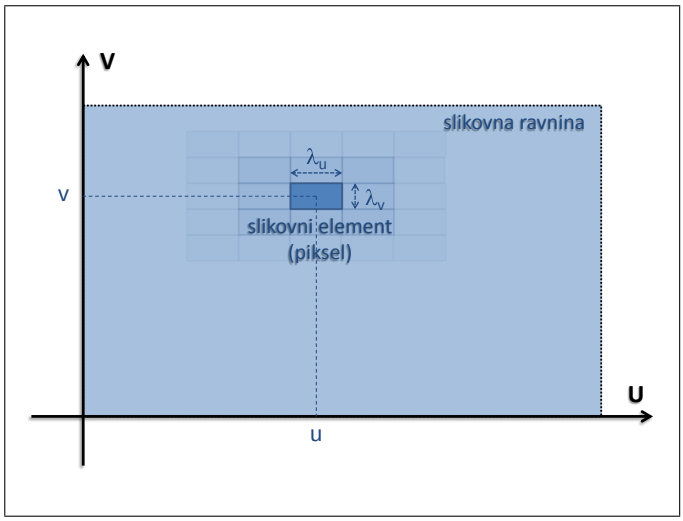
Običajno uporabimo model kamere z 11 parametri:

- Zunanji parametri:
  - položaj kamere (3)
  - orientacija kamere (3)
- Notranji parametri:
  - goriščna razdalja (1)
  - optično središče slike (2)
  - velikost slikovnega elementa (piksla) (2)

Položaj predmeta je običajno definiran v zunanjem (svetovnem) koordinatnem sistemu. S pomočjo zunanjih parametrov ga zasukamo in premaknemo v

koordinatni sistem kamere, nato pa s pomočjo notranjih parametrov preslikamo v slikovno ravnino. Goriščna razdalja f definira perspektivično projekcijo iz 3-D v 2-D prostor, ki jo glede na oznake na Sliki 3.2 zapišemo kot:

$$\begin{aligned} x &= f \cdot \frac{X}{Z} \\ y &= f \cdot \frac{Y}{Z} \end{aligned} \tag{3.1}$$



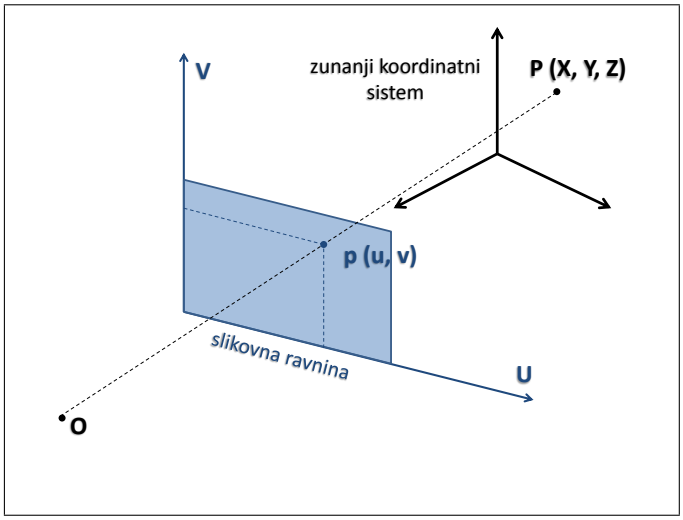
Slika 3.3: *Slikovna ravnina*

Kot je prikazano na Sliki 3.3, sta koordinati slikovne ravnine (u, v) diskretni in odvisni od velikosti slikovnih elementov (pikslov). Velikosti označimo z λ_u in λ_v, slikovni koordinati pa izrazimo v pikslih:

$$\begin{aligned} u &= \frac{x}{\lambda_u} \\ v &= \frac{y}{\lambda_v} \end{aligned} \tag{3.2}$$

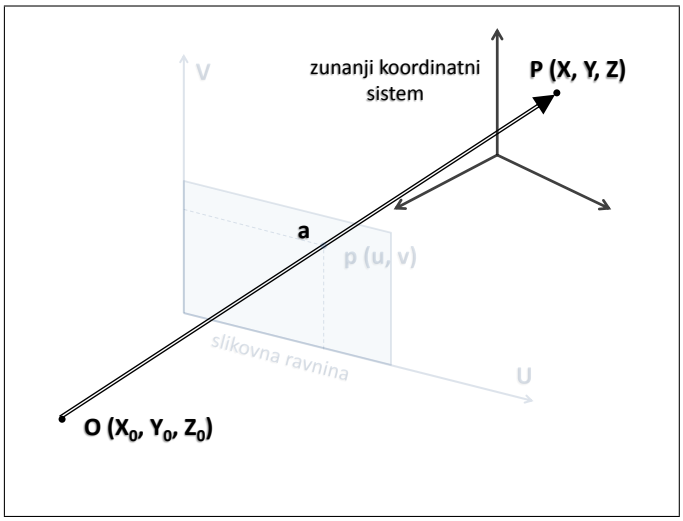
3.2 Direktna linearna transformacija — DLT

Direktna linearna transformacija (*angl. Direct Linear Transformation — DLT*) je šolski primer metode za kalibracijo modela kamere, s katero določimo notranje in zunanje parametre kamere na osnovi šestih kontrolnih točk, katerih koordinate v zunanjem koordinatnem sistemu poznamo. Kontrolne točke morajo biti nekoplanarne (ne smejo ležati v isti ravnini); pravimo, da morajo tvoriti t. i. kalibracijski volumen.



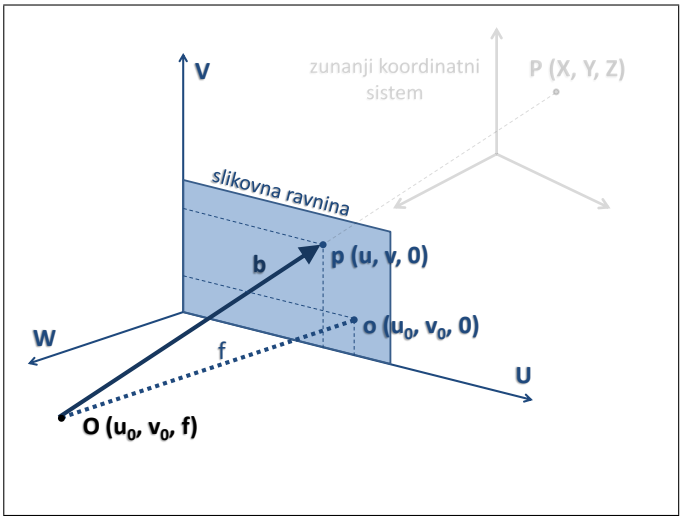
Slika 3.4: *Projekcija točke P v slikovno ravnino*

Model kamere opisuje projekcijo točke P v zunanjem (svetovnem) koordinatnem sistemu v točko p v slikovni ravnini, kot je prikazano na Sliki 3.4. Točko O imenujemo središče projekcije. Na Sliki 3.4 sta definirana dva koordinatna sistema — zunanji (svetovni) koordinatni sistem XYZ in koordinatni sistem slikovne ravnine (sistem UV). Koordinate točke P v zunanjem koordinatnem sistemu so tako (X, Y, Z), koordinati projekcije p v slikovni ravnini pa (u, v). Točke P, p in O so kolinearne; ta t. i. pogoj kolinearnosti je osnova za metodo DLT.



Slika 3.5: *Središče projekcije v zunanjem koordinatnem sistemu in vektor* $\boldsymbol{a}$

Predpostavimo, da so koordinate središča projekcije O v zunanjem koordinatnem sistemu (X_0, Y_0, Z_0); vektor $\boldsymbol{a}$ iz točke O v P lahko po komponentah zapišemo kot $(X - X_0, Y - Y_0, Z - Z_0)$ (Slika 3.5).



Slika 3.6: *Koordinatni sistem slikovne ravnine in vektor* $\boldsymbol{b}$

Koordinatnemu sistemu slikovne ravnine dodamo os W (Slika 3.6); ta tretja koordinata je za vse točke v slikovni ravnini enaka nič. Koordinate projekcije točke p v slikovni ravnini tako postanejo $(u, v, 0)$. Premico, ki je vzporedna osi W in gre skozi središče projekcije O, imenujemo optična os (*angl. principal axis*), točko o, v kateri optična os prebada slikovno ravnino, pa optično središče (*angl. principal point*). Razdaljo f med optičnim središčem o in središčem projekcije O imenujemo goriščna razdalja. Koordinate optičnega središča v koordinatnem sistemu slikovne ravnine označimo z $(u_0, v_0, 0)$; koordinate središča projekcije so potem (u_0, v_0, f), vektor $\boldsymbol{b}$ med O in o pa lahko zapišemo kot $(u - u_0, v - v_0, -f)$.

Točke O, p in P so kolinearne; za vektorski zapis kolinearnosti moramo vektor $\boldsymbol{a}$, ki je zapisan v zunanjem koordinatnem sistemu, zapisati v koordinatnem sistemu slikovne ravnine, v katerem je zapisan vektor $\boldsymbol{b}$. To storimo s pomočjo transformacijske matrike $\boldsymbol{R}$, ki predstavlja preslikavo iz zunanjega koordinatnega sistema v koordinatni sistem slikovne ravnine:

$$\boldsymbol{b} = c \cdot \boldsymbol{R} \cdot \boldsymbol{a} \tag{3.3}$$

3.2.1 Parametri DLT

Enačbo (3.3) zapišemo v matrični obliki:

$$\begin{bmatrix} U - U_0 \\ V - V_0 \\ -f \end{bmatrix} = c \cdot \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \cdot \begin{bmatrix} X - X_0 \\ Y - Y_0 \\ Z - Z_0 \end{bmatrix} \tag{3.4}$$

oziroma po komponentah:

$$\begin{aligned} U - U_0 &= c \cdot [r_{11} \cdot (X - X_0) + r_{12} \cdot (Y - Y_0) + r_{13} \cdot (Z - Z_0)] \\ V - V_0 &= c \cdot [r_{21} \cdot (X - X_0) + r_{22} \cdot (Y - Y_0) + r_{23} \cdot (Z - Z_0)] \\ -f &= c \cdot [r_{31} \cdot (X - X_0) + r_{32} \cdot (Y - Y_0) + r_{33} \cdot (Z - Z_0)] \end{aligned} \tag{3.5}$$

Iz zadnje enačbe izrazimo c:

$$c = \frac{-f}{r_{31} \cdot (X - X_0) + r_{32} \cdot (Y - Y_0) + r_{33} \cdot (Z - Z_0)} \tag{3.6}$$

Izraz za c vstavimo v prvi dve enačbi:

$$\begin{aligned} U - U_0 &= -f \cdot \frac{r_{11} \cdot (X - X_0) + r_{12} \cdot (Y - Y_0) + r_{13} \cdot (Z - Z_0)}{r_{31} \cdot (X - X_0) + r_{32} \cdot (Y - Y_0) + r_{33} \cdot (Z - Z_0)} \\ V - V_0 &= -f \cdot \frac{r_{21} \cdot (X - X_0) + r_{22} \cdot (Y - Y_0) + r_{23} \cdot (Z - Z_0)}{r_{31} \cdot (X - X_0) + r_{32} \cdot (Y - Y_0) + r_{33} \cdot (Z - Z_0)} \end{aligned} \tag{3.7}$$

Sledi diskretizacija slikovnih koordinat; (u_0, v_0) sta koordinati optičnega središča, velikost piksla pa je določena s faktorjema skaliranja λ_u in λ_v:

$$\begin{aligned} U - U_0 &= \lambda_u \cdot (u - u_0) \\ V - V_0 &= \lambda_v \cdot (v - v_0) \end{aligned} \tag{3.8}$$

Tako dobimo enačbi za u in v:

$$\begin{aligned} u - u_0 &= -\frac{f}{\lambda_u} \cdot \frac{r_{11} \cdot (X - X_0) + r_{12} \cdot (Y - Y_0) + r_{13} \cdot (Z - Z_0)}{r_{31} \cdot (X - X_0) + r_{32} \cdot (Y - Y_0) + r_{33} \cdot (Z - Z_0)} \\ v - v_0 &= -\frac{f}{\lambda_v} \cdot \frac{r_{21} \cdot (X - X_0) + r_{22} \cdot (Y - Y_0) + r_{23} \cdot (Z - Z_0)}{r_{31} \cdot (X - X_0) + r_{32} \cdot (Y - Y_0) + r_{33} \cdot (Z - Z_0)} \end{aligned} \tag{3.9}$$

Zgornji enačbi zapišemo v standardni in preglednejši obliki:

$$\begin{aligned} u &= \frac{L_1 \cdot X + L_2 \cdot Y + L_3 \cdot Z + L_4}{L_9 \cdot X + L_{10} \cdot Y + L_{11} \cdot Z + 1} \\ v &= \frac{L_5 \cdot X + L_6 \cdot Y + L_7 \cdot Z + L_8}{L_9 \cdot X + L_{10} \cdot Y + L_{11} \cdot Z + 1} \end{aligned} \tag{3.10}$$

Koeficienti L_i $(i = 1, 2, ..., 11)$ so t. i. parametri DLT.

$$
\begin{aligned}
D &= -(X_0 r_{31} + Y_0 r_{32} + Z_0 r_{33}) \\
f_u &= \frac{f}{\lambda_u} \\
f_v &= \frac{f}{\lambda_v} \\
L_1 &= \frac{u_0 r_{31} - f_u r_{11}}{D} \\
L_2 &= \frac{u_0 r_{32} - f_u r_{12}}{D} \\
L_3 &= \frac{u_0 r_{33} - f_u r_{13}}{D} \\
L_4 &= \frac{(f_u r_{11} - u_0 r_{31}) \cdot X_0 + (f_u r_{12} - u_0 r_{32}) \cdot Y_0 + (f_u r_{13} - u_0 r_{33}) \cdot Z_0}{D} \\
L_5 &= \frac{v_0 r_{31} - f_v r_{21}}{D} \\
L_6 &= \frac{v_0 r_{32} - f_v r_{22}}{D} \\
L_7 &= \frac{v_0 r_{33} - f_v r_{23}}{D} \\
L_8 &= \frac{(f_v r_{21} - v_0 r_{31}) \cdot X_0 + (f_v r_{22} - v_0 r_{32}) \cdot Y_0 + (f_v r_{23} - v_0 r_{33}) \cdot Z_0}{D} \\
L_9 &= \frac{r_{31}}{D} \\
L_{10} &= \frac{r_{32}}{D} \\
L_{11} &= \frac{r_{33}}{D}
\end{aligned}
\tag{3.11}
$$

3.2.2 Distorzija zaradi leče

Če želimo kalibrirati tudi distorzijo zaradi leče, lahko enačbi (3.10) razširimo z uvedbo petih dodatnih parametrov, s katerimi opišemo distorzijo zaradi leče. V tem primeru za kalibracijo potrebujemo vsaj 8 kontrolnih točk.

Kalibracija distorzije zaradi leče s stališča teme diplomske naloge ni zanimiva, zato se v razširitev enačb, ki je opisana v [20], na tem mestu ne bomo spuščali.

3.2.3 Iskanje vrednosti parametrov DLT

Pri DLT kalibraciji iščemo neznane vrednosti parametrov DLT. Enačbi (3.10) preuredimo v naslednjo obliko:

$$\begin{aligned} L_1 \cdot X + L_2 \cdot Y + L_3 \cdot Z + L_4 - uX \cdot L_9 - uY \cdot L_{10} - uZ \cdot L_{11} = u \\ L_5 \cdot X + L_6 \cdot Y + L_7 \cdot Z + L_8 - vX \cdot L_9 - vY \cdot L_{10} - vZ \cdot L_{11} = v \end{aligned} \tag{3.12}$$

Zgornji enačbi lahko zapišemo v matrični obliki:

$$\begin{bmatrix} X & Y & Z & 1 & 0 & 0 & 0 & 0 & -uX & -uY & -uZ \\ 0 & 0 & 0 & 0 & X & Y & Z & 1 & -vX & -vY & -vZ \end{bmatrix} \cdot \begin{bmatrix} L_1 \\ L_2 \\ L_3 \\ L_4 \\ L_5 \\ L_6 \\ L_7 \\ L_8 \\ L_9 \\ L_{10} \\ L_{11} \end{bmatrix} = \begin{bmatrix} u \\ v \end{bmatrix} \tag{3.13}$$

Poznamo koordinate točke v prostoru (X, Y in Z) ter koordinati iste točke v slikovni ravnini (u in v), iščemo pa neznane parametre L_i. Vsaka točka prispeva dve enačbi, neznanih parametrov pa je enajst; potrebujemo torej vsaj šest kontrolnih oziroma kalibracijskih točk. V praksi jih vzamemo več, da dobimo

predoločen sistem:

$$\begin{bmatrix} X_1 & Y_1 & Z_1 & 1 & 0 & 0 & 0 & 0 & -u_1X_1 & -u_1Y_1 & -u_1Z_1 \\ 0 & 0 & 0 & 0 & X_1 & Y_1 & Z_1 & 1 & -v_1X_1 & -v_1Y_1 & -v_1Z_1 \\ X_2 & Y_2 & Z_2 & 2 & 0 & 0 & 0 & 0 & -u_2X_2 & -u_2Y_2 & -u_2Z_2 \\ 0 & 0 & 0 & 0 & X_2 & Y_2 & Z_2 & 2 & -v_2X_2 & -v_2Y_2 & -v_2Z_2 \\ . & . & . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . & . & . \\ X_n & Y_n & Z_n & n & 0 & 0 & 0 & 0 & -u_nX_n & -u_nY_n & -u_nZ_n \\ 0 & 0 & 0 & 0 & X_n & Y_n & Z_n & n & -v_nX_n & -v_nY_n & -v_nZ_n \end{bmatrix} \cdot \begin{bmatrix} L_1 \\ L_2 \\ L_3 \\ L_4 \\ L_5 \\ L_6 \\ L_7 \\ L_8 \\ L_9 \\ L_{10} \\ L_{11} \end{bmatrix} = \begin{bmatrix} u_1 \\ v1 \\ . \\ . \\ u_n \\ v_n \end{bmatrix} \tag{3.14}$$

Dobljeni predoločen sistem rešimo z metodo najmanjših kvadratov:

$$\begin{aligned} &\boldsymbol{A} \cdot \boldsymbol{L} = \boldsymbol{b} \\ &[\boldsymbol{A}^T \cdot \boldsymbol{A}] \cdot \boldsymbol{L} = \boldsymbol{A}^T \cdot \boldsymbol{b} \\ &[\boldsymbol{A}^T \cdot \boldsymbol{A}]^{-1} \cdot [\boldsymbol{A}^T \cdot \boldsymbol{A}] \cdot \boldsymbol{L} = [\boldsymbol{A}^T \cdot \boldsymbol{A}]^{-1} \cdot \boldsymbol{A}^T \cdot \boldsymbol{b} \\ &\boldsymbol{L} = [\boldsymbol{A}^T \cdot \boldsymbol{A}]^{-1} \cdot \boldsymbol{A}^T \cdot \boldsymbol{b} \end{aligned} \tag{3.15}$$

3.2.4 Določitev parametrov kamere

Z rešitvijo predoločenega sistema (3.14) dobimo vrednosti enajstih parametrov DLT (L_i), iz teh pa lahko določimo notranje in zunanje parametre kamere s pomočjo relacij (3.11) in nekaterih dodatnih omejitev. Na tem mestu bomo podali samo enačbe za izračun parametrov, njihove izpeljave pa so na voljo v [17] in [20].

Položaj kamere (X_0, Y_0, Z_0) izračunamo z naslednjo enačbo:

$$\begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \end{bmatrix} = \begin{bmatrix} L_1 & L_2 & L3 \\ L_5 & L_6 & L_7 \\ L_9 & L_{10} & L_{11} \end{bmatrix}^{-1} \cdot \begin{bmatrix} -L_4 \\ -L_8 \\ -1 \end{bmatrix} \tag{3.16}$$

Koordinati optičnega središča (u_0, v_0) določimo ob postavitvi zahteve, da je rotacijska matrika $\boldsymbol{R}$ ortogonalna. Potem lahko izpeljemo enačbe:

$$\begin{aligned} {L_9}^2 + {L_{10}}^2 + {L_{11}}^2 &= \frac{1}{D^2} \\ u_0 = D^2 \cdot (L_1 L_9 + L_2 L_{10} + L_3 L_{11}) &= \frac{L_1 L_9 + L_2 L_{10} + L_3 L_{11}}{{L_9}^2 + {L_{10}}^2 + {L_{11}}^2} \\ v_0 = D^2 \cdot (L_5 L_9 + L_6 L_{10} + L_7 L_{11}) &= \frac{L_5 L_9 + L_6 L_{10} + L_7 L_{11}}{{L_9}^2 + {L_{10}}^2 + {L_{11}}^2} \end{aligned} \tag{3.17}$$

Enačbi za koeficienta f_u in f_v, ki ju potrebujemo za izračun matrike $\boldsymbol{R}$:

$$\begin{aligned} {f_u}^2 &= D^2 \cdot [(u_0 L_9 - L_1)^2 + (u_0 L_{10} - L_2)^2 + (u_0 L_{11} - L_3)^2] \\ {f_v}^2 &= D^2 \cdot [(v_0 L_9 - L_5)^2 + (v_0 L_{10} - L_6)^2 + (v_0 L_{11} - L_7)^2] \end{aligned} \tag{3.18}$$

Izračun rotacijske matrike $\boldsymbol{R}$:

$$\boldsymbol{R} = D \cdot \begin{bmatrix} \frac{u_0 L_9 - L1}{f_u} & \frac{u_0 L_{10} - L_2}{f_u} & \frac{u_0 L_{11} - L_3}{f_u} \\ \frac{v_0 L_9 - L_5}{f_v} & \frac{v_0 L_{10} - L_6}{f_v} & \frac{v_0 L_{11} - L_7}{f_v} \\ L_9 & L_{10} & L_{11} \end{bmatrix} \tag{3.19}$$

3.2.5 Rekonstrukcija s pomočjo DLT

Rekonstrukcija je obrnjen problem kalibracije; poznamo DLT parametre modela kamere (L_i) in slikovni koordinati točke (u, v), iščemo pa koordinate točke v prostoru (X, Y, Z).

Zopet vzamemo enačbi (3.10); tokrat ju preuredimo v obliko:

$$\begin{aligned} (L_1 - uL_9) \cdot X + (L_2 - uL_{10}) \cdot Y + (L_3 - uL_{11}) \cdot Z &= L_4 - u \\ (L_5 - vL_9) \cdot X + (L_6 - vL_{10}) \cdot Y + (L_7 - vL_{11}) \cdot Z &= L_8 - v \end{aligned} \tag{3.20}$$

V matrični obliki:

$$\begin{bmatrix} L_1 - uL_9 & L_2 - uL_{10} & L_3 - uL_{11} \\ L_5 - vL_9 & L_6 - vL_{10} & L_7 - vL_{11} \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} L_4 - u \\ L_8 - v \end{bmatrix} \tag{3.21}$$

Vsaka točka v sliki (u, v) prispeva dve enačbi, vendar prinese tri neznanke (X, Y, Z); zato vzamemo več slik iste točke v prostoru, in sicer z različnih (različno

postavljenih in obrnjenih) kamer v prostoru. S tem dobimo predoločen sistem:

$$\begin{bmatrix} (L_1 - uL_9)_1 & (L_2 - uL_{10})_1 & (L_3 - uL_{11})_1 \\ (L_5 - vL_9)_1 & (L_6 - vL_{10})_1 & (L_7 - vL_{11})_1 \\ . & . & . \\ . & . & . \\ (L_m - uL_9)_m & (L_2 - uL_{10})_m & (L_3 - uL_{11})_m \\ (L_5 - vL_9)_m & (L_6 - vL_{10})_m & (L_7 - vL_{11})_m \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} (L_4 - u)_1 \\ (L_8 - v)_1 \\ . \\ . \\ (L_4 - u)_m \\ (L_8 - v)_m \end{bmatrix} \tag{3.22}$$

Predoločen sistem (3.22) zopet rešimo z metodo najmanjših kvadratov in tako dobimo koordinate točke (X, Y, Z).

3.2.6 Napaka reprojekcije

Točnost kalibracije kamere lahko ocenimo z izračunom napake reprojekcije; kontrolne točke s pomočjo izračunanih parametrov DLT projeciramo v slikovno ravnino in izračunamo evklidsko razdaljo med izmerjenimi in reprojeciranimi točkami v slikovni ravnini:

$$\begin{aligned} e_{u_i} &= u_i - \frac{L_1 X_i + L_2 Y_i + L_3 Z_i + L_4}{L_9 Xi + L_{10} Y_i + L_{11} Z_i + 1} \\ e_{v_i} &= v_i - \frac{L_5 X_i + L_6 Y_i + L_7 Z_i + L_8}{L_9 Xi + L_{10} Y_i + L_{11} Z_i + 1} \\ \varepsilon_{reproj} &= \frac{1}{n} \sum_{i=0}^{n} \sqrt{{e_{u_i}}^2 + {e_{v_i}}^2} \end{aligned} \tag{3.23}$$

3.2.7 Napaka rekonstrukcije

Pogosto nas zanima tudi napaka rekonstrukcije, ki predstavlja odstopanje rekonstruiranih točk $(X_{r_i}, Y_{r_i}, Z_{r_i})$ od izmerjenih točk (X_i, Y_i, Z_i) v zunanjem koordinatnem sistemu. Kot mero za razdaljo uporabimo evklidsko razdaljo:

$$\varepsilon_{rekon} = \frac{1}{n} \sum_{i=0}^{n} \sqrt{(X_{r_i} - X_i)^2 + (Y_{r_i} - Y_i)^2 + (Z_{r_i} - Z_i)^2} \tag{3.24}$$

3.2.8 Modificirana metoda DLT

Čeprav v enačbah DLT (3.10) nastopa enajst parametrov, je dejansko število neodvisnih neznank deset — X_0, Y_0 ter Z_0, u_0 in v_0, f_u in f_v ter trije Eulerjevi

koti. Goriščna razdalja f in faktorja skaliranja λ_u ter λ_v so namreč soodvisni in se združijo v dva neodvisna parametra, f_u in f_v. Eden od DLT parametrov je torej odveč.

Pri standardni metodi DLT rešujemo sistem 3.14 z metodo najmanjših kvadratov, kot da bi imeli opravka z enajstimi neodvisnimi parametri. Ker ne upoštevamo, da je eden od parametrov odvisen od ostalih, transformacija med zunanjim koordinatnim sistemom in slikovno ravnino izgubi lastnost ortogonalnosti; do tega pride predvsem zaradi šuma v meritvah.

Zato večina kalibracijskih orodij, ki uporabljajo metodo DLT, namesto standardne metode uporablja iterativni postopek, ki ga je predlagal Hatze [12]. Pri tej modificirani metodi DLT izrazimo en parameter L_i v odvisnosti od ostalih, s čimer dobimo nelinearno omejitev.

Z uporabo enačb (3.10) in (3.20) lahko pokažemo, da velja enakost:

$$(DL_1) \cdot (DL_5) + (DL_2) \cdot (DL_6) + (DL_3) \cdot (DL_7) = u_0 \cdot v_0 \tag{3.25}$$

Z vstavljanjem izrazov za parametre (3.11) v enačbo (3.25) dobimo nelinearno odvisnost med parametri:

$$\begin{aligned} &(L_1L_5 + L_2L_6 + L_3L_7) \cdot ({L_9}^2 + {L_{10}}^2 + {L_{11}}^2) = \\ &= (L_1L_9 + L_2L_{10} + L_3L_{11}) \cdot (L_5L_9 + L_6L_{10} + L_7L_{11}) \end{aligned} \tag{3.26}$$

Postopek v Algoritmu 1 izvedemo za vsako od kamer, ki jih želimo kalibrirati.

Algoritem 1 Postopek modificirane metode DLT

1: Posnamemo vsaj 6 kontrolnih točk in izmerimo njihove koordinate v zunanjem koordinatnem sistemu.
2: S standardno metodo DLT izračunamo vrednosti 11 DLT parametrov.
3: Iz enačb za DLT izločimo en parameter (npr. L_1) tako, da vstavimo vrednost, ki smo jo zanj dobili v prvi iteraciji. Rešimo reducirani sistem 10 parametrov. Vrednost odstranjenega parametra izračunamo iz 10 pravkar izračunanih parametrov in enačbe (3.26).
4: Ponavljamo korak 3, dokler ne dobimo stabilne rešitve.

3.2.9 Pomanjkljivosti metode

Osnovna metoda DLT je sicer enostavna, vendar ima to slabost, da v prisotnosti šuma v meritvah transformacijska matrika, ki opisuje premik in

zasuk iz zunanjega koordinatnega sistema v koordinatni sistem kamere, izgubi lastnost ortogonalnosti, kar povzroči napako v ocenjenih zunanjih parametrih kamere. Zato raje uporabimo iterativno modificirano metodo DLT, s katero to pomanjkljivost odpravimo.

S stališča preprostosti za izvedbo (v smislu izvedenih meritev) je pomanjkljivost metode tudi zahteva po poznavanju koordinat kontrolnih točk v prostoru. Za izvedbo kalibracije načeloma zadostuje že šest kontrolnih točk, vendar v praksi poskušamo doseči čim boljšo pokritost prostora. Posebej neugodna je zahteva po nekoplanarnosti kontrolnih točk, zaradi katere moramo slednje razporediti v vse smeri po prostoru, kar nam oteži meritve.

3.3 Methoda Zhengyou Zhanga

Kalibracijska metoda, ki jo je predlagal Zhengyou Zhang [34], je precej podobna metodi DLT. Tudi v tem primeru določimo notranje in zunanje parametre kamere s pomočjo kontrolnih točk, katerih koordinate v prostoru poznamo. Medtem ko morajo pri metodi DLT kontrolne točke tvoriti kalibracijski volumen (ne smejo ležati v isti ravnini), točke pri Zhangovi metodi tvorijo kalibracijsko ravnino.

Metoda je pravzaprav zasnovana tako, da kalibriramo s kalibracijskim vzorcem, natisnjenim na ravno površino — primer kalibracijskega vzorca je šahovnica. Za kalibracijo sta potrebna vsaj dva posnetka kalibracijskega vzorca, napravljena z različnih zornih kotov. Premikamo lahko bodisi vzorec bodisi kamero, lahko pa vzorec posnamemo z več kamerami, če gre za kalibracijo sistema več kamer.

Podobno kot pri metodi DLT moramo tudi pri Zhangovi metodi poznati koordinate točk na kalibracijskem vzorcu. Če je kalibracijski vzorec šahovnica, zadošča poznavanje razmika med točkami, koordinat enega od vogalov ter orientacije šahovnice; najlažje je, če šahovnico postavimo v izhodišče koordinatnega sistema in jo poravnamo z osmi.

3.3.1 Osnovne enačbe

Kot je prikazano na Sliki 3.4, imamo opravka s projekcijo točke $P = (X, Y, Z)^T$ iz zunanjega koordinatnega sistema v točko $p = (u, v)^T$ v slikovni ravnini. Uvedemo

homogene koordinate; oba vektorja koordinat razširimo tako, da jima na zadnjem mestu dodamo element z vrednostjo 1: $\tilde{P} = (X, Y, Z, 1)^T$ in $\tilde{p} = (u, v, 1)^T$.

Uporabimo centralno projekcijski model; projekcijo opišemo z enačbo:

$$\begin{aligned} s \cdot \tilde{\boldsymbol{p}} &= \boldsymbol{A} \cdot \begin{bmatrix} \boldsymbol{R} & \boldsymbol{t} \end{bmatrix} \cdot \tilde{\boldsymbol{P}} \\ \boldsymbol{A} &= \begin{bmatrix} \alpha & c & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix} \end{aligned} \tag{3.27}$$

Faktor s je poljuben faktor skaliranja. $\boldsymbol{R}$ in $\boldsymbol{t}$ predstavljata zasuk in premik iz zunanjega koordinatnega sistema v koordinatni sistem kamere — gre torej za zunanje parametre kamere. Matriko $\boldsymbol{A}$ imenujemo matrika notranjih parametrov kamere; (u_0, v_0) sta koordinati optičnega središča, α in β sta faktorja skaliranja v smeri osi U in V, faktor poševnosti (*angl. skew*) c pa je povezan s kotom med njima (kot med osema je običajno 90° in c je enak nič).

Brez izgube na splošnosti lahko predpostavimo, da kalibracijski vzorec leži v ravnini $Z = 0$. Če označimo i-ti stolpec rotacijske matrike $\boldsymbol{R}$ z $\boldsymbol{r_i}$, lahko enačbo (3.27) zapišemo kot:

$$s \cdot \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \boldsymbol{A} \cdot \begin{bmatrix} \boldsymbol{r_1} & \boldsymbol{r_2} & \boldsymbol{r_3} & \boldsymbol{t} \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ 0 \\ 1 \end{bmatrix} = \boldsymbol{A} \cdot \begin{bmatrix} \boldsymbol{r_1} & \boldsymbol{r_2} & \boldsymbol{t} \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} \tag{3.28}$$

Ker je Z koordinata vedno enaka nič, bomo njen zapis v nadaljevanju izpuščali; $P = (X, Y)^T$ in $\tilde{P} = (X, Y, 1)^T$. Točka kalibracijskega vzorca P in njena slika p sta povezani prek homografije $\boldsymbol{H}$:

$$\begin{aligned} s \cdot \tilde{\boldsymbol{p}} &= \boldsymbol{H} \cdot \tilde{\boldsymbol{P}} \\ \boldsymbol{H} &= \boldsymbol{A} \cdot \begin{bmatrix} \boldsymbol{r_1} & \boldsymbol{r_2} & \boldsymbol{t} \end{bmatrix} \end{aligned} \tag{3.29}$$

Kot vidimo, je 3×3 matrika homografije $\boldsymbol{H}$ določena do faktorja skaliranja. Matriko $\boldsymbol{H}$ lahko ocenimo na različne načine; eden od možnih pristopov je predstavljen v Zhangovem članku [34] (Dodatek A) in temelji na uporabi kriterija

največjega verjetja (*angl. maximum likelihood*). Ob predpostavki, da so slike točk kalibracijskega vzorca pošumljene z Gaussovim šumom, se ocena homografije reducira v nelinearen problem najmanjših kvadratov, ki ga rešimo z Levenberg-Marquardtovim algoritmom [29]. Pred iskanjem začetnih vrednosti podatke normaliziramo, da dosežemo boljšo numerično pogojenost.

Ko ocenimo homografijo $\boldsymbol{H} = \begin{bmatrix} \boldsymbol{h_1} & \boldsymbol{h_2} & \boldsymbol{h_3} \end{bmatrix}$, lahko ponovno zapišemo enačbo (3.29):

$$\begin{bmatrix} \boldsymbol{h_1} & \boldsymbol{h_2} & \boldsymbol{h_3} \end{bmatrix} = \lambda \cdot \boldsymbol{A} \cdot \begin{bmatrix} \boldsymbol{r_1} & \boldsymbol{r_2} & \boldsymbol{t} \end{bmatrix} \tag{3.30}$$

Faktor λ je poljuben faktor skaliranja; ker vemo, da sta $\boldsymbol{r_1}$ in $\boldsymbol{r_2}$ ortonormalna, lahko zapišemo:

$$\begin{aligned} &\boldsymbol{h_1}^T \cdot \boldsymbol{A}^{-T} \cdot \boldsymbol{A}^{-1} \cdot \boldsymbol{h_2} = 0 \\ &\boldsymbol{h_1}^T \cdot \boldsymbol{A}^{-T} \cdot \boldsymbol{A}^{-1} \cdot \boldsymbol{h_1} = \boldsymbol{h_2}^T \cdot \boldsymbol{A}^{-T} \cdot \boldsymbol{A}^{T} \cdot \boldsymbol{h_2} \end{aligned} \tag{3.31}$$

Enačbi (3.31) predstavljata osnovni omejitvi za notranje parametre pri poznavanju ene homografije; ker ima homografija 8 prostostnih stopenj in ima kamera 6 zunanjih parametrov (3 za zasuk in 3 za premik), lahko iz ene homografije dobimo samo dve omejitvi za notranje parametre.

3.3.2 Kalibracija kamere

Za rešitev sistema (3.31) označimo:

$$\begin{aligned} \boldsymbol{B} = \boldsymbol{A}^{-T} \cdot \boldsymbol{A}^{-1} &= \begin{bmatrix} B_{11} & B_{12} & B_{13} \\ B_{21} & B_{22} & B_{23} \\ B_{31} & B_{32} & B_{33} \end{bmatrix} = \\ &= \begin{bmatrix} \frac{1}{\alpha^2} & -\frac{c}{\alpha^2\beta} & \frac{cv_0 - u_0\beta}{\alpha^2\beta} \\ -\frac{c}{\alpha^2\beta} & \frac{c^2}{\alpha^2\beta^2} + \frac{1}{\beta^2} & -\frac{c(cv_0 - u_0\beta)}{\alpha^2\beta^2} - \frac{v_0}{\beta^2} \\ \frac{cv_0 - u_0\beta}{\alpha^2\beta} & -\frac{c(cv_0 - u_0\beta)}{\alpha^2\beta^2} - \frac{v_0}{\beta^2} & \frac{(cv_0 - u_0\beta)^2}{\alpha^2\beta^2} + \frac{{v_0}^2}{\beta^2} + 1 \end{bmatrix} \end{aligned} \tag{3.32}$$

Matrika $\boldsymbol{B}$ je simetrična; zato je povsem določena s 6-D vektorjem:

$$\boldsymbol{b} = \begin{bmatrix} B_{11} & B_{12} & B_{22} & B_{13} & B_{23} & B_{33} \end{bmatrix}^T \tag{3.33}$$

Če označimo vektor i-tega stolpca matrike $\boldsymbol{H}$ s $\boldsymbol{h_i} = \begin{bmatrix} h_{i1}, h_{i2}, h_{i3} \end{bmatrix}^T$, velja:

$$\begin{aligned} \boldsymbol{h_i}^T \cdot \boldsymbol{B} \cdot \boldsymbol{h_j} &= \boldsymbol{v_{ij}}^T \cdot \boldsymbol{b} \\ \boldsymbol{v_{ij}} &= \begin{bmatrix} h_{i1}h_{j1} \\ h_{i1}h_{j2} + h_{i2}h_{j1} \\ h_{i2}h_{j2} \\ h_{i3}h_{j1} + h_{i1}h_{j3} \\ h_{i3}h_{j2} + h_{i2}h_{j3} \\ h_{i3}h_{j3} \end{bmatrix} \end{aligned} \tag{3.34}$$

Enačbi omejitev iz homografije (3.31) lahko zapišemo kot homogeni enačbi v odvisnosti od $\boldsymbol{b}$:

$$\begin{bmatrix} \boldsymbol{v_{12}}^T \\ (\boldsymbol{v_{11}} - \boldsymbol{v_{22}})^T \end{bmatrix} \cdot \boldsymbol{b} = 0 \tag{3.35}$$

Če imamo n slik kalibracijskega vzorca, dobimo sistem:

$$\boldsymbol{V} \cdot \boldsymbol{b} = 0 \tag{3.36}$$

Matrika $\boldsymbol{V}$ je dimenzij $2n \times 6$; če imamo $n \geq 3$ slik, lahko rešimo sistem (3.36) z metodo najmanjših kvadratov in dobimo vektor $\boldsymbol{b}$, določen do poljubnega faktorja skaliranja. V primeru, ko imamo samo dve sliki, lahko sistem rešimo z uvedbo zahteve, da je $c = 0$ (pravokotnost pikslov); v tem primeru sistemu (3.36) dodamo enačbo $\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \cdot \boldsymbol{b} = 0$.

Ko izračunamo vektor $\boldsymbol{b}$, lahko določimo matriko notranjih parametrov $\boldsymbol{A}$:

$$\begin{aligned} v_0 &= \frac{B_{12}B_{13} - B_{11}B_{23}}{B_{11}B_{22} - {B_{12}}^2} \\ \lambda &= B_{33} - \frac{{B_{13}}^2 + v_0(B_{12}B_{13} - B_{11}B_{23})}{B_{11}} \\ \alpha &= \sqrt{\frac{\lambda}{B_{11}}} \\ \beta &= \sqrt{\frac{\lambda B_{11}}{B_{11}B_{22} - {B_{12}}^2}} \\ c &= -\frac{B_{12}\alpha^2\beta}{\lambda} \\ u_0 &= \frac{cv_0}{\alpha} - \frac{B_{13}\alpha^2}{\lambda} \end{aligned} \tag{3.37}$$

Z določeno matriko $\boldsymbol{A}$ lahko določimo zunanje parametre kamere:

$$\begin{aligned} \lambda &= \frac{1}{\|\boldsymbol{A}^{-1} \cdot \boldsymbol{h_1}\|} = \frac{1}{\|\boldsymbol{A}^{-1} \cdot \boldsymbol{h_2}\|} \\ \boldsymbol{r_1} &= \lambda \cdot \boldsymbol{A}^{-1} \cdot \boldsymbol{h_1} \\ \boldsymbol{r_2} &= \lambda \cdot \boldsymbol{A}^{-1} \cdot \boldsymbol{h_2} \\ \boldsymbol{r_3} &= \boldsymbol{r_1} \times \boldsymbol{r_2} \\ \boldsymbol{t} &= \lambda \cdot \boldsymbol{A}^{-1} \cdot \boldsymbol{h_3} \end{aligned} \tag{3.38}$$

Zaradi šuma v podatkih tako izračunana matrika $\boldsymbol{R}$ v splošnem ni prava rotacijska matrika (ne zadošča vsem pogojem).

Dobljene parametre modela zato natančneje določimo s pomočjo ocene največjega verjetja. Imamo n slik kalibracijskega vzorca, ki ga tvori m točk; če predpostavimo, da so točke na slikah pošumljene z neodvisnim in identično porazdeljenim šumom (*angl. independent and identically distributed noise*), dobimo oceno po kriteriju največjega verjetja z minimizacijo naslednjega funkcionala:

$$\sum_{i=1}^{n}\sum_{j=1}^{m} \left\| \boldsymbol{p}_{ij} - \hat{\boldsymbol{p}}(\boldsymbol{A}, \boldsymbol{R_i}, \boldsymbol{t_i}, P_j) \right\|^2 \tag{3.39}$$

S $\hat{\boldsymbol{p}}(\boldsymbol{A}, \boldsymbol{R_i}, \boldsymbol{t_i}, P_j)$ označimo projekcijo točke P_j v sliki i, ki jo izračunamo po enačbi (3.27); gre torej za minimizacijo napake reprojekcije. Rotacijsko matriko $\boldsymbol{R}$

parametriziramo z vektorjem treh parametrov $\boldsymbol{r}$, ki ima smer osi zasuka, njegova dolžina pa ustreza kotu zasuka. Matriko $\boldsymbol{R}$ in vektor $\boldsymbol{r}$ povezuje Rodriguesova formula [8]. Minimizacija (3.39) predstavlja nelinearen minimizacijski problem, ki ga rešimo z Levenberg-Marquardtovim algoritmom [29].

Postopek kalibracije, ki je povzet v Algoritmu 2, lahko razširimo, tako da vključuje tudi oceno distorzije zaradi leče. Ker ta s stališča teme diplomske naloge ni zanimiva, se na tem mestu v razširitve enačb, ki so opisane v [34], ne bomo spuščali.

Algoritem 2 Zhangov postopek kalibracije kamere

1: Na ravno površino natisnemo kalibracijski vzorec. Posnamemo nekaj slik kalibracijskega vzorca pod različnimi koti tako, da premikamo bodisi kamero, bodisi vzorec; lahko pa posnetke napravimo z več enakimi kamerami, ki so postavljene na različnih mestih.
2: Notranje in zunanje parametre kamer določimo z rešitvijo sistema (3.36) z metodo najmanjših kvadratov ter enačbami (3.37) in (3.38).
3: V prejšnjem koraku dobljene vrednosti parametrov uporabimo kot začetne vrednosti pri nelinearni optimizaciji parametrov z algoritmom Levenberg-Marquardta, tekom katere minimiziramo napako reprojekcije (3.39).

Pri kalibraciji moramo paziti, da je kalibracijski vzorec na vsakem posnetku ne samo premaknjen, ampak tudi drugače obrnjen. V primeru, ko se kalibracijski vzorec na dveh slikah nahaja v vzporednih ravninah, homografija druge slike namreč ne prinese nobene dodatne informacije o omejitvah za notranje parametre in je s stališča kalibracije neuporabna.

3.3.3 Implementacija v programskem okolju Matlab

Zhangova metoda kalibracije kamer je skupaj z metodo DLT implementirana v orodju *Camera Calibration Toolbox for Matlab*, ki ga je napisal Jean-Yves Bouguet. Orodje je napisano v programskem okolju Matlab in je prosto dostopno na avtorjevi spletni strani [9].

Slika 3.7: *Grafični vmesnik orodja Camera Calibration Toolbox for Matlab*

Grafični vmesnik in kalibracija s pomočjo šahovnice

Orodje je v prvi vrsti namenjeno kalibraciji s pomočjo kalibracijskega vzorca — šahovnice, zato ima vgrajen pregleden grafični vmesnik (Slika 3.7), ki omogoča postopno izvedbo vseh korakov kalibracije:

- Nalaganje slik v pomnilnik.
- Iskanje vogalov kvadratkov na šahovnici; določitev koordinat točk v prostoru in v slikovni ravnini.
- Kalibracija z Zhangovo metodo in nelinearno optimizacijo (Levenberg-Marquardt).
- Prikaz reprojekcije točk na slikah in določitev napake reprojekcije.
- Grafičen prikaz zunanjih parametrov kamer.
- Izboljšava rezultatov s ponovitvijo določanja vogalov in ponovno kalibracijo.
- Vizualizacija distorzije zaradi leče.
- Izvoz izračunanih parametrov kamer.

Podrobnejši opis korakov in navodila za uporabo vmesnika so na voljo na avtorjevi spletni strani [9].

Kalibracija z vnosom numeričnih podatkov

S stališča teme diplomske naloge je pomembnejša kot zgoraj opisani grafični vmesnik možnost kalibracije z vnosom numeričnih podatkov, ki jo uporabimo, kadar smo postopek določanja koordinat kontrolnih točk v prostoru in slikovni ravnini opravili sami.

Postopek, ki ga zaženemo z zagonom skripte **go_calib_optim**, omogoča kalibracijo največ 6 kamer; število kamer podamo prek spremenljivke **n_ima**. Koordinate točk v prostoru podamo prek matrik **X_1** – **X_n**, kjer vsak stolpec matrike ustreza (X, Y, Z) koordinatam točke. Koordinate točk v slikovni ravnini podamo prek matrik **x_1** – **x_n**, kjer vsak stolpec matrike predstavlja koordinati točke (u, v) v slikovni ravnini.

Ker postopku kalibracije podamo tako koordinate v prostoru kot v slikovni ravnini, ni potrebno, da kontrolne točke ležijo v isti ravnini, niti, da tvorijo pravilen vzorec. V primeru koplanarnih točk postopek za določitev začetnih vrednosti parametrov uporabi Zhangovo metodo, v primeru nekoplanarnih točk pa metodo DLT. V obeh primerih sledi nelinearna optimizacija, tekom katere prek minimizacije napake reprojekcije dosežemo končne vrednosti parametrov. Postopek shrani vrednosti parametrov v naslednje spremenljivke:

- Zunanji parametri (za vsako kamero posebej):

  Položaj kamere: 3×1 vektor **Tc_1** – **Tc_n**; vektor predstavlja koordinate izhodišča zunanjega koordinatnega sistema v koordinatnem sistemu kamere

  Orientacija kamere: 3×3 rotacijska matrika **Rc_1** – **Rc_n**

- Notranji parametri (skupni za vse kamere):

  Goriščna razdalja (v pikslih): 2×1 vektor **fc**

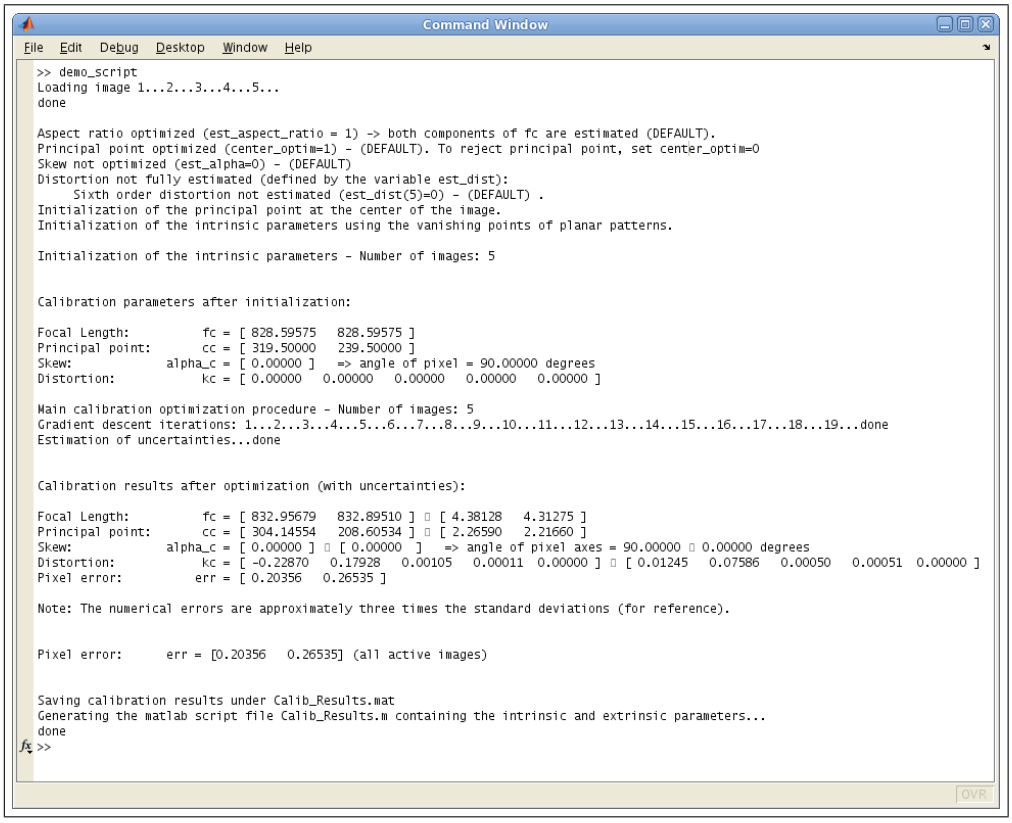
  Optično središče slike: 2×1 vektor **cc**

  Kot med osema slikovne ravnine: skalarna spremenljivka **alpha_c**

  Distorzija zaradi leče: 5×1 vektor **kc**

Postopku lahko podamo začetne vrednosti za notranje parametre, in sicer tako, da jih shranimo v ustrezne spremenljivke. Optimizacijo notranjih parametrov lahko izključimo tako, da spremenljivke **est_aspect_ratio**, **center_optim** in **est_alpha** oziroma ustrezna mesta v vektorjih **est_fc** (2×1 vektor) in **est_dist** (5×1 vektor) nastavimo na nič.

Po končani kalibraciji program izpiše vrednosti notranjih parametrov kamer, kot je prikazano na Sliki 3.8. Na Sliki 3.9 je prikazan primer rezultata kalibracije s Zhangovimi testnimi podatki: Slika 3.9 (a) prikazuje zunanje parametre kamer v



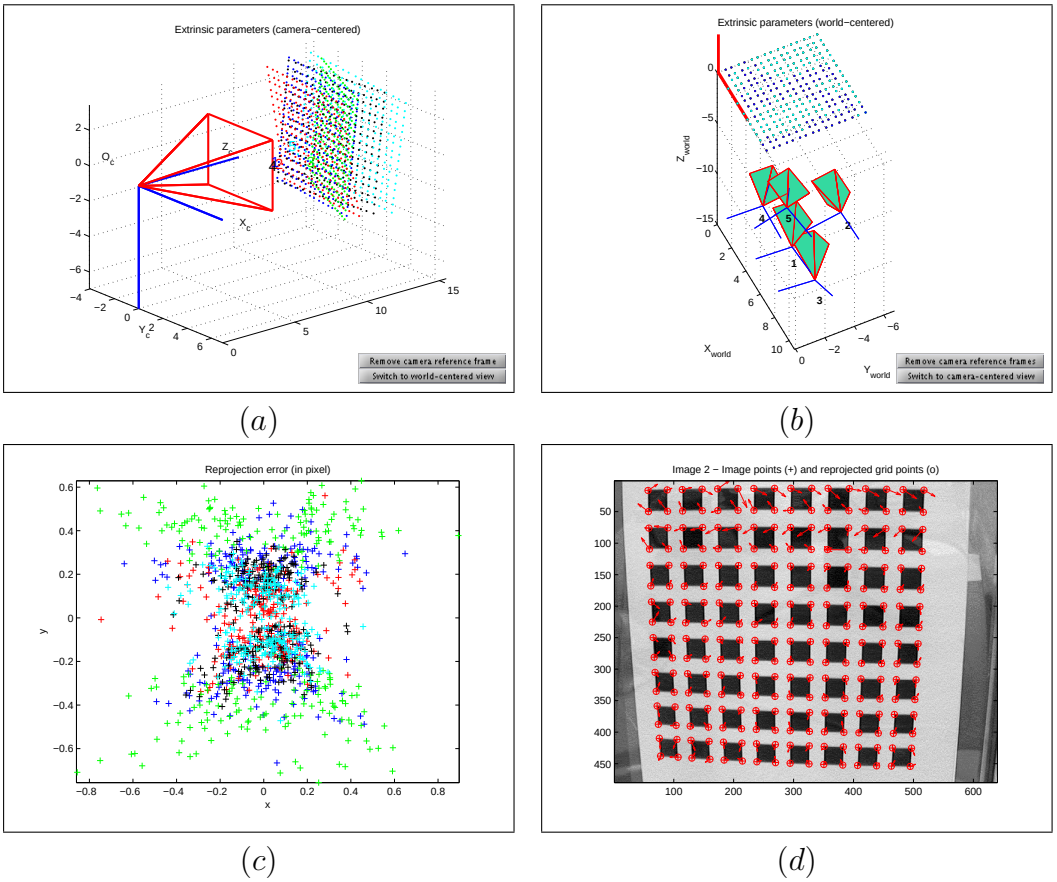
Slika 3.8: *Izpis notranjih parametrov kamer po končani kalibraciji z orodjem Camera Calibration Toolbox*

koordinatnem sistemu kamere, Slika 3.9 (b) pa v zunanjem koordinatnem sistemu. Na Sliki 3.9 (c) so prikazane vrednosti napake reprojekcije za vsako kontrolno točko na vseh petih slikah, Slika 3.9 (d) pa prikazuje reprojekcijo kontrolnih točk na eni od kalibracijskih slik.

3.3.4 Pomanjkljivosti metode

Glavna pomanjkljivost Zhangove metode s stališča preprostosti za izvedbo je, podobno kot pri metodi DLT, zahteva po poznavanju koordinat kontrolnih točk v prostoru. Kljub temu je Zhangova metoda ugodnejša, ker so kontrolne točke koplanarne, kar nam močno olajša meritve.

Za kalibracijo vseh notranjih in zunanjih parametrov ene same kamere sicer potrebujemo dva posnetka dveh različno obrnjenih ravnin kontrolnih točk. S tem celotna množica kontrolnih točk postane nekoplanarna in metoda postane



Slika 3.9: *Rezultat kalibracije z orodjem Camera Calibration Toolbox: zunanji parametri kamer v koordinatnem sistemu kamere (a), zunanji parametri kamer v zunanjem koordinatnem sistemu (b), vrednosti napake reprojekcije za vsako od slik (c) in reprojekcija kontrolnih točk na eni od kalibracijskih slik (d)*

ekvivalentna metodi DLT. Po drugi strani pa, če kalibriramo dve ali več kamer hkrati, to lahko storimo z eno samo ravnino kontrolnih točk, ki jo posnamemo z vsako od kamer — vendar morajo biti v tem primeru notranji parametri vseh kamer enaki.

3.4 Metoda Tomáša Svobode

Metodo Tomáša Svobode [26] uvrščamo med t. i. avtokalibracijske metode; gre za skupino v zadnjem času razvitih metod, pri katerih kalibracija poteka brez kalibracijskega objekta oziroma poznavanja koordinat kontrolnih točk v prostoru.

Metoda je v prvi vrsti namenjena samodejni kalibraciji velikega števila kamer, nameščenih v prostoru; primer so interaktivna okolja navidezne resničnosti. Kalibracija je povsem samodejna, od uporabnika zahteva le, da po prostoru premika svetel, lahko zaznaven, objekt — npr. laserski kazalec, na vrh katerega namestimo košček prozorne plastike, da razpršimo svetlobo v vse smeri.

Algoritem nato poišče svetlo točko na vseh posnetkih z vsake kamere in veljavnost s slik izluščenih točk preveri v parih s pomočjo omejitev epipolarne geometrije. Jedro metode je razcep t. i. skalirane matrike meritev na matriko kamer in matriko koordinat točk v prostoru; s t. i. postopkom evklidske stratifikacije in uvedbo geometrijskih omejitev dosežemo, da ti matriki postaneta evklidski. V matriki kamer so vsebovane matrike parametrov vseh kamer, zato je z njeno določitvijo postopek kalibracije končan. S tako določenimi parametri kamer in koordinatami točk v prostoru izračunamo napako reprojekcije in na osnovi lete izločimo odstopajoče točke; postopek ponavljamo, dokler napaka reprojekcije ne pade pod predpisano vrednost.

Po avtorjevih navedbah naj bi s postopkom, katerega implementacija v okolju Matlab je prosto dostopna na avtorjevi spletni strani [25], lahko dosegli napako reprojekcije, manjšo od 1/5 piksla, tudi pri kamerah z izrazito radialno distorzijo leče.

3.4.1 Teoretično jedro metode

Na tem mestu bomo podali samo teoretično jedro metode; podroben opis vseh korakov in postopkov presega področje diplomske naloge in je na voljo v [26].

Predpostavimo, da imamo m kamer in n točk v prostoru; $\boldsymbol{X}_j = (X_j, Y_j, Z_j)^T$, $j = 1, ..., n$. Uporabljamo centralno projekcijski model, zato lahko projekcijo 3-D točke $\boldsymbol{X}_j$ v 2-D točko v slikovni ravnini $\boldsymbol{u}_j^i$ zapišemo kot:

$$\lambda_j^i \cdot \begin{bmatrix} u_j^i \\ v_j^i \\ 1 \end{bmatrix} = \lambda_j^i \cdot \boldsymbol{u}_j^i = \boldsymbol{P}^i \cdot \boldsymbol{X}_j \tag{3.40}$$

Matrika $\boldsymbol{P}^i$ je matrika i-te kamere in vsebuje enajst parametrov kamere — šest zunanjih in pet notranjih. Faktor λ_j^i imenujemo tudi projekcijska globina

(*angl. projective depth*). Končni cilj kalibracije je ocena faktorjev λ_j^i in matrik kamer $\boldsymbol{P}^i$.

Vse vektorje točk in matrike kamer lahko združimo v eno matriko:

$$\begin{aligned}
\boldsymbol{W_s} &= \begin{bmatrix} \lambda_1^1 \cdot \begin{bmatrix} u_1^1 \\ v_1^1 \\ 1 \end{bmatrix} & \dots & \lambda_n^1 \cdot \begin{bmatrix} u_n^1 \\ v_n^1 \\ 1 \end{bmatrix} \\ \vdots & \ddots & \vdots \\ \lambda_1^m \cdot \begin{bmatrix} u_1^m \\ v_1^m \\ 1 \end{bmatrix} & \dots & \lambda_n^m \cdot \begin{bmatrix} u_n^m \\ v_n^m \\ 1 \end{bmatrix} \end{bmatrix} = \begin{bmatrix} \boldsymbol{P}^1 \\ \vdots \\ \boldsymbol{P}^m \end{bmatrix}_{3m\times 4} \cdot \begin{bmatrix} \boldsymbol{X}_1 & \dots & \boldsymbol{X}_n \end{bmatrix}_{4\times n} \\
\boldsymbol{W_s} &= \boldsymbol{P} \cdot \boldsymbol{X}
\end{aligned} \tag{3.41}$$

Matriko $\boldsymbol{W_s}$ imenujemo skalirana matrika meritev (*angl. scaled measurement matrix*); $\boldsymbol{P}$ je matrika kamer (*angl. projective motion*), $\boldsymbol{X}$ pa matrika koordinat točk v prostoru (*angl. projective shape*). Če zberemo dovolj nepošumljenih slik točk (u_j^i, v_j^i) in poznamo faktorje λ_j^i, je rang matrike $\boldsymbol{W_s}$ enak štiri in matriko lahko razcepimo v $\boldsymbol{P}$ in $\boldsymbol{X}$. Matriki, ki ju dobimo s tovrstnim razcepom, sta določeni do poljubne 4×4 transformacijske matrike $\boldsymbol{H}$:

$$\boldsymbol{W_s} = \boldsymbol{P} \cdot \boldsymbol{X} = \boldsymbol{P}\boldsymbol{H} \cdot \boldsymbol{H}^{-1}\boldsymbol{X} = \hat{\boldsymbol{P}} \cdot \hat{\boldsymbol{X}} \tag{3.42}$$

Med $\boldsymbol{P}$ in $\boldsymbol{X}$ v enačbi (3.42) lahko namreč vstavimo poljubno nesingularno 4×4 matriko $\boldsymbol{H}$ in tako dobimo neskončno parov matrik $\hat{\boldsymbol{P}}$ in $\hat{\boldsymbol{X}}$, ki zadoščajo enačbi. Postopek kalibracije poišče takšno matriko $\boldsymbol{H}$, da postaneta $\hat{\boldsymbol{P}}$ in $\hat{\boldsymbol{X}}$ evklidski — ta postopek zato imenujemo evklidska stratifikacija (*angl. Euclidean stratification*). Matriko $\boldsymbol{H}$ lahko poiščemo z uvedbo dodatnih omejitev; najosnovnejša je predpostavka, da so piksli slikovne ravnine pravokotni (oziroma da sta osi slikovne ravnine med seboj pravokotni). Druga možna predpostavka je, da so notranji parametri vseh kamer enaki.

Število kamer, potrebnih za uspešno avtokalibracijo, je odvisno od števila parametrov kamer, ki jih poznamo oziroma ki so neznani, a so skupni vsem kameram. Tako potrebujemo vsaj osem kamer, če je edino, kar vemo o njih to,

da imajo pravokotne piksle. Po drugi strani pa avtokalibracijo lahko izvedemo že s samo tremi kamerami, če poznamo slikovne koordinate optičnih središč za vse kamere, ali če imajo vse kamere enake notranje parametre.

Podroben opis postopkov ocene faktorjev λ_j^i in razcepa matrike $\boldsymbol{W_s}$ ter postopka evklidske stratifikacije je na voljo v [26].

Algoritem 3 Avtokalibracijska metoda Tomáša Svobode

1: Z laserskim kazalcem, ki smo ga s koščkom prozorne plastike modificirali tako, da sveti v vse smeri se premikamo po prostoru in s kamerami zajemamo slike. Na slikah poiščemo svetle pike, ki ustrezajo projekciji laserja.
2: Zavržemo napačno detektirane točke, ki jih odkrijemo z RANSAC analizo nad pari točk.
3: Ocenimo projekcijske globine λ_j^i in poiščemo manjkajoče točke.
4: Po potrebi optimiziramo točke z postopkom prilagoditve svežnja (*angl. Bundle Adjustment*).
5: Izvedemo razcep matrike $\boldsymbol{W_s}$, s čimer dobimo matriko kamer in matriko koordinat točk v prostoru.
6: V prejšnjem koraku pridobljeni matriki nadgradimo v evklidski matriki s postopkom evklidske stratifikacije.
7: Izračunamo reprojekcijo točk in napako reprojekcije, na osnovi katere poiščemo odstopajoče točke. Korake 3 – 6 ponavljamo, dokler ne izločimo vseh odstopajočih točk.
8: Po potrebi ocenimo parametre nelinearne distorzije leče z orodjem *Camera Calibration Toolbox for Matlab* [9] in ponovimo korake 2 – 7 (globalna iteracija). Prenehamo, ko se napaka reprojekcije zmanjša pod predpisano vrednost oziroma ko presežemo predpisano število iteracij.
9: Če imamo na voljo informacijo o postavitvi sistema v prostoru, lahko poravnamo koordinatni sistem, v katerem so izraženi parametri kamer in koordinate točk, z zunanjim koordinatnim sistemom.

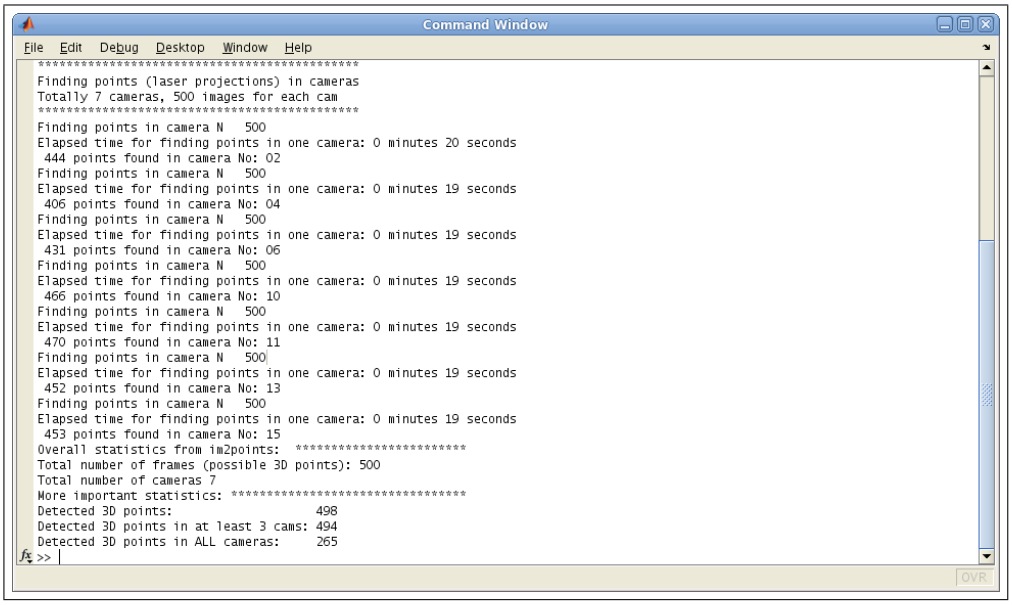
3.4.2 Implementacija metode v programskem okolju Matlab

Implementacija metode, povzete v Algoritmu 3, v okolju Matlab je na voljo na avtorjevi spletni strani [25].

Iskanje točk

Postopek avtokalibracije izvedemo tako, da poženemo dve m-skripti; prva, **im2points**, poišče projekcije točk laserske svetlobe na slikah in shrani slikovne koordinate za poznejšo uporabo. Pred zagonom skripte moramo v nastavitveni datoteki *expname.m* nastaviti ime eksperimenta, čigar parametre nato nastavimo v nastavitveni datoteki *configdata.m*.

Za iskanje točk na slikah je potrebno nastaviti vsaj pot do slik, vzorec njihovih imen in končnice, pa tudi barvo laserja, povprečno velikost točke na slikah in želeno natančnost v pikslih. Po koncu iskanja nam skripta izpiše število vseh slik, število kamer, število posnetkov, na katerih je točka vidna na vseh kamerah in število posnetkov, na katerih je točka vidna na vsaj treh kamerah (Slika 3.10).



Slika 3.10: *Primer izpisa po koncu iskanja točk na slikah s skripto* ***im2points***

Skripta **im2points** ustvari tri izhodne datoteke — *points.mat*, *Res.mat* in *IdMat.mat*.

V datoteki *points.mat* so shranjene koordinate točk v slikovni ravnini v obliki $3m \times n$ matrike (m je število kamer, n pa število točk oziroma posnetkov). V $(3i+1)$-i in $(3i+2)$-i vrstici j-tega stolpca se nahajata slikovni koordinati u in v j-te slike na i-ti kameri, v $(3i+3)$-ji vrstici pa se nahaja vrednost ena, če je i-ta točka vidna na j-ti kameri, oziroma NAN[1], če ni.

[1]Not A Number — vrednost, ki označuje neveljavno število s plavajočo vejico

V datoteki *Res.mat* so shranjene dimenzije slik v obliki $m \times 2$ matrike; v prvem stolpcu so shranjene širine, v drugem pa višine slik, izražene v pikslih.

V datoteki *IdMat.mat* je shranjena $m \times n$ matrika, pri kateri se v i-tem stolpcu in j-ti vrstici nahaja vrednost ena, če je i-ta točka vidna na j-ti kameri, oziroma nič, če ni.

Vse tri datoteke lahko ustvarimo ročno, npr. če smo slikovne koordinate točk določili na kakšen drug način.

Kalibracija

Iskanju točk na slikah sledi dejanska kalibracija. Pred zagonom skripte **gocal** v nastavitveni datoteki *configdata.m* določimo parametre kalibracije.

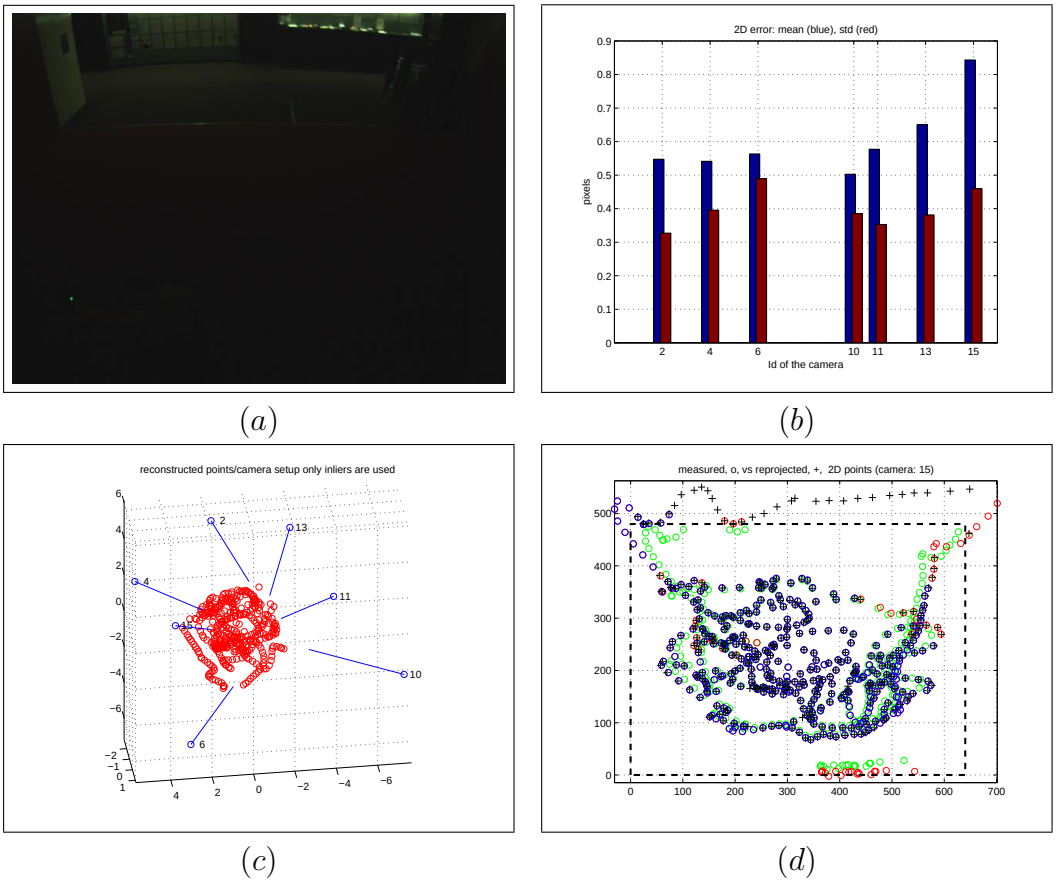
Pomembnejši parameter je začetna toleranca za preverjanje pravilnosti točk na osnovi epipolarne geometrije — ta vrednost vpliva tako na korak 2 kot na korak 7 v postopku, podanem v Algoritmu 3. Vrednost naj bi ustrezala pričakovani radialni distorziji kamer, se pa med iterativnim postopkom optimizacije postopno zmanjšuje.

Nastaviti moramo največje dovoljeno število iteracij in največjo dovoljeno vrednost napake reprojekcije. Poleg tega lahko po želji vklopimo postopek prilagajanja svežnja (*Bundle Adjustment*) bodisi po vsaki iteraciji bodisi na koncu postopka.

Določimo lahko tudi, ali želimo odstraniti učinke radialne distorzije; prek parametrov lahko podamo začetne vrednosti parametrov distorzije leče ter katere od le-teh želimo v nadaljevanju optimirati.

Z zagonom skripte **gocal** se izvede avtokalibracijski postopek 3. Po končanem postopku skripta za vsako kamero izpiše napako reprojekcije in nariše sliko z izmerjenimi in reprojeciranimi točkami, kot jo vidi vsaka od kamer (primer prikazuje Slika 3.11 *d*). Na koncu nariše še sliko rekonstruirane postavitve kamer in rekonstruirane položaje izmerjenih točk (Slika 3.11 *c*).

V primeru, da je napaka reprojekcije relativno visoka, nas program na to opozori in izvede postopek ocenitve nelinearne distorzije zaradi leče. V primeru, da tega ne želimo, moramo v nastavitveni datoteki nastaviti število globalnih iteracij na vrednost ena.



(a) (b) (c) (d)

Slika 3.11: *Rezultat kalibracije z metodo T. Svobode: primer posnetka laserskega kazalca (a), srednja vrednost in standardna deviacija napake reprojekcije za vsako od kamer (b), rekonstruirana postavitev kamer (c) in reprojekcija točk za eno od kamer (d) — prikazane so izmerjene točke (modri krogci), točke, ki jih je postopek zavrgel (rdeči krogci), točke po kalibrirani distorziji leče (zeleni krogci) in reprojecirane točke (črni križci)*

3.4.3 Parametri kamer in poravnava koordinatnega sistema

Rezultat postopka avtokalibracije so notranji in zunanji parametri kamer; slednji so izraženi v neznanem koordinatnem sistemu z izhodiščem v središču oblaka rekonstruiranih točk (Slika 3.11 *c*).

Pogosto želimo imeti parametre izražene v zunanjem koordinatnem sistemu; zato moramo poiskati preslikavo, ki koordinatni sistem rezultata kalibracije poravna z zunanjim koordinatnim sistemom (v splošnem moramo poiskati premik, zasuk in faktorje skaliranja vzdolž vseh treh osi).

Pri poravnavi koordinatnih sistemov si lahko pomagamo s predmeti v prostoru, katerih dimenzije in položaj poznamo, ali pa s poznavanjem koordinat nekaterih točk v prostoru.

Po drugi strani je v okoljih navidezne resničnosti z velikim številom kamer, za katere je bila predstavljena metoda avtokalibracije prvotno razvita, zunanji koordinatni sistem pogosto določen s položajem kamer, zato poravnava koordinatnih sistemov ni problematična.

3.4.4 Pomanjkljivosti metode

Avtokalibracijska metoda T. Svobode ima v primerjavi z Zhangovo metodo in metodo DLT veliko prednost — ne zahteva poznavanja koordinat kontrolnih točk v prostoru. Postopek s pomočjo posnetkov določi tako parametre kamer kot geometrijo kalibracijskega volumna (koordinate kalibracijskih točk v prostoru). Zaradi tega sicer potrebujemo precej več kalibracijskih točk kot v primeru Zhangove metode ali metode DLT, vendar s stališča preprostosti za izvedbo to ni problem, saj nam ni treba izmeriti njihovih koordinat. Vseeno pa moramo, če želimo na koncu izvesti poravnavo koordinatnega sistema, poznati koordinate vsaj nekaj točk.

Prednost avtokalibracijske metode T. Svobode pa je lahko hkrati tudi njena pomanjkljivost. Ker postopek ocenjuje tako parametre kamer kot geometrijo kalibracijskega volumna, je najverjetneje precej bolj občutljiv na šum in odstopanja v slikovnih koordinatah kot kalibracijske metode, ki zahtevajo poznavanje koordinat kontrolnih točk. Vendar pa avtorjevi poskusi kažejo, da se metoda dobro obnese tudi v primerih, ko imamo opravka s kamerami z izrazito distorzijo zaradi leče in odstopajočimi meritvami.

Poglavje 4

Kalibracija sistema Ubisense z metodami za kalibracijo kamer

V pričujočem poglavju se nahaja jedro diplomske naloge — kalibracija sistema Ubisense z metodami za kalibracijo kamer.

Čemu sploh želimo uporabiti tovrstni postopek kalibracije oziroma zakaj bi se radi izognili vgrajenemu postopku kalibracije, ki smo ga spoznali v Podpoglavju 2.4? Na to vprašanje bomo odgovorili v Podpoglavju 4.1, kjer bomo podrobneje predstavili naš namen uporabe sistema Ubisense in z njim povezane razloge za željo po alternativnem postopku kalibracije.

V Podpoglavju 4.2 bomo nato predstavili idejo o ekvivalentnosti informacije o kotih prihoda pri Ubisense senzorjih in informacije o koordinatah v slikovni ravnini pri kamerah. Ta ideja nam omogoča, da senzorje obravnavamo kot navidezne kamere, ki jih lahko kalibriramo z metodami, predstavljenimi v tretjem poglavju.

Pri tem se pojavi vprašanje, ali je nivo šuma v meritvah kotov prihoda pri Ubisense senzorjih primerljiv z nivojem šuma v slikovnih koordonatah pri kamerah. Oziroma, ali je potrebna predobdelava meritev kotov prihoda (npr. filtriranje), preden jih lahko uporabimo kot vhodne podatke za kalibracijo senzorjev z metodami za kalibracijo kamer. Z namenom odgovoriti na to vprašanje bomo v Podpoglavju 4.3 izvedli avtokalibracijsko metodo Tomáša Svobode na surovih (nefiltriranih) podatkih. To metodo smo izbrali zato, ker se po eni strani zdi najprimernejša s stališča enostavnosti izvedbe (ne potrebuje koordinat kontrolnih točk), po drugi strani pa glede na analizo v Podpoglavju 3.4 predvidevamo, da je tudi najbolj občutljiva na šum in odstopanja v meritvah.

Nato bomo v Podpoglavju 4.4 postopek avtokalibracije izvedli še na filtriranih podatkih.

Na koncu, v Podpoglavju 4.5, pa bomo izvedli primerjavo med vsemi tremi metodami, ki smo jih predstavili v tretjem poglavju, in sicer s pomočjo filtriranih podatkov.

4.1 Motivacija za iskanje alternativnega postopka

Kot smo omenili že v drugem poglavju, je sistem Ubisense namenjen stalni (fiksni) namestitvi; ker se položaj in orientacija senzorjev po namestitvi načeloma ne spreminjata, je treba vgrajeni postopek kalibracije sistema, ki smo ga predstavili v Podpoglavju 2.4, izvesti samo enkrat.

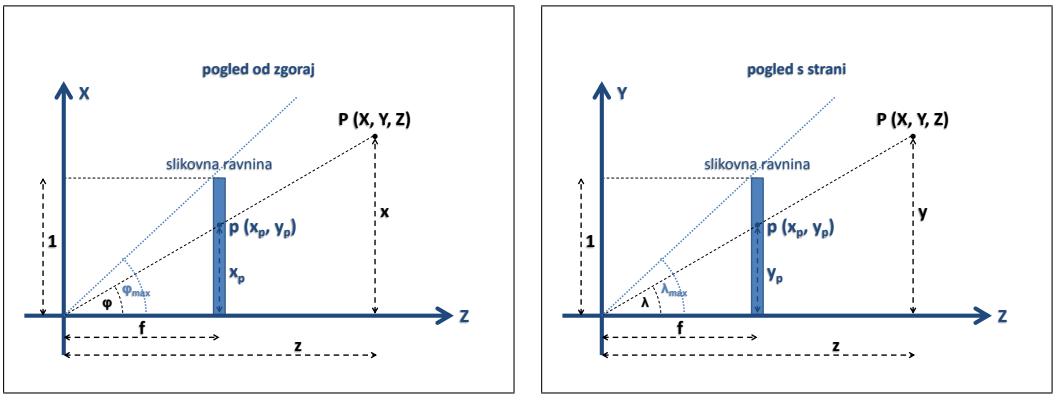
V primeru zgoraj opisanega scenarija tudi določanje položajev senzorjev ni problematično; ker gre za sistematično in načrtovano postavitev, so položaji senzorjev na voljo v načrtu postavitve oziroma moramo položaje mest za pritrditev senzorjev izmeriti, da zagotovimo skladnost z načrtom. V tem primeru lahko tudi predpostavimo razpoložljivost primerne merilne opreme (laserska merilna postaja). S stališča načrtovanih stalnih postavitev sistema Ubisense je vgrajeni postopek kalibracije vsekakor ustrezen in zadovoljiv.

Po drugi strani pa je namen diplomske naloge integracija sistema Ubisense v premičen video sistem za sledenje igralcem v športu. Ubisense senzorje želimo namestiti poleg video kamer na stojala, ki jih prinesemo na prizorišče športnega dogodka (npr. v športno dvorano). Tovrstna mobilna postavitev sistema Ubisense bi omogočila, da sistem uporabimo tudi tam, kjer drugače ne bi bil na voljo — bodisi zaradi cene sistema bodisi zaradi pomanjkanja potrebne infrastrukture (npr. improvizirana igrišča na prostem).

Izkazalo se je, da glavni problem zgoraj opisane postavitve sistema Ubisense predstavlja njegova kalibracija oziroma zahteva vgrajenega postopka kalibracije po poznavanju položaja senzorjev. Koordinatni sistem običajno pripnemo na igrišče, stojala s senzorji pa moramo zaradi varnosti ter primernejšega zornega kota namesto ob rob igrišča namestiti na tribune. Takšna postavitev zelo oteži natančno določitev položaja senzorjev. Problem bi sicer lahko omilili z uporabo laserske merilne postaje, vendar ta v našem primeru ni na voljo.

Zato želimo poiskati alternativni postopek kalibracije senzorjev, ki ne bi potreboval poznavanja položajev senzorjev. Kot smo nakazali že koncu Podpoglavja 2.7, želimo kalibracijo izvesti na osnovi meritev kotov prihoda, ki jih lahko dobimo z izvozom dogodkov v .xcm datoteko, dobljeni parametri senzorjev pa bi bili primerni za direkten vnos v Ubisensov program *Location Engine Config* za nastavljanje in nadzor delovanja sistema.

4.2 Senzorji kot navidezne kamere



(a) (b)

Slika 4.1: *Projekcija točke P v slikovno ravnino; pogled od zgoraj (a) in pogled s strani (b)*

Ponovno se vrnimo k centralno projekcijskemu modelu kamere, ki smo ga predstavili v Podpoglavju 3.1. Na Sliki 4.1 je zopet prikazana projekcija točke P iz zunanjega koordinatnega sistema v točko p v slikovni ravnini kamere, ki jo lahko opišemo z enačbama (3.1).

Kot lahko vidimo na Sliki 4.1, imamo tudi pri kamerah pravzaprav opravka s koti prihoda; žarki iz točke P prihajajo v središče projekcije O pod kotoma, katerih tangensa lahko zapišemo kot:

$$\begin{aligned} \tan\varphi &= \frac{X}{Z} \\ \tan\lambda &= \frac{Y}{Z} \end{aligned} \tag{4.1}$$

S φ smo označili azimut, z λ pa elevacijo. Enačbi (3.1) lahko zapišemo kot:

$$\begin{aligned} x &= f \cdot \tan\varphi \\ y &= f \cdot \tan\lambda \end{aligned} \tag{4.2}$$

Goriščna razdalja f predstavlja oddaljenost slikovne ravnine od središča projekcije in (pri fiksni velikosti slikovne ravnine) določa vidno polje kamere φ_{max} oziroma λ_{max}. Predpostavimo, da je slikovna ravnina kvadratna ($\varphi_{max} = \lambda_{max}$) in da sta koordinati (x, y) v optičnem središču enaki $(0, 0)$, koordinati levega zgornjega kota pa $(1, 1)$. Za goriščno razdaljo potem velja zveza:

$$\frac{1}{f} = \tan\varphi_{max} = \tan\lambda_{max} \tag{4.3}$$

Enačbi projekcije točke v slikovno ravnino tako postaneta:

$$\begin{aligned} x &= \frac{\tan\varphi}{\tan\varphi_{max}} \\ y &= \frac{\tan\lambda}{\tan\lambda_{max}} \end{aligned} \tag{4.4}$$

Tako dobljeni koordinati v slikovni ravnini (x, y) nato diskretiziramo v (u, v) z enačbama (3.2); poleg tega upoštevamo, da izhodišče koordinatnega sistema v diskretizirani slikovni ravnini običajno postavimo v zgornji levi kot.

Iz enačb (4.4) sledi, da koordinati točke v slikovni ravnini nosita isto informacijo kot kota prihoda. Zato si v primeru uporabe Ubisense senzorjev lahko predstavljamo, da se pred senzorjem nahaja navidezna slikovna ravnina, v katero se točke ("slike" značk) preslikajo glede na kot prihoda signala z značke.

Ubisense senzorje si tako lahko predstavljamo kot navidezne kamere — in jih kot takšne kalibriramo z metodami za kalibracijo kamer. Podatke o kotih prihoda, ki jih dobimo z izvozom dogodkov v .xcm datoteko, s pomočjo zgornjih enačb preslikamo v navidezno slikovno ravnino in tako dobljene navidezne slike uporabimo kot podatke za kalibracijsko metodo.

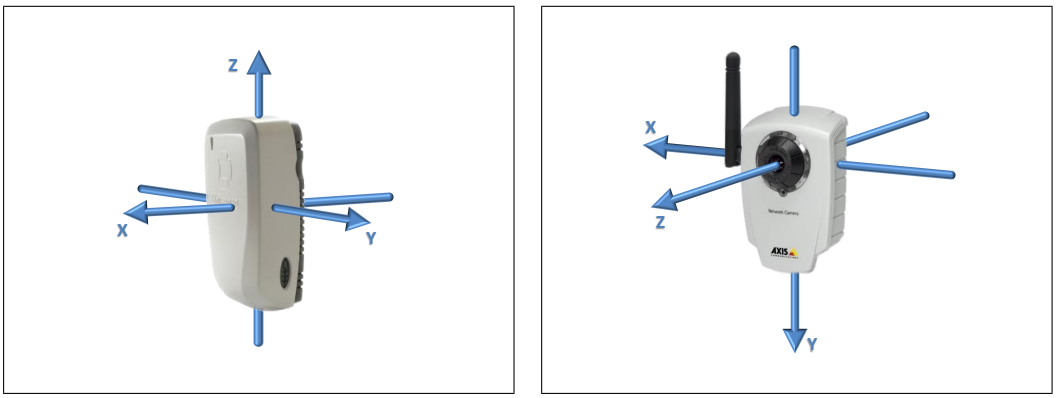
4.2.1 Notranji in zunanji parametri navideznih kamer

Zunanji parametri navideznih kamer ustrezajo dejanskemu položaju in orientaciji senzorjev, notranje parametre kamer pa izberemo:

- Optično središče slike postavimo v sredino navidezne slikovne ravnine.

- Velikost piksla (ločljivost slikovne ravnine) izberemo poljubno; ker nikjer v postopku ne zaokrožujemo številskih vrednosti, ločljivost slikovne ravnine nima vpliva na rezultate kalibracije [1].

- Z goriščno razdaljo (oziroma ekvivalentnima φ_{max} in λ_{max}) določimo efektivno vidno polje senzorjev; v primeru, ko je $\varphi > \varphi_{max}$ oziroma $\lambda > \lambda_{max}$, "slika" značke ni vidna. Z izbiro vidnega polja, ki je manjše od nazivnih 120° (oziroma 100°), lahko dosežemo izločanje meritev z robov vidnega polja senzorjev.



(*a*) (*b*)

Slika 4.2: *Koordinatni sistem senzorja (a) in kamere (b)*

Pri ocenjenih zunanjih parametrih kamer moramo upoštevati, da program *Location Engine Config* uporablja drugačen koordinatni sistem kot metode za kalibracijo kamer (oziroma uporabljene implementacije v Matlabu). Koordinatni sistem kamere in koordinatni sistem senzorja sta prikazana na Sliki 4.2; koordinatna sistema sta različno zasukana, kar moramo upoštevati, ko želimo prebrati RPY kote iz rotacijskih matrik, ki smo jih določili s kalibracijo navideznih kamer.

[1] Edino pri metodi T. Svobode je vrednost tolerance za določanje odstopajočih točk izražena v pikslih, zato moramo izbrati začetno vrednost v ustreznem velikostnem razredu

4.3 Avtokalibracija s pomočjo surovih podatkov

Glede na omejitve in pomanjkljivosti metod za kalibracijo kamer, ki smo jih opisali v tretjem poglavju, se najprimernejša za uporabo zdi avtokalibracijska metoda T. Svobode. Ker metoda zahteva samo, da po prostoru premikamo zlahka prepoznaven svetel predmet (v primeru kalibracije sistema Ubisense slednjega nadomestimo z značko), bi se z njeno uporabo lahko elegantno izognili merjenju koordinat točk v prostoru.

Ideja praktične izvedbe kalibracije je sledeča: ker želimo sistem Ubisense uporabiti za sledenje igralcem na igrišču, lahko za avtokalibracijo uporabimo kar značke, ki jih nosijo igralci. Tako nam igralci, ko se gibljejo po igrišču, hkrati pokrijejo celotno celico (igrišče). Ker je v vsakem časovnem oknu aktivna samo ena značka, lahko za avtokalibracijo uporabimo vse značke, ki so prisotne na igrišču, saj nimamo problemov z iskanjem korespondenc. Ta problem bi se pojavil, če bi šlo za kalibracijo navadnih kamer z več svetlimi predmeti.

4.3.1 Eksperiment in rezultati

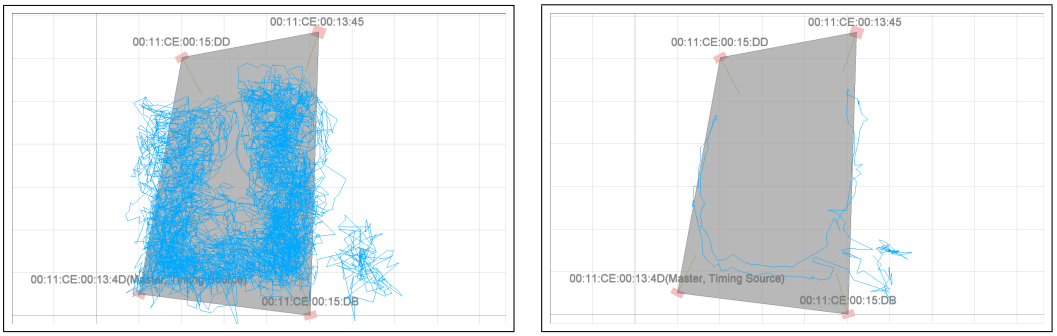
Zgoraj opisano idejo smo preizkusili s postavitvijo štirih senzorjev v vogale laboratorija ter nošenjem ene značke po prostoru. Sistem Ubisense smo pred začetkom eksperimenta kalibrirali z vgrajenim postopkom kalibracije; to za zajem surovih podatkov sicer ni potrebno, nam pa omogoča ovrednotenje rezultatov postopka avtokalibracije.

Parametri senzorjev so prikazani v Tabeli 4.1. Senzorje smo obrnili proti sredini laboratorija tako, da gledajo naravnost.

	X $[m]$	**Y** $[m]$	**Z** $[m]$	**R** [°]	**P** [°]	**Y** [°]
Senzor #1	5,21	6,62	1,77	0,00	-12,93	-108,80
Senzor #2	1,00	0,50	1,86	0,00	0,48	64,20
Senzor #3	5,01	0,00	1,54	0,00	2,26	106,05
Senzor #4	2,01	6,03	1,63	0,00	2,37	-61,94

Tabela 4.1: *Parametri senzorjev, določeni z vgrajenim postopkom kalibracije*

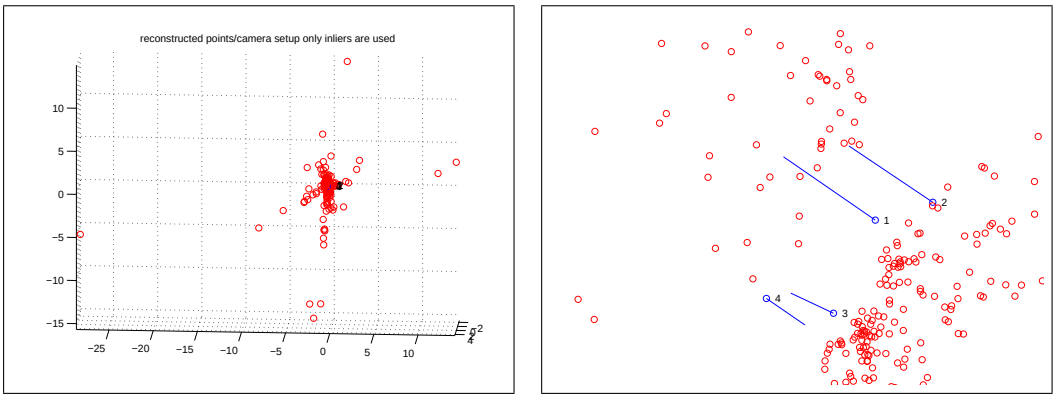
S programom *Location Engine Config* smo zajeli največje možno število dogodkov — 10000. Pokritost prostora je prikazana na Sliki 4.3, kjer je s svetlo modro barvo označena pot značke po prostoru.



Slika 4.3: *Pot značke po prostoru; eksperiment z 10000 dogodki (a) in s 310 dogodki (b)*

Za projekcijo kotov prihoda, ki smo jih prebrali iz izvoza surovih podatkov v .xcm datoteko, smo uporabili naslednje parametre, ki so hkrati notranji parametri naših navideznih kamer: slikovna ravnina dimenzij 600×600 pikslov z optičnim središčem v sredini ter vidnim poljem 120° v horizontalni in vertikalni smeri. Začetno vrednost tolerance pri validaciji smo nastavili na 50 pikslov[2].

Postopek avtokalibracije je na sodobnem osebnem računalniku[3] trajal več kot eno uro, kar je posledica velike količine vhodnih podatkov. Po drugi strani pa je od 10000 zajetih točk začetno validacijo prestalo samo 697 točk, do konca postopka pa je ostalo samo 393 veljavnih točk. Postopek avtokalibracije ni uspel; kot lahko vidimo na Sliki 4.4, dobimo povsem napačno ocenjene položaje ter orientacije senzorjev in položaje točk.



Slika 4.4: *Rezultat neuspele avtokalibracije*

[2]Ker se vrednost tolerance med kalibracijo postopno zmanjšuje, se izkaže, da sprememba začetne vrednosti na končni rezultat kalibracije nima bistvenega vpliva.

[3]procesor: Intel Core i7 z delovnim taktom $2,67\ GHz$

Vzrok za to je predvsem šum v meritvah, ki v primeru kalibracije navadnih kamer ni prisoten v tolikšni meri. Poleg tega imamo težave z odboji in zakrivanjem: kljub temu, da smo med nošenjem značko držali stran od sebe, je bilo od vseh zajetih točk samo $49,39\%$ točk zaznanih na vseh štirih senzorjih, $33,86\%$ na treh senzorjih, $12,86\%$ na dveh in $3,39\%$ na samo enem senzorju, $0,5\%$ pa na nobenem. Čeprav imamo relativno visok delež primerov, ko je bila značka zaznana na vseh štirih senzorjih, pa v številnih od teh primerov zaradi odbojev dobimo napačne meritve kotov prihoda na vsaj enem od senzorjev. Zaradi tega je meritev na vseh štirih senzorjih nekonsistentna in posledično jo postopek med validacijo zavrže.

Za samo delovanje (kalibriranega) sistema Ubisense to ne predstavlja težav, saj ima vseeno dovolj meritev, da pravilno določi položaj značke. Pri avtokalibraciji pa se po drugi strani zanašamo samo na podatke o kotih prihoda; ker so ti najbolj občutljivi na šum in odboje, avtokalibracija ne uspe.

4.3.2 Teoretični koti prihoda

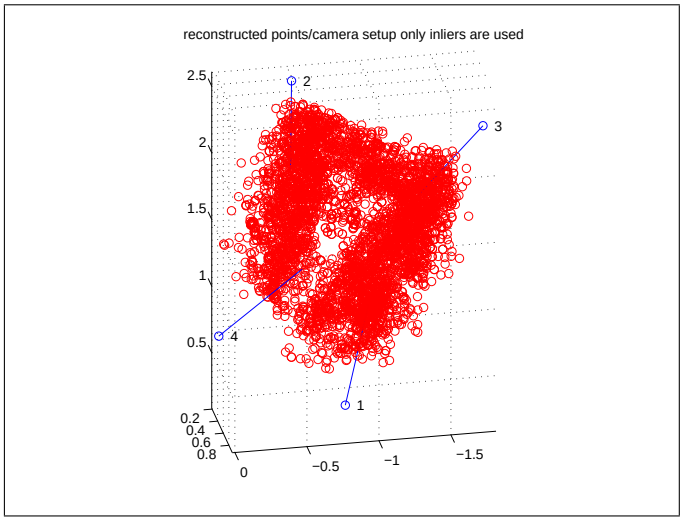
Za primerjavo smo postopek avtokalibracije izvedli še s "teoretičnimi koti prihoda": ker smo sistem Ubisense pred eksperimentom kalibrirali, smo skupaj s surovimi podatki dobili tudi ocene položajev točk. Ker poznamo tudi parametre senzorjev, lahko izračunamo teoretične kote prihodov, ki bi jih morali senzorji izmeriti.

Če izvedemo avtokalibracijo s tako dobljenimi teoretičnimi koti prihoda, postopek uspe, kar potrjuje domnevo, da imamo pri izmerjenih podatkih težave s šumom ter odboji; rezultat je prikazan na Sliki 4.5.

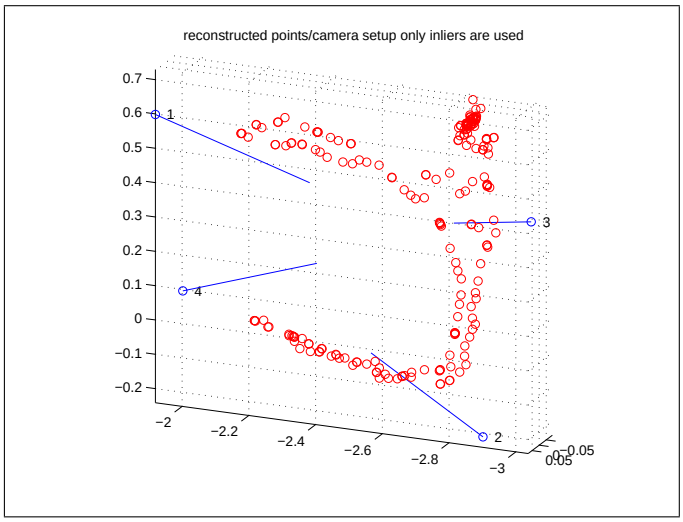
4.3.3 Eksperiment z manjšim številom meritev

Zgoraj opisani postopek smo ponovili še enkrat, le da smo tokrat zajeli bistveno manj dogodkov — 310. V tem primeru postopek avtokalibracije med validacijo izloči preveliko število točk in ne uspe (pride do napake v programu). V primeru uporabe teoretičnih kotov prihoda avtokalibracija zopet uspe (Slika 4.6).

Slika 4.7 prikazuje vrednosti kotov prihoda na vseh štirih senzorjih; z rdečo so označene izmerjene vrednosti, z modro pa izračunane (teoretične) vrednosti. Kot lahko vidimo, pride na nekaterih senzorjih do precejšnjih odstopanj med izmerjeno

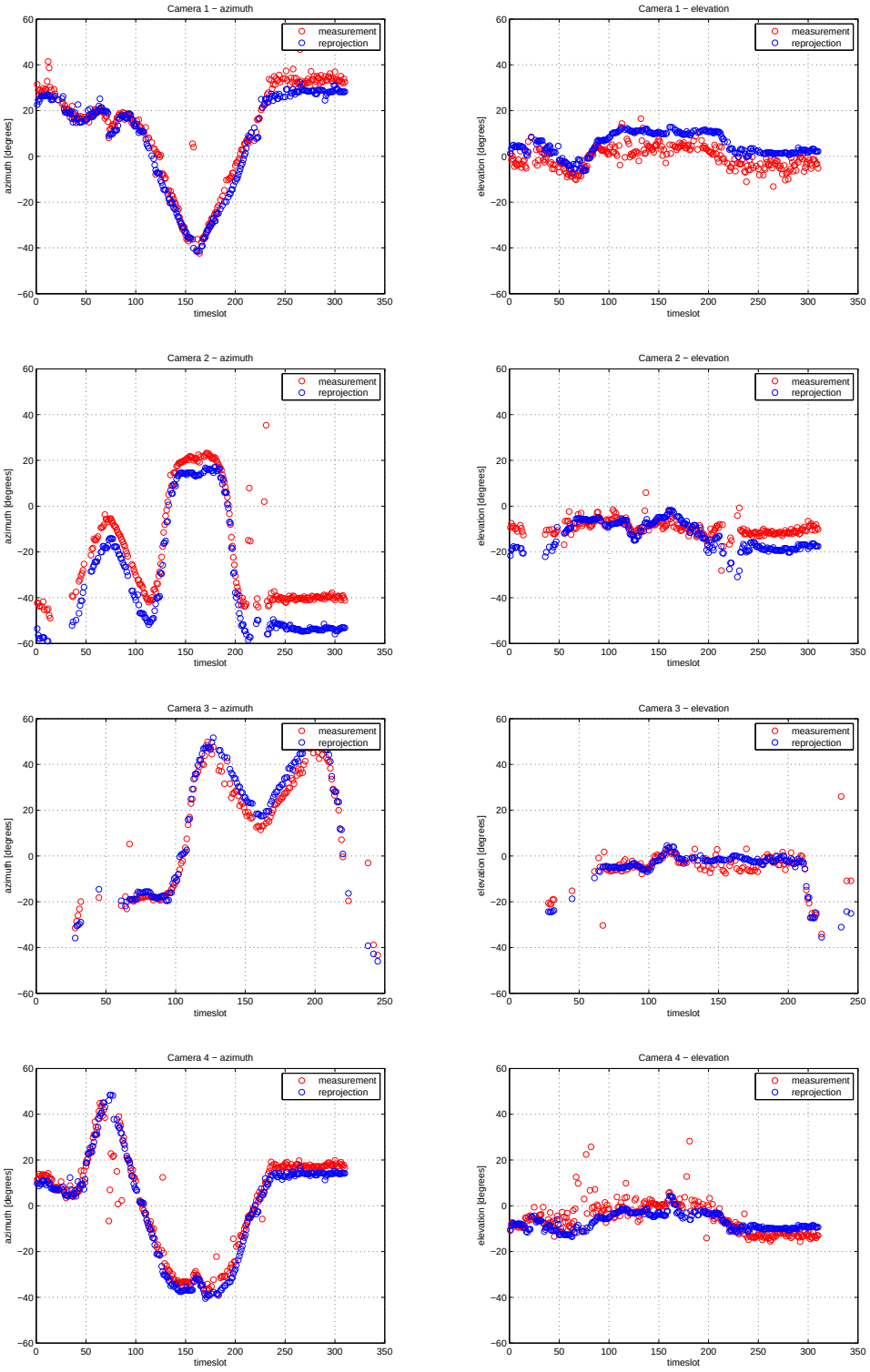


Slika 4.5: *Rezultat uspele avtokalibracije s teoretičnimi koti prihoda (10000 dogodkov)*



Slika 4.6: *Rezultat uspele avtokalibracije s teoretičnimi koti prihoda (310 dogodkov)*

in teoretično vrednostjo kotov prihoda. Vzrok za to so predvsem odboji in na nekaterih mestih šum, delno pa je vzrok tudi dejstvo, da so teoretične vrednosti kotov izračunane iz položajev točk, ki jih je določil sistem Ubisense in niso nujno povsem točni.



Slika 4.7: *Vrednosti kotov prihoda na vseh štirih senzorjih*

4.4 Avtokalibracija s pomočjo filtriranih podatkov

Prejšnji poskus je pokazal, da imamo pri avtokalibraciji sistema Ubisense z metodo T. Svobode težave predvsem zaradi šuma v meritvah in odstopanj, ki so posledica odbojev. Ena od idej za zmanjšanje vpliva šuma in odbojev je, da v vsaki točki zajamemo več meritev in pustimo postopku avtokalibracije, da med validacijo zavrže meritve, ki preveč odstopajo.

Ker mora značka v vsaki točki nekaj časa mirovati, za avtokalibracijo ne moremo uporabiti značk, ki jih nosijo igralci. Poleg tega lahko iz prejšnjega eksperimenta zaključimo tudi, da v primeru nošenja značke pride do zakrivanja in posledično do izraza pridejo odboji, kar pokvari avtokalibracijo. Ideja praktične izvedbe kalibracije je zato sledeča: značko premikamo po različnih točkah na igrišču (različne višine lahko dosežemo npr. z uporabo stojala), pri čemer značko v vsaki točki pustimo nekaj sekund, da zajamemo več meritev. Poleg tega oseba, ki značko premika, med zajemom meritev poskrbi, da ne zakriva značke; to pomeni, da mora biti igrišče med zajemom meritev bolj ali manj prazno. Še vedno pa lahko uporabimo več značk, ker vrstni red zajema dogodkov ni pomemben.

4.4.1 Eksperiment in rezultati

Tudi tokrat smo eksperiment izvedli v laboratoriju in s štirimi senzorji. Senzorje smo obrnili proti sredini prostora tako, da gledajo proti tlom, s čimer dosežemo optimalno pokritost prostora. Sistem smo najprej kalibrirali z vgrajenim postopkom kalibracije; parametri senzorjev so prikazani v Tabeli 4.2.

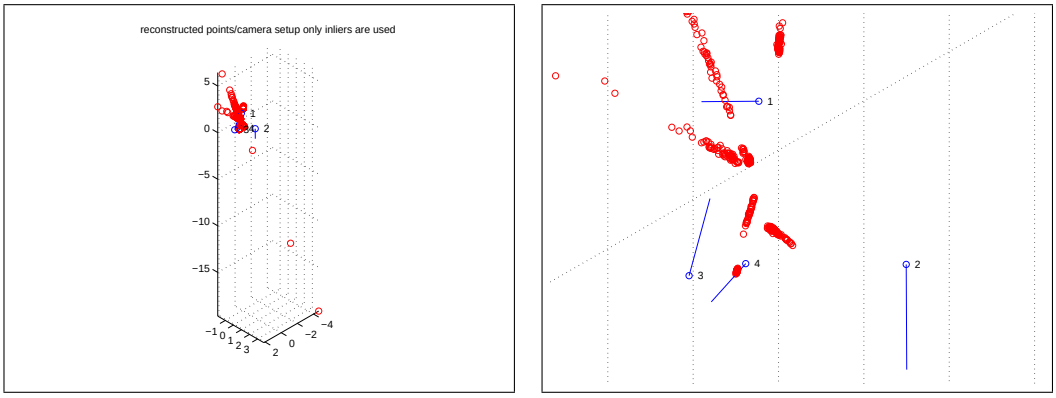
	X $[m]$	**Y** $[m]$	**Z** $[m]$	**R** [°]	**P** [°]	**Y** [°]
Senzor #1	0,61	6,40	2,01	0,00	-31,53	-53,82
Senzor #2	3,19	3,30	2,15	0,00	-28,58	130,36
Senzor #3	3,65	6,30	2,10	0,00	-28,47	-141,01
Senzor #4	0,44	0,45	2,03	0,00	-32.90	41.82

Tabela 4.2: *Parametri senzorjev, določeni z vgrajenim postopkom kalibracije*

Značko smo postavili na različna naključno izbrana mesta v laboratoriju (različne višine smo dosegli s postavitvijo na tla, na mize, stole in kartonaste škatle). Na vsakem mestu smo zajeli 100 dogodkov; skupno smo zajeli 10000 dogodkov, torej 100 različnih točk v prostoru. Med premikanjem značke smo zajem dogodkov izključili.

Za projeciranje kotov prihodov v navidezno slikovno ravnino smo uporabili enake parametre navidezne kamere kot v prejšnjem eksperimentu: slikovna ravnina dimenzij 600 × 600 pikslov z optičnim središčem v sredini in vidnim poljem 120°.

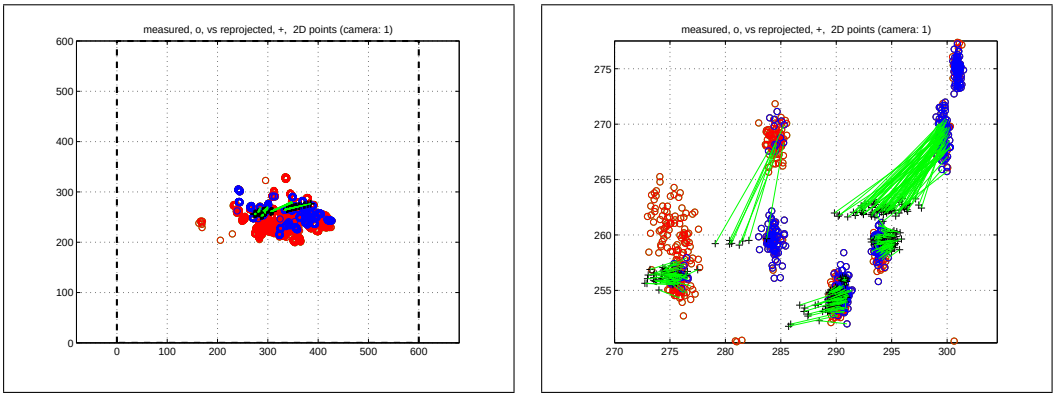
Zaradi velikega števila zajetih dogodkov tudi tokrat postopek avtokalibracije traja več kot uro; na koncu avtokalibracije ne uspe. Od 10000 točk jih na koncu postopka ostane veljavnih samo 473. Slika 4.8 prikazuje rezultat neuspele kalibracije, na Sliki 4.9 pa so prikazane reprojekcije točk skupaj z izmerjenimi točkami v eni od navideznih slikovnih ravnin.



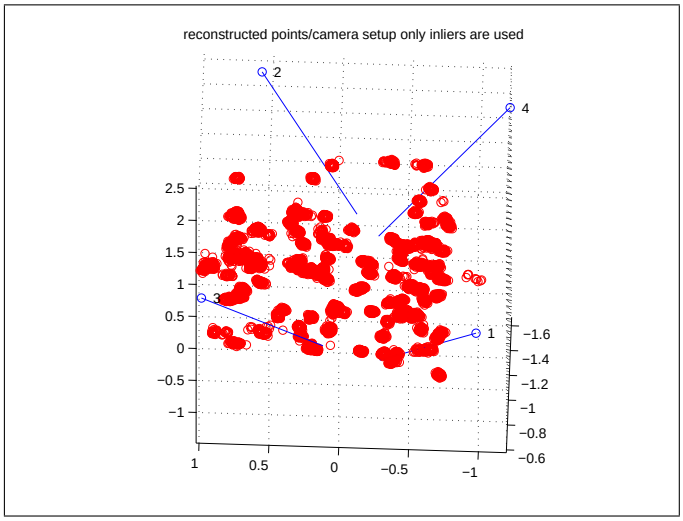
Slika 4.8: *Rezultat neuspele avtokalibracije*

Modri krogci označujejo veljavne izmerjene točke, rdeči krogci pa izmerjene točke, ki jih je postopek avtokalibracije zaradi prevelikega odstopanja zavrgel kot neveljavne. S črnimi križci so označene reprojekcije veljavnih točk, njihova odstopanja od ustreznih veljavnih izmerjenih točk pa so označene z zelenimi črtami. Kot lahko vidimo, je postopek med validacijo zavrgel veliko večino točk, razen nekaj gruč, od katerih pa vsaka gruča ustreza pošumljenim meritvam z enega samega položaja značke v prostoru. Postopek je zato kalibracijo v resnici izvedel na samo nekaj točkah v prostoru, iz katerih pa ni dobil zadostne informacije za pravilno izvedbo kalibracije.

Če postopek izvedemo s teoretičnimi (preračunanimi) vrednostmi kotov prihoda, avtokalibracija uspe; rezultat je prikazan na Sliki 4.10.



Slika 4.9: *Izmerjene in reprojecirane točke v slikovni ravnini*



Slika 4.10: *Rezultat uspele avtokalibracije s teoretičnimi koti prihoda*

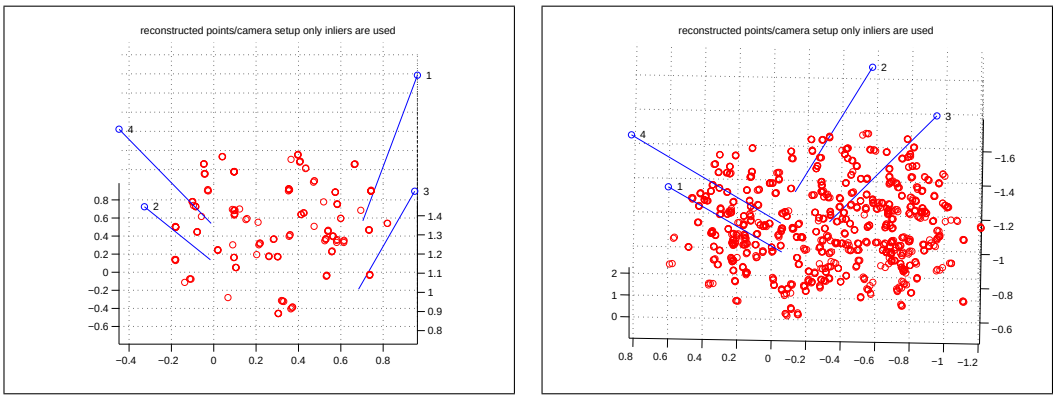
4.4.2 Uporaba mediane

Iz zgoraj opisanega eksperimenta lahko zaključimo, da večje število meritev v eni točke ne pripomore k izboljšanju rezultatov avtokalibracije; postopek očitno ni dovolj robusten, da bi pravilno deloval v primeru nivoja šuma, kakršen je prisoten v meritvah kotov prihoda z Ubisense senzorjev.

Zato iz 100 meritev v vsaki točki izračunamo robustno srednjo vrednost — mediano. V tem primeru postane vrstni red zajema dogodkov pomemben oziroma bi v primeru uporabe več značk morali dogodke razvrščevati po oznakah značk.

Z izračunom mediane smo izrazito zmanjšali količino vhodnih podatkov, zato se postopek avtokalibracije izvede v nekaj minutah. Za preizkus smo posneli

30000 dogodkov (tri .xcm datoteke) v 300 točkah v prostoru in za vsako točko izračunali mediano kotov prihoda. Zaradi zmanjšane količine podatkov lahko uporabimo tudi postopek prilagajanja svežnja (*Bundle Adjustment*). Postopek avtokalibracije je v tem primeru nekoliko uspešnejši (Slika 4.11 *a*); kljub temu so dobljene orientacije senzorjev usmerjene preveč v tla, kar potrdi tudi primerjava z rezultatom avtokalibracije s teoretičnimi koti prihoda (Slika 4.11 *b*).



Slika 4.11: *Rezultat avtokalibracije z izračunom mediane; izmerjeni podatki (a) in sintetični podatki (b)*

4.5 Primerjava treh metod

Iz prejšnjih eksperimentov lahko zaključimo, da se avtokalibracijska metoda T. Svobode pri kalibraciji Ubisense senzorjev ne obnese najbolje; razlog so problemi, ki so vezani na radijsko lociranje in sistem Ubisense. Z uporabo mediane težave pri avtokalibraciji lahko omilimo do te mere, da lahko njene rezultate primerjamo z rezultati Zhangove metode in metode DLT, kar bomo storili v nadaljevanju.

Za izvedbo kalibracije z Zhangovo metodo oziroma metodo DLT potrebujemo kontrolne točke z znanimi koordinatami. S tem sicer izgubimo prednost, ki jo je ponujala avtokalibracijska metoda T. Svobode, vendar pa predpostavimo, da je še vedno precej lažje izmeriti koordinate kontrolnih točk, kot pa izmeriti položaje senzorjev. Ta predpostavka je še posebej upravičena v primeru kalibracije v športni dvorani. Pri določanju koordinat kontrolnih točk si lahko namreč pomagamo z oznakami na igrišču.

Kalibracijo zato izvedemo tako, da postavimo značko na ustrezno izbrana mesta na igrišču in zajamemo meritve. V primeru uporabe Zhangove metode

kontrolne točke lahko ležijo v isti ravnini, zato lahko značko vedno postavimo na tla.

4.5.1 Testno okolje

Eksperiment smo tokrat izvedli na prostem. S tem razmere približamo razmeram v športni dvorani; v celici (na igrišču) načeloma ni večjih odbojnih površin, zaradi česar se vpliv odbojev precej zmanjša. Poleg tega lažje dosežemo dobro pokritost prostora s kontrolnimi točkami.

Za preizkus ideje smo postavili štiri senzorje v vogale pravokotnega območja dimenzij $6 \times 4\ m$. Znotraj območja smo narisali mrežo s kvadrati velikosti $1\ m$, s čimer smo dobili 35 točk z znanimi koordinatami. V vsako od točk smo postavili značko, in sicer enkrat na tla, enkrat pa en meter nad tlemi. Tako smo dobili 70 položajev značk; za vsakega smo posneli 100 dogodkov.

Pred izvedbo meritev smo sistem Ubisense kalibrirali z vgrajenim postopkom kalibracije; parametri senzorjev so prikazani v prvem stolpcu Tabele 4.5.

4.5.2 Rezultati

Kalibracijo z Zhangovo metodo oziroma metodo DLT smo izvedli z orodjem *Camera Calibration Toolbox for Matlab*.

Za notranje parametre navideznih kamer smo izbrali slikovno ravnino dimenzij 600×600 pikslov z optičnim središčem v sredini. Za velikost vidnega polja senzorjev smo najprej uporabili 120°, za primerjavo pa smo kalibracijo izvedli tudi z vidnim poljem 60°.

Ker nas zanimajo samo zunanji parametri navideznih kamer (notranje parametre smo izbrali sami), smo izključili optimizacijo notranjih parametrov in parametrov distorzije zaradi leče. S tem predvsem preprečimo, da bi postopek kalibracije poskušal kompenzirati šum meritev s spreminjanjem notranjih parametrov.

Kalibracijo smo izvedli na tri načine; prvič samo s kontrolnimi točkami na tleh (A), drugič samo s kontrolnimi točkami nad tlemi (B), tretjič pa z vsemi kontrolnimi točkami (A+B). Ker v prvih dveh primerih kontrolne točke ležijo v

isti ravnini, je za kalibracijo uporabljena Zhangova metoda, v tretjem primeru, ko točke ležijo v dveh ravninah, pa metoda DLT.

Za vsako od točk smo iz 100 meritev azimuta oziroma elevacije (za vsak senzor) izračunali eno samo meritev z uporabo mediane, s čimer smo zmanjšali vpliv odbojev in šuma. Pozneje se je izkazalo, da lahko zajamemo bistveno manj meritev; tudi če od 100 zajetih meritev za izračun mediane uporabimo samo 10 meritev, ne pride do bistvenih sprememb v rezultatih.

Izkaže se, da nam točke, ki se nahajajo na robu mreže in so na robu vidnega polja senzorjev, precej pokvarijo kalibracijo — zato smo jih izločili.

Tekom kalibracije poskuša orodje *Camera Calibration Toolbox for Matlab* minimizirati napako reprojekcije; končne vrednosti napake reprojekcije so prikazane v Tabeli 4.3. Kot lahko vidimo, je napaka reprojekcije precej večja, če namesto vidnega polja 120° uporabimo 60°. To je razumljivo, saj se ob zmanjšanju vidnega polja in nespremenjeni ločljivosti navidezne slikovne ravnine absolutna napaka v pikslih, ki ustreza eni kotni stopinji, poveča.

Vidno polje	120°			60°		
Kontrolne točke	**A**	**B**	**A+B**	**A**	**B**	**A+B**
Napaka X [*px*]	2,62	3,14	3,50	6,80	9,57	9,94
Napaka Y [*px*]	3,20	5,23	5,22	8,20	11,61	12,40

Tabela 4.3: *Napaka reprojekcije pri Zhangovi metodi oziroma metodi DLT v horizontalni in vertikalni smeri*

Zato je bolj kot napaka reprojekcije zanimiva napaka rekonstrukcije. V primeru kalibracije z Zhangovo metodo oziroma metodo DLT napako rekonstrukcije izračunamo tako, da iz dobljenih notranjih in zunanjih parametrov navideznih kamer izračunamo parametre DLT in nato rekonstruiramo točke po metodi, opisani v Podpoglavju 3.2.5. S pomočjo rekonstruiranih točk in meritev nato izračunamo napako rekonstrukcije (Podpoglavje 3.2.7).

Vrednosti napake rekonstrukcije so prikazane v Tabeli 4.4. Poleg napake rekonstrukcije Zhangove metode in metode T. Svobode Tabela 4.4 vsebuje še dve napaki. Prva je napaka sistema Ubisense, ki jo lahko izračunamo kot odstopanje koordinat točk, ki jih je določil sistem Ubisense, od (ročno) izmerjenih koordinat. Druga napaka pa je napaka rekonstrukcije na osnovi kotov prihoda in parametrov senzorjev, ki smo jih določili s standardnim postopkom kalibracije; v tem primeru je postopek enak kot pri izračunu napake rekonstrukcije za Zhangovo metodo, le

da kot zunanje parametre navideznih kamer uporabimo že določene parametre senzorjev, notranje parametre pa izberemo poljubno.

	Ubisense	**AoA**	**Zhang / DLT**						**Svoboda**
			120°			60°			
			A	**B**	**A+B**	**A**	**B**	**A+B**	
μ $[m]$	0,1699	0,2090	0,1285	0,1419	0,1128	0,1207	0,1354	0,1060	0,1635
σ $[m]$	0,0987	0,1037	0,0803	0,0639	0,0510	0,0797	0,0540	0,0529	0,1074

Tabela 4.4: *Napaka rekonstrukcije; srednja vrednost (μ) in standardna deviacija (σ)*

V Tabeli 4.5 so prikazani zunanji parametri senzorjev, ki smo jih določili s pomočjo kalibracije navideznih kamer, za primerjavo pa so podani tudi parametri, ki smo jih določili z vgrajenim postopkom kalibracije sistema Ubisense; vrednosti odstopanj parametrov so prikazane na Sliki 4.12. Kot lahko vidimo, so si parametri zadovoljivo podobni; prav tako lahko vidimo, da ob zmanjšanju vidnega polja senzorjev ali uporabe kontrolnih točk v eni sami ravnini ne pride do večjih sprememb v izračunanih parametrih.

Avtokalibracija z metodo T. Svobode in zajetimi podatki uspe, vendar samo pod določenimi pogoji: za vsako od točk smo iz 100 meritev azimuta oziroma elevacije (za vsak senzor) izračunali štiri meritve z uporabo mediane na ustreznih intervalih; pri izračunu ene same meritve avtokalibracija ne uspe. Prav tako avtokalibracija ne uspe, če je vidno polje senzorjev manjše od 120°. Poleg tega moramo za uspešno avtokalibracijo uporabiti vse kontrolne točke (tudi tiste na robu mreže).

Pri postopku kalibracije smo uporabili postopek prilagajanja svežnja (*Bundle Adjustment*), ki naj bi pomagal pri pošumljenih meritvah. Prav tako smo izvedli samo eno globalno iteracijo, s čimer smo onemogočili izvedbo postopka za oceno distorzije zaradi leče, ki bi se sprožil zaradi relativno visoke napake reprojekcije, vendar v našem primeru ni smiseln. Implementacija metode v Matlabu nam po končani kalibraciji izpiše vrednosti napake reprojekcije, ki so prikazane v Tabeli 4.6.

Rezultat kalibracije so parametri senzorjev in koordinate kontrolnih točk; oboji so izraženi v koordinatnem sistemu, ki ima izhodišče v središču oblaka kontrolnih točk in ima poljubno orientacijo (Slika 4.13). V našem primeru

		Ubisense	Zhang / DLT						Svoboda
			120°			60°			
			A	**B**	**A+B**	**A**	**B**	**A+B**	
Senzor #1	**X** $[m]$	6,00	5,96	5,88	5,97	5,84	5,75	5,87	6,31
	Y $[m]$	4,10	4,37	4,16	4,24	4,41	4,17	4,27	4,48
	Z $[m]$	2,20	2,04	2,08	2,21	1,97	2,07	2,21	1,99
	R [°]	0,00	0,57	-5,01	-2,54	2,81	0,02	1,06	9,50
	P [°]	-37,90	-36,42	-37,13	-38,69	-35,63	-37,09	-38,65	-26,00
	Y [°]	-131,18	-127,78	-131,06	-129,71	-125,66	-127,98	-127,50	-139,57
Senzor #2	**X** $[m]$	0,00	-0,42	-0,34	-0,42	-0,44	-0,35	-0,43	-0,53
	Y $[m]$	0,00	-0,30	-0,06	-0,17	-0,29	-0,07	-0,19	-0,50
	Z $[m]$	2,17	1,86	2,01	2,11	1,87	2,04	2,13	2,33
	R [°]	0,00	1,05	-1,72	-0,79	0,49	-1,59	-0,80	-1,34
	P [°]	-32,74	-28,02	-29,14	-30,80	-28,05	-29,78	-31,01	-26,45
	Y [°]	44,38	44,34	41,83	42,85	43,98	41,84	42,86	35,96
Senzor #3	**X** $[m]$	5,90	5,95	5,95	5,98	5,97	5,97	5,98	6,04
	Y $[m]$	0,00	-0,22	-0,15	-0,19	-0,21	-0,17	-0,19	0,17
	Z $[m]$	1,50	1,36	1,50	1,49	1,38	1,52	1,50	1,36
	R [°]	0,00	-1,09	0,54	-0,08	-0,93	1,40	0,27	-7,49
	P [°]	-27,83	-25,71	-27,19	-27,15	-25,86	-27,88	-27,46	-13,22
	Y [°]	139,82	137,84	138,98	138,54	138,12	139,11	138,64	140,30
Senzor #4	**X** $[m]$	0,00	-0,62	-0,50	-0,60	-0,66	-0,47	-0,62	-0,13
	Y $[m]$	4,10	3,93	3,98	3,99	4,08	3,98	4,04	4,01
	Z $[m]$	1,55	1,49	1,49	1,55	1,46	1,48	1,57	1,47
	R [°]	0,00	1,02	1,06	0,76	0,52	2,10	1,45	5,60
	P [°]	-14,95	-15,49	-14,04	-15,48	-14,28	-13,77	-15,50	-17,88
	Y [°]	-43,33	-36,58	-38,10	-37,38	-37,99	-38,42	-37,84	-39,31

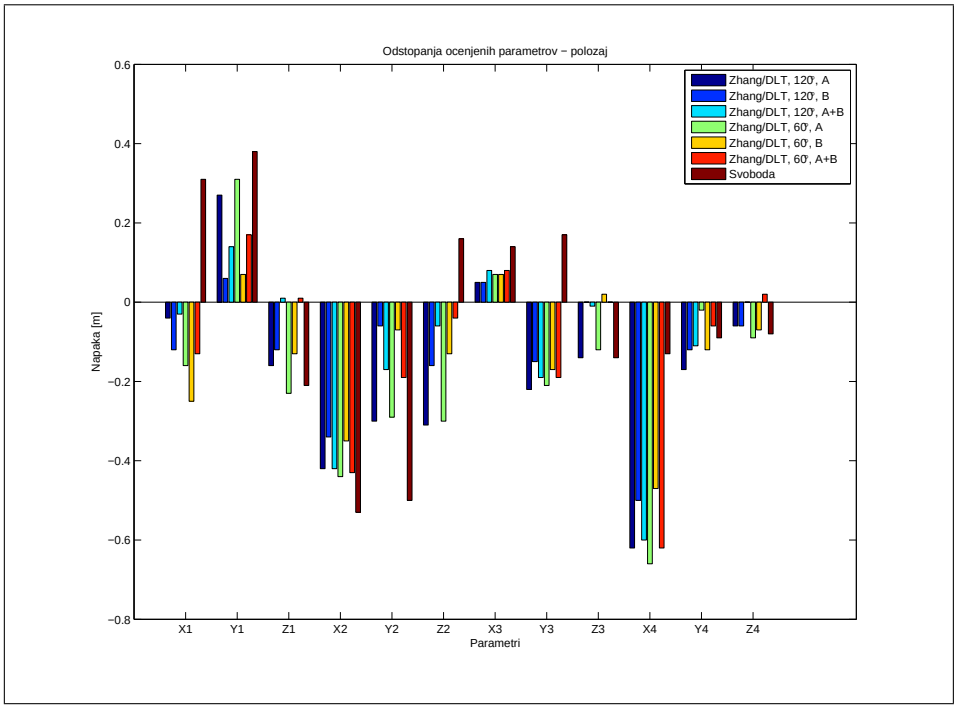
Tabela 4.5: *Parametri senzorjev*

	Senzor #1	**Senzor #2**	**Senzor #3**	**Senzor #4**	**Skupno**
μ $[px]$	4.35	3.78	4.10	3.72	3.98
σ $[px]$	3.78	2.55	2.56	2.08	2.82

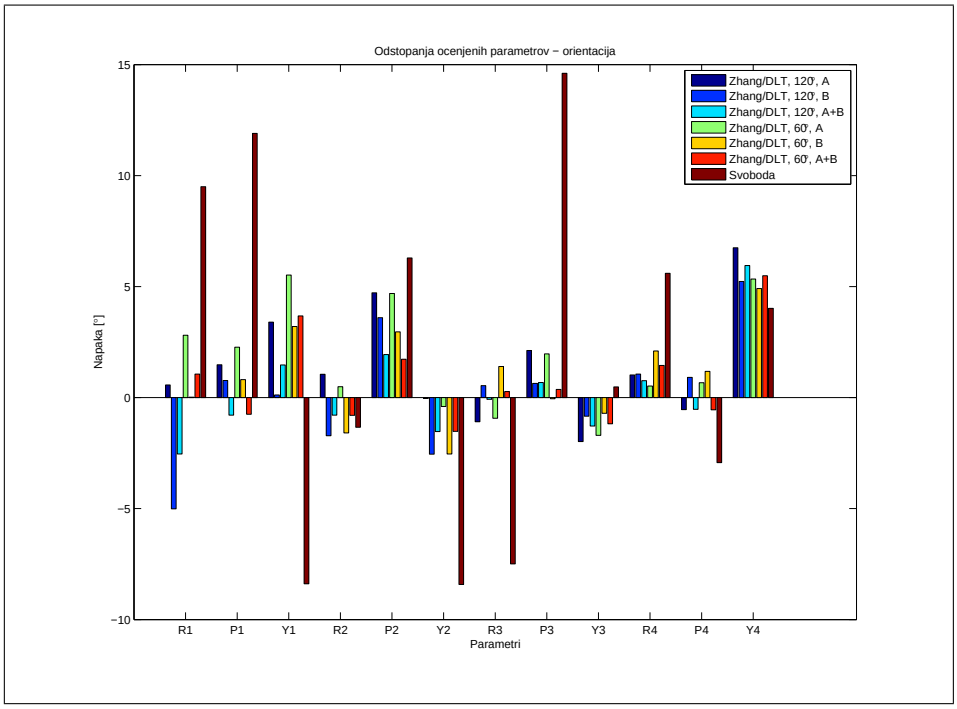
Tabela 4.6: *Napaka reprojekcije pri metodi T. Svobode; srednja vrednost (μ) in standardna deviacija (σ)*

lahko enostavno izvedemo poravnavo koordinatnega sistema z izbiro štirih točk (Slika 4.14).

Po poravnavi koordinatnega sistema (Slika 4.15) lahko izračunamo napako rekonstrukcije, in sicer kot razdaljo med znanimi koordinatami kontrolnih točk in koordinatami, določenimi v postopku kalibracije (Tabela 4.4). Iz matrik kamer lahko izračunamo zunanje parametre senzorjev; ti so prikazani v Tabeli 4.5.

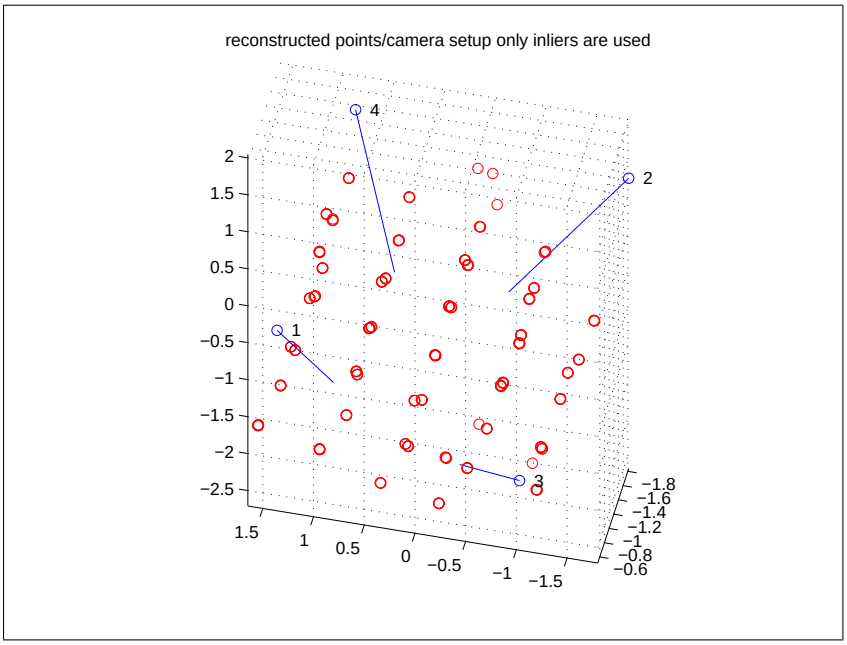


(a)

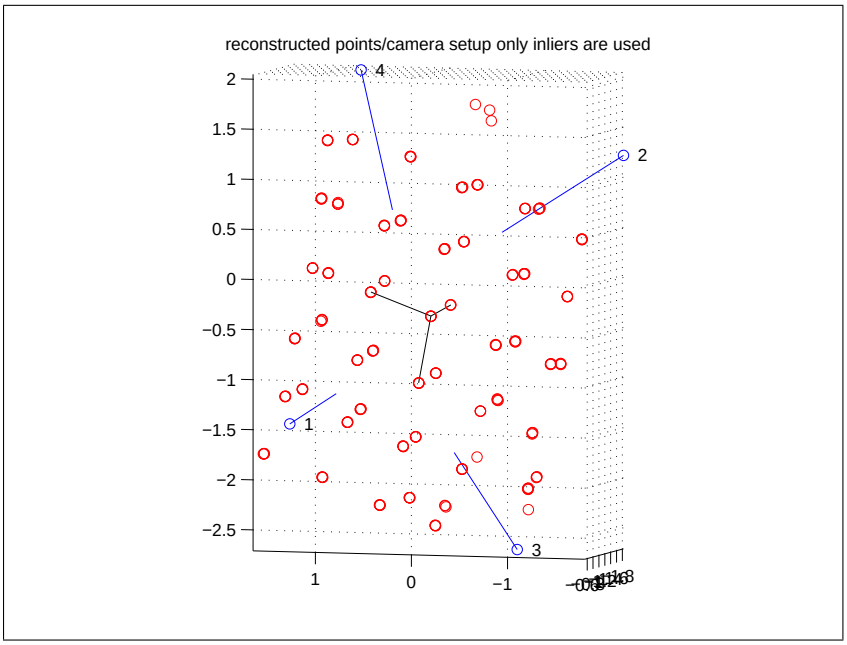


(b)

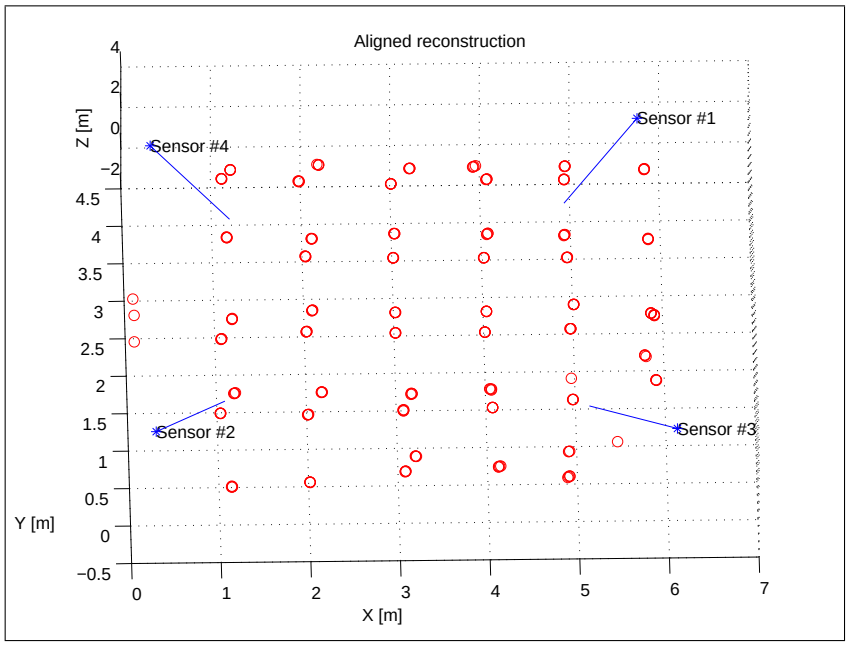
Slika 4.12: *Odstopanja ocenjenih parametrov; položaji (a) in orientacije (b)*



Slika 4.13: *Neporavnani rezultat uspešne kalibracije z metodo T. Svobode*



Slika 4.14: *Izbira štirih točk za poravnavo koordinatnega sistema*



Slika 4.15: *Poravnani koordinatni sistem*

Poglavje 5

Diskusija in sklep

Iz rezultatov eksperimentov, opisanih v četrtem poglavju, lahko zaključimo, da je kalibracija sistema Ubisense z metodami za kalibracijo kamer možna, vendar z določenimi kompromisi in omejitvami; to se nanaša predvsem na uporabo avtokalibracijske metode T. Svobode.

Z uporabo avtokalibracije smo se želeli predvsem izogniti vsakršnemu merjenju koordinat v prostoru, s čimer bi postopek kalibracije približali alternativnemu postopku, ki so ga za sisteme, ki delujejo izključno na osnovi meritev časov prihoda, predlagali Hol, Schön in Gustafsson (Podpoglavje 2.7).

Kot smo lahko videli v Podpoglavjih 4.3 in 4.4, imamo pri uporabi avtokalibracijske metode T. Svobode težave predvsem zaradi relativno visokega nivoja šuma v meritvah kotov prihoda in posledično tudi v slikovnih koordinatah projeciranih točk. Poleg tega zaradi zakrivanja in odbojev pogosto pride do izrazitega odstopanja v kotih prihoda na vsaj enem od senzorjev; posledično meritve s senzorjev v danem časovnem oknu niso konsistentne in zato jih postopek avtokalibracije med validacijo zavrže.

Metoda T. Svobode ne omogoča ločenega ocenjevanja zunanjih in notranjih parametrov kamer, pač pa določi celotno matriko kamere; ocenjeni notranji parametri navideznih kamer tako sicer minimizirajo napako reprojekcije, vendar se razlikujejo od tistih, s katerimi smo projecirali točke v navidezno slikovno ravnino. Posledično se tudi ocenjeni zunanji parametri navideznih kamer lahko precej razlikujejo od dejanskih parametrov senzorjev.

Izkazalo se je, da je v splošnem problematična tudi poravnava koordinatnega sistema. Metoda je bila razvita predvsem za kalibracijo velikega števila kamer

v okoljih navidezne resničnosti; v tem primeru je zunanji koordinatni sistem pogosto določen s položajem kamer in ne obratno. Koordinatni sistem tako lahko poravnamo s pomočjo npr. treh fiksno nameščenih kamer, ki določajo ravnino. V primeru kalibracije sistema Ubisense to ni mogoče; program *Location Engine Config* zahteva, da so parametri senzorjev izraženi v metrih, zato je poravnava z zunanjim koordinatnim sistemom nujna.

Za poravnavo moramo določiti premik, rotacijo ter skaliranje v vseh treh smereh, kar najlažje izvedemo s poznavanjem koordinat štirih točk v prostoru. V Podpoglavju 4.5 poravnava zato ni bila problematična, v Podpoglavjih 4.3 in 4.4 pa bi bila. V praksi poravnavo koordinatnega sistema lahko izvedemo tako, da si pomagamo s tremi vogali igrišča; značko v vseh treh vogalih postavimo na tla, v enem vogalu pa še en meter nad tlemi. Problem seveda nastopi, če postopek avtokalibracije med validacijo zavrže meritve s teh kontrolnih točk.

Tako se na koncu izkaže, da za poravnavo vseeno potrebujemo (vsaj) štiri kontrolne točke; zato je verjetno enostavneje, če določimo še dve dodatni kontrolni točki in izvedemo kalibracijo z Zhangovo metodo ali z metodo DLT. Ti dve metodi sta se izkazali za precej robustnejši in glede na rezultate izvedenih eksperimentov predstavljata boljšo izbiro za kalibracijo sistema Ubisense.

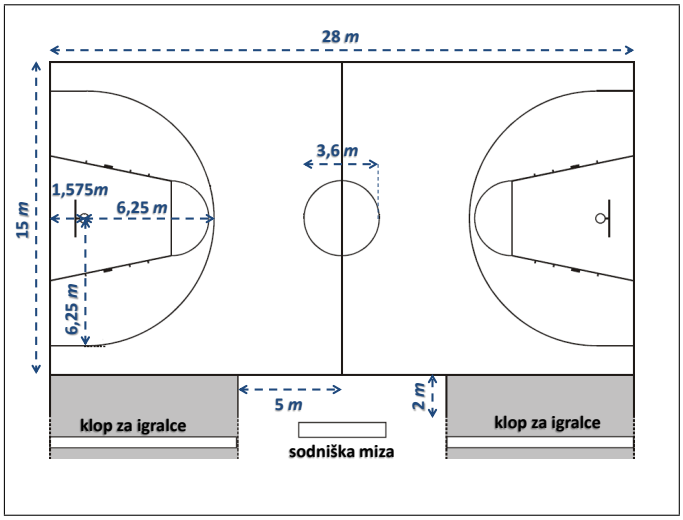
Uporaba Zhangove metode ima še eno veliko prednost — ker za kalibracijo potrebujemo koplanarne točke, lahko vse meritve izvedemo tako, da značko postavimo na tla. Pri izbiri kontrolnih točk si pomagamo z robovi igrišča ter oznakami na igrišču (primer oznak na košarkarskem igrišču je prikazan na Sliki 5.1); v primeru, da poleg sistema Ubisense želimo uporabiti tudi video sistem za sledenje, lahko z istimi kontrolnimi točkami kalibriramo tudi kamere video sistema.

5.1 Predlagani praktični postopek kalibracije

Predlagani alternativni postopek kalibracije sistema Ubisense je tako sledeč:

1. S pomočjo oznak na igrišču določimo primerne kontrolne točke na površini igrišča.

2. Značko postavimo na vsako od kontrolnih točk; s programom *Location Engine Config* v vsaki kontrolni točki zajamemo več (npr. 20) dogodkov.



Slika 5.1: *Oznake na košarkarskem igrišču (vir: [10])*

3. Po koncu zajema meritev vse dogodke izvozimo v .xcm datoteko.

4. Iz .xcm datoteke preberemo podatke o kotih prihoda; za vsako kontrolno točko izračunamo ustrezne mediane kotov prihoda.

5. Izberemo ustrezne notranje parametre navideznih kamer (npr. slikovna ravnina 600 × 600 pikslov z optičnim središčem v sredini in vidnim poljem 120°) in projeciramo kote prihoda v navidezne slikovne ravnine.

6. Z dobljenimi slikami izvedemo postopek kalibracije navideznih kamer z orodjem *Camera Calibration Toolbox for Matlab*.

7. Dobljene zunanje parametre navideznih kamer — položaj in orientacijo — vnesemo v program *Location Engine Config*.

8. Preostane nam še kalibracija kablov, ki jo izvedemo s programom *Location Engine Config*; za kontrolno točko, ki jo postopek potrebuje, lahko uporabimo kar eno od kontrolnih točk, ki smo jih uporabili za kalibracijo zunanjih parametrov.

Kalibracijo kablov še vedno izvedemo s postopkom, ki je vgrajen v Ubisensov program *Location Engine Config*. Vzrok je predvsem v tem, da za kalibracijo kablov potrebujemo meritve razlik v časih prihoda; način zapisa le-teh v .xcm datoteki pa ni nikjer dokumentiran, zato nam v praksi te meritve niso dostopne.

Po drugi strani pa postopek kalibracije kablov ni več problematičen, saj smo položaje senzorjev predhodno določili s kalibracijo navideznih kamer.

5.2 Nadaljnje delo

Nadaljnje delo bo usmerjeno predvsem v preizkus predlaganega postopka za kalibracijo realnih postavitev sistema Ubisense v športnih dvoranah, s poudarkom na oceni napake lociranja in primerjavi z napako ocene v primeru kalibriracije z vgrajenim postopkom. Eno od možnih področij za nadaljnje delo je tudi določanje optimalnega števila kontrolnih točk.

V nadaljevanju bi lahko tudi poskusili razvozlati točen pomen podatkov, povezanih z razliko v časih prihoda in dodali postopek za kalibracijo kablov na osnovi že zajetih meritev. Alternativna možnost uporabe meritev časov prihoda bi bila ocena razdalj med senzorji in med točkami (podobno, kot je to storjeno v [13]), kar bi nam omogočilo enostavno poravnavo koordinatnega sistema v primeru uporabe avtokalibracijske metode T. Svobode. V tem primeru bi morali za časovno sinhronizacijo vedno uporabiti enako topologijo in iste kable; slednje bi morali enkrat kalibrirati z vgrajenim postopkom in v nadzorovanih razmerah.

Zlasti zanimiva pa bi bila možnost zajema surovih podatkov s senzorjev v realnem času; to bi omogočilo samodejen zajem primernega števila meritev in enostavno uporabo več značk hkrati, obenem pa bi za kalibracijo neustrezne meritve lahko zavrgli takoj po zajemu. S tem bi še dodatno poenostavili in pohitrili postopek kalibracije; izvedba te možnosti pa bi bila dejansko možna le s sodelovanjem s strani podjetja Ubisense oziroma z pridobitvijo statusa njihovega razvojnega partnerja.

Literatura

[1] SandLinks. http://www.sandlinks.com. Zadnji dostop: avgust 2010.

[2] TimeDomain. http://www.timedomain.com. Zadnji dostop: avgust 2010.

[3] Ubisense. http://www.ubisense.com. Zadnji dostop: avgust 2010.

[4] Ubisense Research Network — Individual Sensor Raw Data. http://www.ubisense.org/viewtopic.php?f=7\&t=16. Zadnji dostop: avgust 2010.

[5] Ubisense Research Network — Version 2 Software Compatibility Matrix Request. http://www.ubisense.org/viewtopic.php?t=191. Zadnji dostop: avgust 2010.

[6] EU approves general usage of Ubisense ultra-wideband (UWB) active RFID location system. http://www.morerfid.com/details.php?subdetail=Report&action=details&report_id=2660&display=RFID, februar 2007. Zadnji dostop: avgust 2010.

[7] Tadej Bajd. *Osnove robotike*. Fakulteta ta elektrotehniko, 2006.

[8] Serge Belongie. Rodrigues' Rotation Formula. MathWorld — A Wolfram Web Resource. http://mathworld.wolfram.com/RodriguesRotationFormula.html. Zadnji dostop: avgust 2010.

[9] Jean-Yves Bouguet. Camera Calibration Toolbox for Matlab. http://www.vision.caltech.edu/bouguetj/calib_doc/index.html. Zadnji dostop: avgust 2010.

[10] FIBA Europe. Official Basketball Rules 2008. Dostopno na http://www.fibaeurope.com/files/%7BA1E7F7B7-BE56-4002-813A-F1BB6F1B2346%7D.pdf, april 2008.

[11] F. Fleuret, J. Berclaz, R. Lengagne, P. Fua. Multi-Camera People Tracking with a Probabilistic Occupancy Map. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 30, No. 2, str. 267–282, February 2008.

[12] H. Hatze. High-precision three-dimensional photogrammetric calibration and object space reconstruction using a modified DLT-approach. *J. Biomechanics*, Vol. 21, str. 533–538, 1988.

[13] Jeroen D. Hol, Thomas B. Schön, Fredrik Gustafsson. Ultra-Wideband Calibration for Indoor Positioning. Dostopno na `http://www.control.isy.liu.se/%7Eschon/Publications/HolSG2010b.pdf`.

[14] R. E. Kalman. A new approach to linear filtering and prediction problems. *Trans. ASME, J. Basic Engineering*, Vol. 82, str. 34–45, 1960.

[15] Marjan Kodelja. Tehnologija ultraširokega spektra. `http://www.mojmikro.si/v_srediscu/tehnologije/tehnologija_ultrasirokega_spektra`, 2007. Zadnji dostop: avgust 2010.

[16] Tomaž Korošec. *Možnost lociranja v mobilnih sistemih*. Seminarska naloga pri podiplomskem predmetu mobilne komunikacije, Univerza v Ljubljani, Fakulteta za elektrotehniko, junij 2006. Dostopno na `http://www.lkn.fe.uni-lj.si/publikacije/Seminarji_06/mobilne/t_korosec.pdf`.

[17] Stanislav Kovačič. Strojni vid: Modeliranje kamere. Dostopno na `http://vision.fe.uni-lj.si/classes/Sv/Sv-2009-2010/MV01-14-November-2009-Dlt.ppt`, 2009.

[18] M. Kristan, J. Perš, M. Perše, M. Bon, S. Kovačič. Multiple interacting targets tracking with application to team sports. *4th International Symposium on Image and Signal Processing and Analysis ISPA*, str. 322–327, september 2005.

[19] M. Kristan, J. Perš, M. Perše., S. Kovačič. Closed-world tracking of multiple interacting targets for indoor-sports applications. *Computer Vision and Image Understanding*, Vol. 113, No. 5, str. 598–611, 2009.

[20] Young-Hoo Kwon. DLT Method. `http://www.kwon3d.com/theory/dlt/dlt.html`. Zadnji dostop: avgust 2010.

[21] Young-Hoo Kwon. Modified DLT. `http://www.kwon3d.com/theory/dlt/mdlt.html`. Zadnji dostop: avgust 2010.

[22] Ubisense Ltd. *Location Engine config manual*, 2007.

[23] MathWorks. MATLAB — The Language Of Technical Computing. `http://www.mathworks.com/products/matlab`. Zadnji dostop: avgust 2010.

[24] J. Perš, S. Kovačič. A System for Tracking Players in Sports Games by Computer Vision. *Electrotechnical Review — Journal for Electrical Engineering and Computer Science*, Vol. 67, No. 5, str. 281–288, 2000.

[25] Tomáš Svoboda. Multi-Camera Self-Calibration. `http://cmp.felk.cvut.cz/~svoboda/SelfCal/index.html`. Zadnji dostop: avgust 2010.

[26] Tomáš Svoboda, Daniel Martinec, Tomáš Pajdla. A Convenient Multi-Camera Self-Calibration for Virtual Environments. *PRESENCE: Teleoperators and Virtual Environments*, Vol. 14, No. 4, str. 407–422, avgust 2005.

[27] Robert Szewczyk. UWB: Technology and implications for sensor networks. Dostopno na `www.cs.berkeley.edu/~binetude/NEST/UWB.ppt`, 2004.

[28] John Vince. *Essential mathematics for computer graphics fast.* Essential series. Springer, 2001.

[29] Eric W. Weisstein. Levenberg-Marquardt Method. MathWorld — A Wolfram Web Resource. `http://mathworld.wolfram.com/Levenberg-MarquardtMethod.html`. Zadnji dostop: avgust 2010.

[30] Wikipedia. Multilateration. `http://en.wikipedia.org/wiki/Multilateration`. Zadnji dostop: avgust 2010.

[31] Wikipedia. Triangulation. `http://en.wikipedia.org/wiki/Triangulation`. Zadnji dostop: avgust 2010.

[32] Wikipedia. Trilateration. `http://en.wikipedia.org/wiki/Trilateration`. Zadnji dostop: avgust 2010.

[33] Wikipedia. Ultra-wide Band. `http://en.wikipedia.org/wiki/Ultrawide_Band`. Zadnji dostop: avgust 2010.

[34] Zhengyou Zhang. Flexible Camera Calibration by Viewing a Plane from Unknown Orientations. *ICCV99*, str. 666 – 673, 1999.

Dodatek A

XCM datoteka

S programom *Location Engine Config* lahko zajete dogodke izvozimo v .xcm datoteko. Gre za zapis podatkov v XML formatu, primer katerega je prikazan spodaj.

Korenska značka v .xcm datoteki je *events*, znotraj nje pa se za vsak dogodek nahaja značka *value*, v kateri so shranjeni podatki o dogodku.

Identifikacijska oznaka značke (oddajnika), ki je sprožila dogodek, je zapisana v znački *id_* znotraj značke *tag_*. Zaporedna številka časovnega okna, v katerem se je dogodek zgodil, je na voljo v znački *timeslot_*.

V znački *location_* so shranjeni obdelani podati, t. j. položaj značke, kot ga je na podlagi meritev ocenil (kalibrirani) sistem Ubisense, ter njegova napaka. Značka *sensors_* vsebuje seznam senzorjev, ki so sprejeli signal z značke; podatki o posameznem senzorju so shranjeni znotraj značke *value*.

V znački *mac_* znotraj značke *sensor_* je shranjen MAC naslov senzorja, ki služi tudi kot identifikacijska oznaka senzorja. Zunanji parametri senzorja (položaj in orientacija, ki ju določimo tekom kalibracije) so shranjeni v znački *position_*, parametri kabla pa v znački *offsets_*.

Surove meritve s senzorja so na voljo v znački *radar_*. V značkah *azimuth_* in *elevation_* sta shranjeni meritvi kotov prihoda (v radianih). Značke *ppo_*, *event1_* in *event2_* vsebujejo informacijo o razlikah v časih prihoda, vendar dejanski pomen številskih vrednosti znotraj njih ni nikjer razložen. V značkah *raw_* in *demod_* je zapisana informacija o moči prejetega signala.

Značka *used_for_location_* ima v primeru, da je sistem Ubisense pri določanju položaja značke uporabil meritve s pripadajočega senzorja, vrednost *T*, v

nasprotnem primeru pa vrednost *F*. V znački *distance_* pa je shranjena ocenjena razdalja med senzorjem in značko (v metrih).

V XCM datoteko lahko izvozimo največ 10000 dogodkov; zaradi obširnosti oblike zapisa je v tem primeru velikost datoteke 42 MB.

primer.xcm: en dogodek v celici s štirimi senzorji

```
<events>
  <value>
    <instance_>0</instance_>
    <tag_>
      <id_>335574327</id_>
    </tag_>
    <timeslot_>294868</timeslot_>
    <location_>
      <flags_>301</flags_>
      <x_>5.60678816e+000</x_>
      <y_>1.37437940e+000</y_>
      <z_>7.79330492e-001</z_>
      <gdop_>0.00000000e+000</gdop_>
      <error_>2.29898006e-001</error_>
    </location_>
    <sensors_>
      <value>
        <sensor_>
          <mac_>0:11:ce:0:13:4d</mac_>
        </sensor_>
        <position_>
          <x_>1.00000000e+000</x_>
          <y_>5.00000000e-001</y_>
          <z_>1.86000001e+000</z_>
          <yaw_>1.12051976e+000</yaw_>
          <pitch_>8.46307911e-003</pitch_>
        </position_>
        <offsets_>
          <zero_offset_>1.46800000e+003</zero_offset_>
          <cable_offset_>0.00000000e+000</cable_offset_>
        </offsets_>
        <flags_>65</flags_>
```

```
      <radar_>
        <azimuth_>-7.37203121e-001</azimuth_>
        <elevation_>-1.62585035e-001</elevation_>
        <ppo_>1.49200000e+003</ppo_>
        <event1_>4.59000000e+002</event1_>
        <event2_>1.20600000e+003</event2_>
        <code_>0.00000000e+000</code_>
        <raw_>2.24576611e+003</raw_>
        <demod_>1.20213848e+004</demod_>
      </radar_>
      <used_for_location_>T</used_for_location_>
      <distance_>2.83830128e+001</distance_>
    </value>
    <value>
      <sensor_>
        <mac_>0:11:ce:0:13:45</mac_>
      </sensor_>
      <position_>
        <x_>5.21999979e+000</x_>
        <y_>6.62500000e+000</y_>
        <z_>1.76999998e+000</z_>
        <yaw_>-1.89887977e+000</yaw_>
        <pitch_>-2.25713804e-001</pitch_>
      </position_>
      <offsets_>
        <zero_offset_>1.46800000e+003</zero_offset_>
        <cable_offset_>3.80714294e+002</cable_offset_>
      </offsets_>
      <flags_>65</flags_>
      <radar_>
        <azimuth_>5.49382806e-001</azimuth_>
        <elevation_>1.86280441e-002</elevation_>
        <ppo_>1.49400000e+003</ppo_>
        <event1_>8.55000000e+002</event1_>
        <event2_>1.60400000e+003</event2_>
        <code_>0.00000000e+000</code_>
        <raw_>1.42815881e+003</raw_>
        <demod_>1.54117627e+003</demod_>
```

```
    </radar_>
    <used_for_location_>T</used_for_location_>
    <distance_>2.89927425e+001</distance_>
  </value>
  <value>
    <sensor_>
      <mac_>0:11:ce:0:15:db</mac_>
    </sensor_>
    <position_>
      <x_>5.01999998e+000</x_>
      <y_>0.00000000e+000</y_>
      <z_>1.53999996e+000</z_>
      <yaw_>1.85099232e+000</yaw_>
      <pitch_>3.95133793e-002</pitch_>
    </position_>
    <offsets_>
      <zero_offset_>1.46800000e+003</zero_offset_>
      <cable_offset_>7.06695313e+002</cable_offset_>
    </offsets_>
    <flags_>129</flags_>
    <radar_>
      <azimuth_>-0.00000000e+000</azimuth_>
      <elevation_>-0.00000000e+000</elevation_>
      <ppo_>1.49200000e+003</ppo_>
      <event1_>3.74000000e+002</event1_>
      <event2_>1.12200000e+003</event2_>
      <code_>0.00000000e+000</code_>
      <raw_>2.47463135e+003</raw_>
      <demod_>1.71409395e+004</demod_>
    </radar_>
    <used_for_location_>F</used_for_location_>
    <distance_>2.48771000e+001</distance_>
  </value>
  <value>
    <sensor_>
      <mac_>0:11:ce:0:15:dd</mac_>
    </sensor_>
    <position_>
```

```
            <x>2.00999999e+000</x>
            <y>6.03000021e+000</y>
            <z>1.63000000e+000</z>
            <yaw>-1.08114934e+000</yaw>
            <pitch>4.14057262e-002</pitch>
          </position>
          <offsets>
            <zero_offset>1.46800000e+003</zero_offset>
            <cable_offset>8.36726379e+001</cable_offset>
          </offsets>
          <flags>65</flags>
          <radar>
            <azimuth>1.98718116e-001</azimuth>
            <elevation>-1.82556435e-001</elevation>
            <ppo>1.49300000e+003</ppo>
            <event1>5.83000000e+002</event1>
            <event2>1.32900000e+003</event2>
            <code>0.00000000e+000</code>
            <raw>1.75642395e+003</raw>
            <demod>1.62563574e+003</demod>
          </radar>
          <used_for_location>T</used_for_location>
          <distance>3.07569542e+001</distance>
        </value>
      </sensors>
    </value>
</events>
```

Izjava

Izjavljam, da sem diplomsko nalogo izdelal samostojno pod vodstvom mentorja prof. dr. Stanislava Kovačiča. Izkazano pomoč drugih sodelavcev sem v celoti navedel v zahvali.

V Ljubljani, 6. september 2010.

Rok Mandeljc