

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 1543

Monokularna vizualna odometrija za cestovna vozila

Ana Bulović

Zagreb, srpanj 2010.

SADRŽAJ

1. Uvod	1
2. Geometrija dvaju pogleda	3
2.1. Epipolarna geometrija	3
2.2. Kalibracija kamera	4
2.3. Značenje esencijalne matrice i epipolarno ograničenje	5
2.4. Svojstva esencijalne matrice	6
2.4.1. Singularna dekompozicija esencijalne matrice	7
2.5. Procjena esencijalne matrice	8
2.5.1. Procjena esencijalne matrice koristeći algoritam za osam točaka	8
2.5.2. Procjena esencijalne matrice koristeći algoritam za pet točaka	9
3. Vizualna odometrija	10
3.1. Pogreške pri obradi slike	11
3.2. Odstranjivanje vanpopulacijskih značajki (outliera)	11
3.3. Gradijentna optimizacija strukture i kretanja	12
3.3.1. Geometrijsko objašnjenje	12
3.3.2. Mjera pogreške procjene 3D položaja točke	13
3.3.3. Programska implementacija	16
3.4. Izračun ukupne rotacije i translacije	16
3.5. Korištenje ograničenja gibanja cestovnog vozila	17
3.5.1. Poseban oblik esencijalne matrice uz ograničenja gibanja . . .	18
3.5.2. Odstranjivanje vanpopulacijskih značajki	19
4. Programska implementacija	20
4.1. Korišteni programski paketi	20
4.2. Glavna aplikacija	21
4.2.1. Vađenje značajki	21

4.2.2. Procjena esencijalne matrice	22
4.3. Izračun položaja kamere	23
5. Testiranje programskog rješenja	25
5.1. Alterniranje broja značajki kojima se inicijalizira praćenje	25
5.2. Uklanjanje vanpopulacijskih značajki korištenjem histograma vrijednosti kuteva	26
6. Moguća poboljšanja	27
6.1. Triangulacija točaka	27
6.2. Ostala moguća poboljšanja	28
7. Zaključak	29
Literatura	30

1. Uvod

Već dugo među ljudima postoji interes za automatizacijom posla. Jedan od pristupa ovom problemu su autonomni inteligentni sustavi koji mogu raditi duže vrijeme bez ljudske kontrole. Primjer ovakvih sustava su tvornički roboti, autonomna kolica za nepokretne, mnoga pomagala u kućanstvu, a mnoge su i primjene u istraživanju svemira gdje je ljudsko upravljanje znatno otežano.

Da bi ti sustavi mogli raditi bez ljudske kontrole, moraju moći na neki način razumjeti okolinu u kojoj se nalaze i najčešće kreću. Odometrija objašnjava kako iskoristiti ulazne podatke za procjenu položaja i kretanja objekta. Jedan od načina da se to ostvari, a ujedno je invarijantan na vrstu pogona, je vizualna odometrija koja vrši procjenu pozicije i usmjerenja kamere iz slijeda slika s njenog izlaza.

Područje od interesa u ovom radu je monokularna vizualna odometrija za cestovna vozila [7]. Mogućnost primjene vizualne odometrije u procjeni kretanja cestovnog vozila, osim već navedenih je predviđanje prostora i kretanja pojedinih objekata čiji je položaj vezan uz cestu (poput cestovnih znakova).

Postupak kojim je ostvarena vizualna odometrija za cestovna vozila obuhvaća vađenje značajki iz okvira u videu, određivanje korespondentnih značajki te procjena esencijalne matrice pomoću istih. Nakon toga se procjenjuju položaji 3D točaka na temelju značajki i esencijalne matrice.

Za vađenje i praćenje značajki je korišten prethodno razvijena implementacija Kanade-Lucas-Tomassi algoritma iz ljske računalnog vida cvsh. Za procjenu esencijalne matrice pomoću pet točaka je korištena implementacija Noaha Snavelyja [2] koja je dostupna pod GNU dozvolom. Za ponovno računanje esencijalne matrice uz optimizacijski postupak je korišten SBA [5](Sparse Bundle Adjustment).

Kao eksperiment je implementiran ponešto drukčiji pristup koji se zasniva na članku Davidea Scaramuzze koji na bolji način koristi ograničenja kretanja cestovnog vozila, točnije automobila.

Sadržaj rada je sljedeći. U drugom poglavlju je objašnjen pojam geometrije dvaju pogleda te njenog dijela od interesa u ovom radu - epipolarne geometrije. Opisani

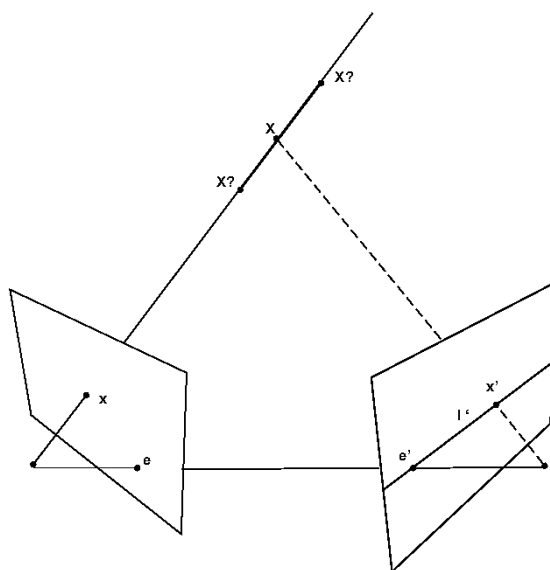
su postupci rekonstrukcije kretanja iz korespondentnih značajki u okvirima videa. U trećem poglavlju je opisana vizualna odometrija te svi potrebni koraci za konačnu procjenu kretanja poput uklanjanja vanpopulacijskih značajki te izračuna ukupne rotacije i translacije za svaki okvir. U četvrtom poglavlju je opisana programska implementacija i već gotovi paketi koji su korišteni. U petom poglavlju su opisani neki rezultati testiranja programskog rješenja, a u šestom su popisana moguća poboljšanja trenutne verzije implementacije.

2. Geometrija dvaju pogleda

Uslika li se ista scena iz dva različita pogleda, veze među 3D pozicijama točaka i njihovim projekcijama na ravnine slika moguće je opisati metodama geometrije dvaju pogleda. Za 3D interpretaciju scene na temelju korespondentnih točaka u dvjema slikama scene potrebno je prvo procijeniti epipolarnu geometriju.

2.1. Epipolarna geometrija

Veze među korespondentnim točkama scene epipolarna geometrija opisuje uz pretpostavku da kamere ne koriste leće za fokus svjetla.



Slika 2.1: Točka prostora X projicirana na dvije ravnine

Iz slike 2.1 moguće je vidjeti da položaj projekcije točke na jednoj slici zaista postavlja ograničenje na mogućnost pojave projekcije te iste točke na drugoj slici. Položaji dviju kamera te položaj točke u 3D prostoru se nalaze na takozvanoj epipolarnoj ravnini koja se označava sa Π . Presjek te ravnine sa ravninom slike druge kamere je pravac na kojem će se sigurno naći projekcija točke. Taj pravac se zove epipolarni

pravac i na slici je označen sa l' . Točke presjeka ravnina i pravca koji spaja položaje kamera nazivaju se epipoli i na slici su označeni sa e i e' . Pretpostavi li se da postoji takva homografija H za koju vrijedi $Hx = x'$ za svaki $X \in \Pi$ tada je epipolarni pravac l' definiran točkama Hx i e' . To je moguće zapisati kao:

$$l' = p(e', Hx)$$

$$l' = e'_{[x]} Hx$$

$$l' = Fx$$

Matrica F se naziva fundamentalna matrica i za svaki par korespondentnih točaka opisuje pravac Fx na kom mora ležati točka x' . Iz opisanog odnosa je očito da za svaki par takvih točaka mora vrijediti sljedeća jednačba:

$$x'^T Fx = 0 \quad (2.1)$$

2.2. Kalibracija kamera

Već je rečeno da epipolarna geometrija pretpostavlja da nema distorzija prostora kao posljedica korištenja leća u kameri. Fundamentalna matrica opisuje odnose projekcija točaka u realnom slučaju u kom je prostor iskrivljen iz spomenutog razloga. Kamere je moguće kalibrirati tako da one zadovoljavaju gore navedeni model [12]. Kalibracijom se dolazi do intrinzičnih parametara kamere i kalibracijske matrice K . Jednom kada je poznata kalibracijska matrica, moguće je normalizirati koordinatni sustav slike. Normalizirane koordinate točke x se dobiju kao:

$$\hat{x} = K^{-1}x \quad (2.2)$$

Vezu koju opisuje fundamentalna matrica moguće je predstaviti drugom matricom koja uzima u obzir ograničenje na model kamere. Ta matrica se naziva esencijalna matrica te se označava sa E . Za normalizirane koordinate $\hat{x}$ i $\hat{x}'$ vrijedi:

$$\hat{x}'^T E \hat{x} = 0 \quad (2.3)$$

Iz jednačbi 2.2 i 2.3 moguće je doći do odnosa fundamentalne i esencijalne matrice:

$$F = K'^{-T} E K^{-1} \quad (2.4)$$

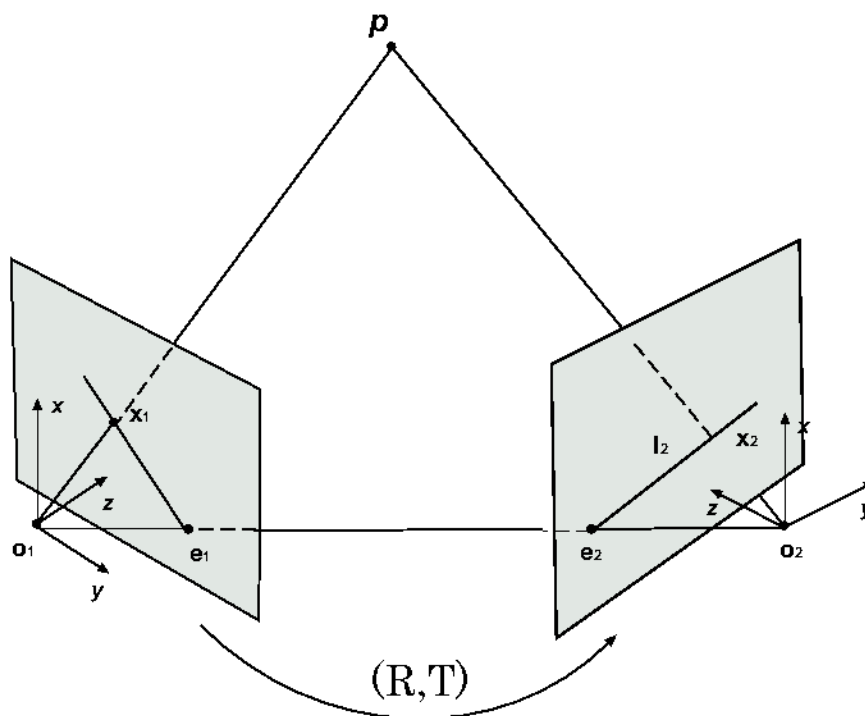
2.3. Značenje esencijalne matrice i epipolarno ograničenje

Svaka kamera ima referentni okvir sa izvištem u optičkom centru o te sa z osi poravnatom sa optičkom osi. Odnos točaka u dva takva okvira može se opisati transformacijama krutog tijela (uz pretpostavku da je scena statična). Moguće je uzeti jedan referentni okvir kao stvarni svijet, dok je drugi moguće opisati u odnosu na njega euklidskom transformacijom $g = (R, T)$.

Koordinate projekcija 3D točaka u koordinatni sustav kamere i izvorne koordinate točke su povezane na sljedeći način:

$$\lambda x = \Pi_0 X \quad (2.5)$$

gdje je $\Pi_0 = [I, 0]$ što znači da se razlikuju samo za dubinski faktor $\lambda \in \mathbb{R}_+$.



Slika 2.2: Odnos koordinata u sustavima dviju kamera opisan pomoću transformacija krutog tijela

Neka su koordinate točke p sa slike 2.2 u koordinatnom sustavu prve kamere označene sa $X_1 \in \mathbb{R}^3$ te u koordinatnom sustavu druge kamere sa $X_2 \in \mathbb{R}^3$. Njihova veza može se opisati sljedećom jednažbom:

$$X_2 = RX_1 + T$$

S obzirom na jednažbu 2.5, ovaj izraz se može zapisati kao:

$$\lambda_2 x_2 = R\lambda_1 x_1 + T$$

Da bi eliminirali nepoznate faktore λ_i , potrebno je obe strane jednadžbe pomnožiti sa $\hat{T}$, čime se dobije:

$$\lambda_2 \hat{T} x_2 = \hat{T} R \lambda_1 x_1$$

Budući da je vektor $\hat{T} x_2 = T \times x_2$ okomit na vektor x_2 , skalarni umnožak $\langle x_2, \hat{T} x_2 \rangle = x_2^T \hat{T} x_2$ je nula. Ako se izraz pomnoži sa lijeve strane sa x_2^T to daje jednakost:

$$x_2^T \hat{T} R \lambda_1 x_1 = 0$$

Kako je $\lambda_i > 0$, jasno je da vrijedi epipolarno ograničenje [3]:

$$\langle x_2, T \times R x_1 \rangle = 0 \quad (2.6)$$

To se još može zapisati kao:

$$x_2^T \hat{T} R x_1 = 0$$

Podatke o relativnoj orijentaciji R i relativnom položaju T opisuje esencijalna matrica E . Iz epipolarnog ograničenja i jednadžbe 2.3 vidi se da se ona može zapisati kao:

$$E = \hat{T} R \in \mathbb{R}^{3 \times 3}$$

Kako pravci $o_1 \vec{o}_2$, $o_1 \vec{p}$ i $o_2 \vec{p}$ se nalaze na istoj ravnini, njihov trostruki umnožak ¹ iznosi nula budući da on mjeri sadržani volumen. Kako to vrijedi za 3D koordinate, tako vrijedi i za koordinate u normaliziranim sustavima slika kamera jer je razlika tih i 3D koordinata jedino u skalarnom faktoru λ_i . Gledano iz položaja druge kamere, $R x_1$ je smjer pravca $o_1 \vec{p}$ dok je T zapravo vektor $o_2 \vec{o}_1$.

2.4. Svojstva esencijalne matrice

Esencijalne matrice pripadaju skupu matrica u $\mathbb{R}^{3 \times 3}$ koji se zove esencijalni prostor i označava sa ϵ :

$$\epsilon \doteq \{\hat{T} R | R \in SO(3), T \in \mathbb{R}^3\} \subset \mathbb{R}^{3 \times 3}$$

gdje je $SO(3)$ specijalna ortogonalna grupa matrica što znači da imaju svojstvo očuvanja unutarnjeg produkta. Dakle za ortogonalnu matricu A vrijedit će:

$$\langle Ax, Ay \rangle = \langle x, y \rangle, \quad \forall x, y \in \mathbb{R}^n$$

¹Trostruki umnožak vektora $a(b \times c)$ je definiran kao skalarni umnožak jednog vektora sa vektorskim umnoškom preostala dva.

Za sve matrice $M \in SO(n)$ također vrijedi da im je determinanta jednaka $+1$. Nadalje, da bi matrica bila esencijalna, potrebno je da ima singularnu dekompoziciju $(SVD)E = U\Sigma V^T$ gdje je

$$\Sigma = \text{diag}\{\sigma, \sigma, 0\}$$

SVD ovdje označava singularnu dekompoziciju matrice (Singular Value Decomposition).

2.4.1. Singularna dekompozicija esencijalne matrice

Singularna dekompozicija matrice je često korištena faktorizacija matrice jer je njome moguće doći do osnovnih svojstava matrice kao što su rang, doseg i nul-prostor matrice, a koristi se u mnogim linearnim algebarskim problemima poput izračuna inverza matrice te linearne procjene najmanjih kvadrata. Singularnu dekompoziciju je moguće provesti za realne i kompleksne matrice. Za matricu $M \in \mathbb{R}^{m \times n}$ ranga p vrijedi:

- $\exists U \in \mathbb{R}^{m \times p}$ čiji stupci su ortonormalni
- $\exists V \in \mathbb{R}^{n \times p}$ čiji stupci su ortonormalni
- $\exists \Sigma \in \mathbb{R}^{p \times p}$, $\Sigma = \text{diag}\{\sigma_1, \sigma_2, \dots, \sigma_p\}$ za koje vrijedi $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_p$

takvi da vrijedi $M = U\Sigma V^T$. Matrice U i V nisu jedinstvene s obzirom na M , dok Σ jeste i njene dijagonalne vrijednosti se zovu singularne vrijednosti matrice M .

Geometrijsko objašnjenje

Ako se matrica $M \in \mathbb{R}^{n \times n}$ promatra kao element koji linearno mapira točku x u točku y , tada je moguće promatrati odnos tih dviju točaka na sljedeći način:

- stupci matrice V su ulazni bazni vektori prostora $\mathbb{R}^n$
- stupci matrice U su izlazni bazni vektori prostora $\mathbb{R}^n$ te
- diagonalne vrijednosti matrice Σ su skalarni faktori kojima se mapiraju koordinate ulazne točke x u izlaznu točku y .

Ako se SVD primijeni na esencijalnu matricu $E \in \mathbb{R}^{3 \times 3}$, uvjeti su nešto stroži. Da bi matrica E bila esencijalna matrica, njena singularna dekompozicija $(SVD)E = U\Sigma V^T$ mora biti takva da vrijedi:

$$\Sigma = \text{diag}\{\sigma, \sigma, 0\}$$

gdje je $\sigma \in \mathbb{R}_+$, a $U, V \in SO(3)$. Kako se standardnim postupkom singularne dekompozicije ne poštuje ograničenje da matrice U i V moraju biti iz specijalne ortogonalne

grupe, a njihova vrijednost nije jedinstveno određena matricom E , moguće je doći do više rješenja za problem dekompozicije esencijalne matrice. Od mogućih rješenja ispravno se pronalazi eliminacijom onih rješenja u kojima se točka ne nalazi ispred obje kamere.

2.5. Procjena esencijalne matrice

Prethodno je pokazano da slike korespondentnih točaka u sustavima kamera povezuje epipolarno ograničenje. Ako se raspolaze dovoljnim brojem korespondentnih točaka, moguće je, koristeći epipolarno ograničenje, iz njihovog odnosa rekonstruirati esencijalnu matricu, a iz nje, naknadno, relativnu rotaciju i translaciju među kamerama.

2.5.1. Procjena esencijalne matrice koristeći algoritam za osam točaka

Prethodno je izrečen uvjet (2.3) koji mora zadovoljavati esencijalna matrica. Zbog jednostavnije notacije u daljnjem pojašnjenju algoritma, uvjet ćemo zapisati kao:

$$y'^T E y = 0 \quad (2.7)$$

gdje su y' i y normalizirane homogene koordinate točaka u dva pogleda. Označe li se elementi pojedinih matrica i vektora kao

$$y = \begin{pmatrix} y_1 \\ y_2 \\ 1 \end{pmatrix}, y' = \begin{pmatrix} y'_1 \\ y'_2 \\ 1 \end{pmatrix}, E = \begin{pmatrix} e_{11} & e_{12} & e_{13} \\ e_{21} & e_{22} & e_{23} \\ e_{31} & e_{32} & e_{33} \end{pmatrix} \quad (2.8)$$

tada se izraz 2.7 može raspisati kao:

$$y'_1 y_1 e_{11} + y'_1 y_2 e_{12} + y'_1 e_{13} + y'_2 y_1 e_{21} + y'_2 y_2 e_{22} + y'_2 e_{23} + y_1 e_{31} + y_2 e_{32} + e_{33} = 0 \quad (2.9)$$

Iz ovog zapisa jasno je da komponente vektora y i y' možemo izdvojiti kao zasebnu 3×3 matricu $y'y^T$. Zapišu li se te dvije matrice kao 9-dimenzionalni vektori, dobiva se sljedeći zapis:

$$\tilde{y} = \begin{pmatrix} y'_1 y_1 & y'_1 y_2 & y'_1 & y'_2 y_1 & y'_2 y_2 & y'_2 & y_1 & y_2 & 1 \end{pmatrix}^T$$

$$e = \begin{pmatrix} e_{11} & e_{12} & e_{13} & e_{21} & e_{22} & e_{23} & e_{31} & e_{32} & e_{33} \end{pmatrix}^T$$

Takav zapis vektora y i y' je zapravo njihov Kroneckerov umnožak ($\tilde{y} = y \otimes y'$).

Ako se raspolaze sa k točaka, sve moraju zadovoljavati ograničenje 2.7, a to se može zapisati kao:

$$Y e^T = 0 \quad (2.10)$$

gdje su $\tilde{y}_k$ reci matrice Y . Ako se koristi točno osam linearno nezavisnih vektora $\tilde{y}_k$, tada ova jednadžba ima jedinstveno rješenje. Ako se pak koristi više od osam točaka, što je često slučaj, tada je cilj pronaći takav e koji minimizira 2.10 kada je $\|e\| = 1$. Ovaj sustav jednadžbi se može riješiti dekompozicijom jedinstvene vrijednosti. Primijeni li se ta metoda na matricu Y , dobije se dekompozicija

$$Y = USV^T$$

gdje su U i V ortonormalne matrice, a S dijagonalna matrica koja sadrži jedinstvene vrijednosti. Ove jedinstvene vrijednosti σ_i su pozitivne vrijednosti u opadajućem poretku. σ_9 će biti jednaka 0 i zadnji stupac matrice V je rješenje (za slučaj osam linearno nezavisnih točaka.)

2.5.2. Procjena esencijalne matrice koristeći algoritam za pet točaka

Matrica koja opisuje transformacije krutog tijela u 3D prostoru ima 6 stupnjeva slobode od kojih 3 otpadaju na rotaciju, 3 na translaciju. Ovo zaključivanje se može primijeniti i na esencijalnu matricu uz jedno ograničenje: zbog toga što ona predstavlja ujedno i projekcijski element, potrebno je oduzeti jedan stupanj slobode koji se onda opisuje razlikom u skalarnom faktoru.

Broj stupnjeva slobode implicira da za procjenu E nije potrebno raspolagati sa osam točaka. Koristeći ovu činjenicu, razvijeni su algoritmi za procjenu matrice koristeći sedam, šest i konačno samo pet točaka [6]. Algoritam s pet točaka ne nudi geometrijski intuitivno objašnjenje kao što je to slučaj s algoritmom za osam točaka i ne oslanja se na rješavanje linearnog sustava jednadžbi. Iako je njegovo rješenje dosta kompliciranije, postoje određene razlike, pa i prednosti u njegovom korištenju:

- Algoritam s 8 točaka daje bezbroj rješenja kada je scena ravninska (od kojih je samo jedno točno), dok algoritam s 5 točaka nema takvih problema
- Postupak za odstranjivanje vanpopulacijskih značajki (koji je optimizacijski dio procjene položaja kamera i bit će pojašnjen kasnije u tekstu) je manje vremenski zahtjevan

Zbog istaknutih razloga, algoritam s 5 točaka je korišten u ovom radu.

3. Vizualna odometrija

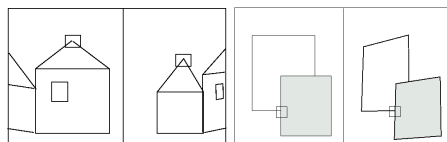
Odometrija je postupak korištenja podataka o kretanju za procjenu relativnog položaja objekta. Kod vizualne odometrije podaci korišteni za takvu procjenu su isključivo slike ili video sekvenca. Takva odometrija nije ovisna o načinu kretanja objekta, već se može upotrijebiti neovisno o pogonu vozila ili robota. Najpoznatija uporaba ove vrste procjene položaja je vjerojatno na Mars Rover-u [11].

Metode korištene u ovom radu su većinom prilagođene za vizualnu odometriju za cestovna vozila. Također, ovo je primjer monokularne odometrije, što implicira da je u postupku procjene kretanja korištena samo jedna kamera koja je bila montirana na prednjem dijelu vozila.

Korišteni postupak je ukratko opisan sljedećim koracima:

1. Pribavljanje slijeda ulaznih slika iz videa
2. Vađenje značajki iz slika
3. Korištenje Kanade-Lucas-Tomasi algoritma za praćenje značajki u slijedu slika
4. Odstranjivanje nepogodnih značajki pomoću RANSAC metode
5. Procjena esencijalne matrice
6. Minimizacija reprojekcijske pogreške
7. Procjena ukupne rotacije i translacije u svakom okviru
8. Procjena kretanja kamere

Svaki od ovih dijelova bit će detaljnije opisan u nastavku rada, osim Kanade-Lucas-Tomasi metode koja nije bila tema ovog rada i za koju je korištena gotova implementacija bez preinaka iz sustava cvsh, a koja se temelji na implementaciji Stana Birchfielda [1].



Slika 3.1: Mogući šumovi kod praćenja značajki na slikama

3.1. Pogreške pri obradi slike

U prethodnom poglavlju opisani algoritmi su pretpostavljali da se radi o ispravnim korespondencijama točaka u sustavima dviju kamera. U realnom svijetu moguće je da će doći do grešaka pri određivanju korespondentnih značajki.

Na slici se mogu vidjeti dvije vrste tipičnih greški koje se pojavljuju pri praćenju značajki. Prva vrsta pogreške jest pogrešna korespondencija značajki u kom je pronađena značajka u jednom okviru povezana sa značajkom na drugom okviru koja pripada drugom elementu scene. Ovakve pogreške narušavaju model scene i potrebno ih je odstraniti pri procjeni kretanja. Druga vrsta moguće pogreške se još naziva i virtualna značajka. Primjer takve značajke bi bio presjek linija dvaju objekata koji se uspješno prati kroz više okvira. Ovakva značajka očigledno ne daje informaciju o stvarnoj prirodi scene i potrebno ju je ukloniti pri procjeni kretanja.

3.2. Odstranjivanje vanpopulacijskih značajki (outliera)

Da bi se mogao što bolje odrediti odnos između dvaju pogleda, potrebno je što je više moguće smanjiti utjecaj šuma na procjenu esencijalne matrice. To se postiže uklanjanjem nepogodnih korespondencija. Takve korespondencije koje ne nude ispravne informacije o strukturi okoline i kretanju još se nazivaju i vanpopulacijske značajke ili outlieri. Korišteni algoritam za tu svrhu zove se RANSAC (RANDOM SAMPLE CONSENSUS). Umjesto da se iz svih podataka računa model i onda uklanjaju točke koje ga ne zadovoljavaju, RANSAC uzima najmanji broj značajki potreban za procjenu modela te računa koliko je postojećih značajki moguće uklopiti u izračunati model.

Da bi se moglo izračunati pripada li određena značajka modelu ili ne, potrebno je postaviti prag (udaljenost) unutar kojeg je značajka dio modela, a izvan kojeg se može klasificirati kao nepogodna. Taj prag se označava sa t (threshold) i potrebno ga je eksperimentalno utvrditi.

Najjednostavniji primjer na kojem se može objasniti način rada RANSAC-a je procjena modela pravca. želi li se pronaći najbolji model na temelju skupa incidentnih točaka (to jest model kojeg zadovoljava najviše takvih korespondencija), RANSAC će

poduzeti sljedeće korake:

1. Nasumično će odabrati dvije značajke te kroz njih provući pravac
2. Odredit će broj značajki koje pripadaju tom modelu (koje se nalaze unutar praga tolerancije)
3. Ako je pronađen model s dovoljnim brojem značajki, testira se njegova kvaliteta (na svim točkama). Ako je bolji od bilo kojeg dotad pronađenog modela, pamti se kao najbolji.
4. Nakon zadanih broja iteracija, vraća se najbolji model.

Mogući parametri za ovaj algoritam su broj iteracija, prag tolerancije i broj točaka koje moraju uklapati u model da bi on bio valjan. Ova metoda je uklopljena u funkciju koja procjenjuje esencijalnu matrice i zove se `compute_pose_ransac`. U [10] je pokazano da je RANSAC metoda za uklanjanje vanpopulacijskih značajki robusnija i daje bolje rezultate od metode najmanjih kvadrata koju se također može koristiti za ovaj postupak. Razlog tome je što podaci mogu biti toliko podložni šumu i time neispravni da je nemoguće iz njih konstruirati ispravan model.

3.3. Gradijentna optimizacija strukture i kretanja

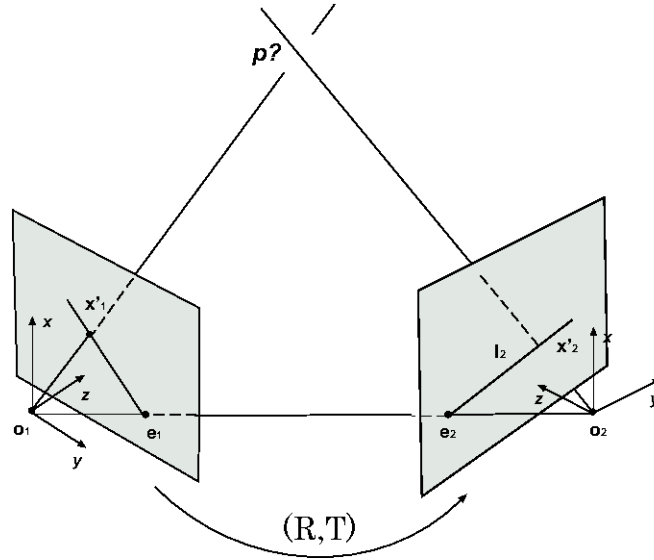
3.3.1. Geometrijsko objašnjenje

Iz dostupnih korespondencija je procijenjena matrica E , te njenom dekompozicijom R i T . Korišteni algoritam pretpostavlja da su korespondentne značajke koje mu se predstavljaju ispravne (slike točno određene točke u 3D prostoru.) Istina je da su na raspolaganju parovi slika podložnih šumu (koji je opisan u prethodnom poglavlju) i da značajke izvađene iz tih slika ne moraju zadovoljavati epipolarno ograničenje. Iz toga slijedi da njihovim korištenjem se ne mora nužno doći do ispravne rekonstrukcije 3D scene kao što se može vidjeti na slici.

Normalizirane koordinate stvarnih značajki x_1 i x_2 su $[x_i, y_i, 1]$. Ono što algoritmu stoji na raspolaganju je par značajki iz slika podložnih šumu $\tilde{x}_1$ i $\tilde{x}_2$ za koje vrijedi:

$$\tilde{x}_1 = x_1 + w_1, \quad \tilde{x}_2 = x_2 + w_2$$

Kao što se vidi iz slike 3.2, takve značajke ne moraju točno zadovoljavati opisano epipolarno ograničenje. Iz takvog skupa korespondentnih značajki je moguće napraviti inicijalnu procjenu esencijalne matrice koja (zbog šuma) ne mora zadovoljavati sva navedena ograničenja niti posjedovati svojstva opisana u prethodnom poglavlju.



Slika 3.2: Korespondentne značajke iz slike podložne šumu ne moraju precizno zadovoljavati elipolarno ograničenje

3.3.2. Mjera pogreške procjene 3D položaja točke

Kod procjene homografije iz skupa korespondencija $x_i \leftrightarrow x'_i$ za koju vrijedi $x'_i = Hx_i$ vektori x_i i x'_i bit će jednaki s mogućom razlikom u skalarnom faktoru. Ovaj odnos možemo zapisati kao $x_i \times Hx_i = 0$. Označi li je j -ti red matrice H sa h^{jT} , moguće je zapisati:

$$Hx_i = \begin{pmatrix} h^{1T}x_i \\ h^{2T}x_i \\ h^{3T}x_i \end{pmatrix}$$

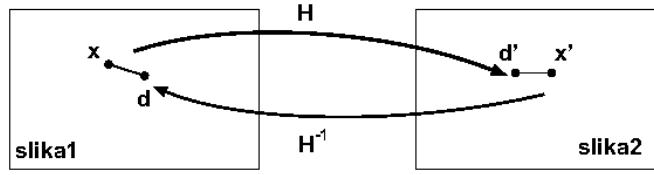
Zapiše li se x'_i kao $(x'_i, y_i, w'_i)^T$, jednadžbu $x'_i = Hx_i$ je moguće zapisati kao:

$$x_i \times Hx_i = \begin{pmatrix} y'_i h^{3T}x_i - w'_i h^{2T}x_i \\ w'_i h^{1T}x_i - x'_i h^{3T}x_i \\ x'_i h^{2T}x_i - y'_i h^{1T}x_i \end{pmatrix}$$

Ova jednadžba se još može zapisati kao:

$$\begin{bmatrix} 0^T & -w'_i x_i^T & y'_i x_i^T \\ w'_i x_i^T & 0^T & -x'_i x_i^T \\ -y'_i x_i^T & x'_i x_i^T & 0^T \end{bmatrix} \begin{pmatrix} h^1 \\ h^2 \\ h^3 \end{pmatrix} = 0 \quad (3.1)$$

Oblik jednadžbe će se u daljnjem tekstu zapisivati kao $A_i h = 0$ gdje je A_i matrica veličine 3×9 , a h vektor duljine 9. Kako jednadžba 3.1 ne mora biti zadovoljena za skup korespondentnih značajki iz slika podložnih šumu, potrebno je na neki način



Slika 3.3: Geometrijska pogreška pri procjeni 3D položaja točke u dvije slike

minimizirati utjecaj šuma na procjenu homografije. Zbog toga se uvode razne mjere koje računaju cijenu pogreške između stvarnih značajki i njihovih procjena. Neke od tih mjera poput algebarske i geometrijske udaljenosti te reprojekcijske i Sampsonove pogreške bit će opisane u nastavku.

Algebarska udaljenost

Da bi se dobilo što bolje rješenje iz skupa korespondentnih značajki, potrebno je minimizirati mjeru $\|Ah\|$. Vektor $\epsilon = Ah$ se naziva rezidualni vektor i potrebno je minimizirati njegovu normu $\|\epsilon\|$. Algebarska udaljenost $d_{alg}x_1, x_2$ gdje x_1 odgovara vektoru x'_i , a x_2 vektoru Hx_i računa se kao:

$$d_{alg}(x_1, x_2) = a_1^2 + a_2^2, \quad a = (a_1, a_2, a_3)^T = x_1 \times x_2$$

Algebarska pogreška nije geometrijski niti statistički informativna. Prednosti minimizacije algebarske pogreške jesu linearna priroda te brzina postupka. često se koristi prije drugih metoda minimizacije te im služi kao početna točka.

Geometrijska udaljenost

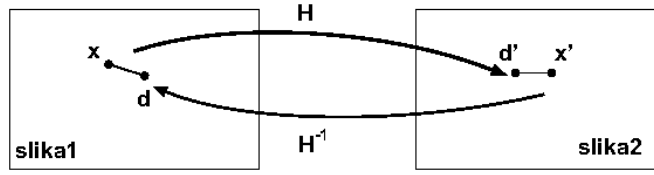
Geometrijska udaljenost se računa kao prijenosna pogreška na jednoj slici. Ako se računa na slici druge kamere, tada je ona definirana kao euklidska udaljenost između izmjerene točke x' i točke $H\bar{x}$ u koju se preslikava korespondentna točka $\bar{x}$ iz slike prve kamere. Prijenosna pogreška za skup korespondentnih značajki je:

$$\sum_i d(x'_i, H\bar{x}_i)^2 \quad (3.2)$$

Homografija za koju je ova pogreška minimizirana je procijenjena homografija $\hat{H}$. Ako se želi minimizirati pogreška u obje slike kao na slici 3.3, moguće je kao bolju funkciju pogreške koristiti unaprijednu (H) i unazadnu (H^{-1}) transformaciju te zbrojiti pogreške nastale ovim transformacijama. Dakle, funkcija pogreške bi bila:

$$\sum_i d(x_i, H^{-1}x'_i)^2 + d(x'_i, Hx_i)^2 \quad (3.3)$$

Prvi član zbroja je pogreška u prvoj slici, a drugi pogreška u drugoj slici.



Slika 3.4: Reprojekcijska pogreška pri procjeni 3D položaja točke

Reprojekcijska pogreška

Moguća funkcija pogreške za obje slike je procjena potrebne korekcije za svaki par korespondentnih značajki. Korekcije se obavljaju za obje slike u cilju dobivanja para točaka koje u potpunosti odgovaraju (geometrijska greška je jednaka nuli).

Potrebno je pronaći takvu homografiju $\hat{H}$ i parove u potpunosti odgovarajućih točaka $\hat{x}_i$ i $\hat{x}'_i$ koji minimiziraju funkciju pogreške:

$$\sum_i d(x_i, \hat{x}_i)^2 + d(x'_i, \hat{x}'_i), \quad \hat{x}'_i = \hat{H} \hat{x}_i \quad (3.4)$$

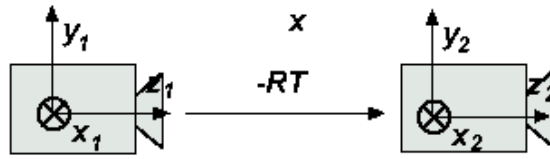
Sampsonova pogreška

Ne postoji jednostavan način za minimizaciju reprojekcijske pogreške 3.4 budući da je potrebno istovremeno podešavati koordinate točaka i procijenjenu homografiju. Sampsonova pogreška je aproksimacija reprojekcijske pogreške.

Vektor $\hat{X}$ koji minimizira udaljenost $\|X - \hat{X}\|$ nije moguće procijeniti nikakvim linearnim postupkom. Sampsonova pogreška se zasniva na procjeni prvog reda vektora $\hat{X}$. Za svaku točku 3D prostora X koja pripada sceni vrijedi $Ah = 0$. Tu jednadžbu je moguće zapisati kao funkciju od X - $C_H(X) = 0$. Ta funkcija se može aproksimirati ekspanzijom u Taylorov red

$$C_H(X + \delta_X) = C_H(X) + \frac{\partial C_H}{\partial X} \delta_X \quad (3.5)$$

Ako je $\delta_X = \hat{X} - X$, gdje vrijedi $C_H(\hat{X}) = 0$, tada je rezultat jednadžbe 3.5 jednak nuli. Iz toga se može zaključiti da je $J\delta_X = -\epsilon$, gdje je J matrica parcijalne derivacije, a ϵ cijena $C_H(X)$. Potrebno je naći najmanji δ_X koji zadovoljava tu jednadžbu. Ovaj problem se može riješiti uporabom Lagrangeovih multiplikatora. U ovom radu neće biti detaljno opisan postupak minimizacije pogreške koristeći Sampsonovu funkciju pogreške. Intuitivno objašnjenje je dano iz razloga što programska implementacija minimizacije pogreške upotrebljena u ovom radu koristi upravo Sampsonovu funkciju pogreške.



Slika 3.5: Odnos sustava dviju kamera

3.3.3. Programska implementacija

Gradijentna optimizacija je najčešće zadnji korak svih algoritama rekonstrukcije strukture i kretanja iz geometrije više pogleda. Optimizacijskom algoritmu (u ovom slučaju se radi o Levenberg-Marquardt algoritmu [5]) je potrebno predati početnu procjenu koju on pokušava poboljšati smanjujući reprojekcijsku pogrešku između zabilježenih točaka i onih predviđenih početnom procjenom.

Standardne implementacije optimizacijskog algoritma (ref) poput MINPACK-a postaju računski iznimno zahtjevne za probleme ove vrste gdje im je predložen problem s tolikim brojem varijabli kao što je to slučaj u optimizaciji strukture i kretanja.

Neovisnost podataka o pojedinim kamerama i 3D točkama rezultira matičnim zapisom koji je rijetko popunjen. Implementacija gradijentne optimizacije strukture i kretanja SBA (Sparse Bundle Adjustment) koristi optimizacijski algoritam na nešto drukčiji način uzimajući u obzir prirodu rijetko popunjenih matrica i rezultira značajnim vremenskim uštedama. Dostupne funkcije za uporabu ove implementacije su:

- `sba_motstr_levmar()` te `sba_motstr_levmar_x()` što su jednostavni i složeni pokretač za gradijentnu optimizaciju strukture i kretanja
- `sba_mot_levmar()` te `sba_mot_levmar_x()` što su jednostavni i složeni pokretač za gradijentnu optimizaciju samo kretanja
- `sba_str_levmar()` te `sba_str_levmar_x()` što su jednostavni i složeni pokretač za gradijentnu optimizaciju samo strukture

3.4. Izračun ukupne rotacije i translacije

Ako se za točku 3D prostora čiji položaj računamo uzme položaj prve kamere, te njen položaj promatra iz sustava druge kamere, onda se udaljenost između ishodišta koordinatnog sustava druge kamere i prve kamere može zapisati kao RT . Budući da je to udaljenost izražena iz koordinatnog sustava druge kamere, ako je želimo promatrati iz koordinatnog sustava prve kamere, potrebno je dobivenu vrijednost pomnožiti sa -1 , dakle $-RT$. Ovaj odnos može se vidjeti na slici 3.4..

Između svaka dva okvira u videu se nanovo računaju rotacijske i translacijske matrice kamere i time se dobiva projekcijska matrica kamere $P = RT$. Kako bi se došlo do ukupne rotacije vozila, za svaki okvir je potrebno naći rotacijsku matricu R_A koja opisuje dotada izvršenu rotaciju te vektor ukupne translacije T_A . Budući da se procjena ovih parametara vrši iz sustava druge kamere te kako se vozilo giba prema naprijed, očito će dobiveni pomak trebati pomnožiti s -1 (faktor nije uzet u obzir u izvodima). Promatra li se točka Q u 3D prostoru, njen položaj u sustavu prve kamere (prije početka kretanja) može se označiti sa Q_O . Njen položaj u sustavu druge kamere će iznositi:

$$Q_1 = R_1 Q_0 + T_1 \quad (3.6)$$

Za sustav treće kamere vrijedit će:

$$Q_2 = R_1 Q_1 + T_2 = R_2 R_1 Q_0 + R_2 T_1 + T_2 \quad (3.7)$$

Iz ovih izraza može se doći do formule koja u svakom okviru opisuje R_A i T_A :

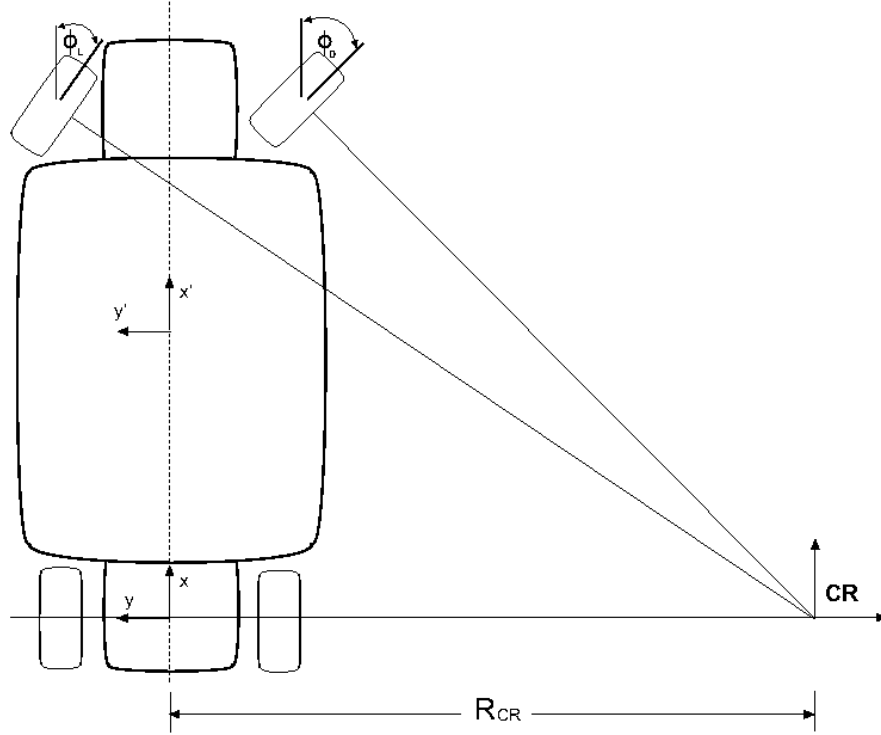
$$Q_n = \prod_{i=n}^1 R_i Q_0 + \sum_{i=1}^n \left(\prod_{j=n}^{i+1} R_j \right) T_i \quad (3.8)$$

Iz izraza 3.8 se može vidjeti da se R_A i T_A mogu izraziti kao:

$$R_A = \prod_{i=n}^1 R_i, T_A = \sum_{i=1}^n \left(\prod_{j=n}^{i+1} R_j \right) T_i \quad (3.9)$$

3.5. Korištenje ograničenja gibanja cestovnog vozila

Dosad opisani postupak je primjenjiv za sve vrste kretanja u 3D prostoru. Ovakva općenitost nije potrebna promotri li se podrobnije priroda kretanja cestovnog vozila. Vozilo ima ograničene mogućnosti kretanja. Pravocrtno kretanje je moguće opisati na jednostavan način. Iz slike 3.6 je moguće zaključiti da je i skretanje cestovnog vozila dovoljno ograničeno za jednostavno objašnjenje. Pri svakom skretanju, postoji jedinstvena točka u prostoru koja je centar kružnice po kojoj se kreću prednji kotači vozila. Moguće ju je izračunati kao presjek pravaca okomitih na ravnine kotača. Ravnine kotača zatvaraju sa x osi koordinatnog sustava kuteve Φ_L (lijevi kotač) te Φ_D (desni kotač). Koristeći ovaj model, pravocrtno kretanje se može opisati kao kretanje po kružnici beskonačnog polumjera. Pretpostavka ovog modela je kretanje po ravnini.



Slika 3.6: Model kretanja automobila po kružnici polumjera R_{CR} s centrom u CR (centar rotacije)

3.5.1. Poseban oblik esencijalne matrice uz ograničenja gibanja

Uzevši da je odnos između dva sustava kamera u opisanom kružnom modelu kretanja kao na slici 3.7, rotacijska matrica R i vektor translacije T poprimaju specifičan oblik:

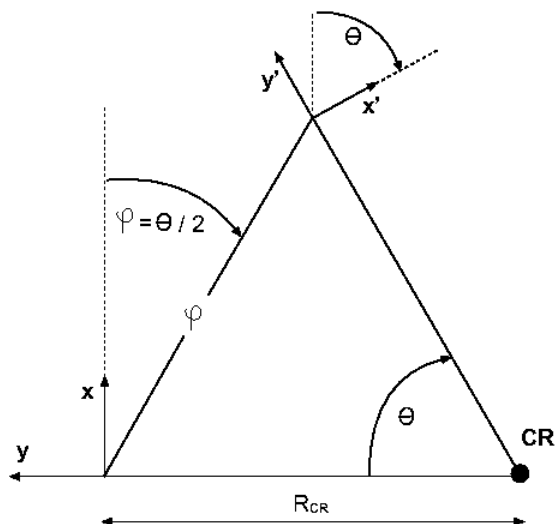
$$R = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad T = \rho \begin{bmatrix} \cos(\theta) \\ \sin(\theta) \\ 0 \end{bmatrix}$$

gdje je ρ skalarni faktor. Koristeći ovakav način procjene esencijalne matrice, do rješenja je moguće doći koristeći samo jednu točku i to na način:

$$\theta = -2 \tan^{-1} \left(\frac{y'z - z'y}{x'z + z'x} \right) \quad (3.10)$$

gdje su $p = [x, y, z]^T$ i $p' = [x', y', z']^T$ koordinate točke u dva sustava kamera. Ako je na raspolaganju više korespondencija, tada je moguće do rješenja doći singularnom dekompozicijom primjenjenom na vektor $[\sin(\frac{\theta}{2}), \cos(\frac{\theta}{2})]$. Prvo je potrebno napraviti matricu D , dimenzija $m \times 2$ gdje je m broj korespondentnih značajki. Svaki redak te matrice će biti oblika:

$$[(x'z + z'x), (y'z - z'y)]$$



Slika 3.7: Odnos koordinatnih sustava kamera tokom kužnog kretanja

Singularna dekompozicija će rezultirati sljedećim matricama:

$$D_{m \times 2} = U_{m \times 2} \Lambda_{2 \times 2} V_{2 \times 2}$$

Stupci matrice V su jedinstveni vektori e_i matrice $D^T D$. Jedinstveni vektor e^* za kog vrijedi $\|e^*\| = 1$ je upravo vektor $[\sin(\frac{\theta}{2}), \cos(\frac{\theta}{2})]$ i iz njega je moguće izračunati kut θ .

3.5.2. Odstranjivanje vanpopulacijskih značajki

U ovakvom ograničenom modelu je moguće na učinkovitiji način uklanjati outliere zbog potrebe za samo jednom korespondencijom za procjenu kretanja. Tako je moguće iskoristiti RANSAC algoritam, ali na način da je potrebno pronaći samo jednu točku koja zadovoljava pretpostavljeni model. To pridonosi ubrzanju sustava.

Drugi mogući način je kreiranje histograma sa vrijednostima kuteva iz svake korespondencije na x osi, a brojem korespondencija sa istim vrijednostima kuta na osi y . Kao značajke koje pripadaju modelu će biti klasificirane one korespondencije koje imaju istu vrijednost kuta koji se načešće pojavljuje ili kao srednju vrijednost svih izračunatih kuteva.

4. Programska implementacija

4.1. Korišteni programski paketi

Bundler [2] je aplikacija koja iz niza slika objekta slikanih iz različitih položaja kamere rekonstruira 3D model objekta. Iako se možda svrha aplikacije čini dosta različitom od svrhe ovog rada, šire područje je isto - SFM (Structure From Motion) ili rekonstrukcija strukture iz kretanja. Analizom izvornog koda utvrđeno je da su u Bundleru iskorištene knjižnice koje bi se mogle integrirati u ovaj rad i znatno olakšati potreban posao. Primjer takve knjižnice je implementacija algoritma 5 točaka za procjenu esencijalne matrice koji se mogao iskoristiti bez preinaka.

Prije korištenja Bundlera bilo je potrebno utvrditi način njegovog rada. Iz tog razloga, kao i neintuitivne prirode postupka, je u daljnjem tekstu ukratko opisan postupak prevođenja i pokretanja aplikacije. Opis se odnosi na neizvršnu verziju 0.4, adaptaciju za Microsoft Visual Studio.

Program isprva nije bilo moguće prevesti zbog referenciranja na nepostojeće datoteke. Potrebno je preimenovati datoteke `sysdep1.h0` u `sysdep1.h` te `signal1.h0` u `signal1.h`. Potrebno je dodati `arith.h` datoteku koja je preuzeta sa sljedeće poveznice: http://www-rocq.inria.fr/secret/MCE/arith_8h-source.html te datoteku `buff.h` koja je preuzeta sa http://www-rocq.inria.fr/secret/MCE/buff_8h-source.html. Naredbu

```
mkdir(output_path, 0770);
```

u datoteci `Bundle2PMVS.cpp` na liniji 151 je bilo potrebno zamijeniti sa sljedećim odsječkom koda:

```
char command[500];  
sprintf (command, "mkdir \"%s\"", output_path);  
system(command);
```

Naredbu

```
char *space = index(buf, ' ');
```

u datoteci RadialUndistort.cpp je bilo potrebno zamijeniti sa naredbom

```
char *space = strchr(buf, ' ');
```

Nakon navedenih izmjena, program se mo2e prevesti. Budući da daljnje korištenje pretpostavlja da se u bin direktoriju nalazi izvršni program Bundler.exe, potrebno ga je kopirati iz Debug ili Release direktorija koji generira MVS, kao i datoteke Bundle2PMVS.exe, KeyMatchFull.exe i RadialUndistort.exe. Da bi se mogao i pokretati, potrebno je skinuti programsku podršku za vađenje SIFT značajki iz slika, dostupan na sljedećem linku: <http://www.cs.ubc.ca/lowe/keypoints/>. Datoteku siftWin32.exe je potrebno spremiti u bin direktorij Bundlera. Nakon toga se mo2e krenuti za korištenjem programa. Iz baznog direktorija Bundlera (tamo gdje je raspakiran) potrebno je pokrenuti RunBundler.sh skriptu koja obavlja sav potreban posao za rekonstrukciju 3D modela objekta. Izlaz koji se dobije moguće je pregledati korištenjem Blendera.

4.2. Glavna aplikacija

Programska implementacija sustava vizualne odometrije za cestovna vozila u sklopu ovog rada je integrirana u već postojeći sustav za implementaciju algoritama računalnog vida - cvsh (Computer Vision SHell).

Sustav za praćenje kamere moguće je iz cvsh pozvati naredbom "alg abulovic" nakon što se ljska pokrene ili pri pokretanju predati parametar u komandnoj liniji "-a abulovic". Općenito, standardni parametri pri pokretanju programa su bili:

1. -c="1 fixed:ipv-panasonic-720" (konfiguracija kamere i KLT-a)
2. -a=abulovic (poziv algoritma za praćenje kamere)
3. -sf="... put do nekog videa ..."
4. -i="p n" (naredba za obradu sljedećeg okvira)

4.2.1. Vađenje značajki

Za proces vađenja i praćenja značajki je korištena implementacija KLT (Kanade-Lucas-Tomasi) sustava za procjenu optičkog toka. Za inicijalizaciju algoritma bilo je potrebno napraviti sljedeće:

```
klt_.setpars(false, 7, 10, 0, 0);
klt_.reset(500);
```

Od metode “setpars” samo su drugi i treći parametar od interesa. Drugi parametar predstavlja model deformacije i kao moguće konfiguracije ima smisla odabrati vrijednosti 5, 6 i 7. Treći parametar je prag reziduala i ima smisla odabrati broj između 5 i 15.

Metode na raspolaganju pri korištenju klt implementacije iz cvsh programskog sustava su:

- `int ext_klt::init(const img_wrap& src)` koja inicijalizira praćenje značajki za sliku koja joj se preda
- `int ext_klt::process(const img_wrap& src)` koja obrađuje predanu sliku tako da prati značajke iz prethodnog okvira u trenutnoj slici
- `int ext_klt::size(void)` koja vraća broj uspješno praćenih značajki
- `int ext_klt::fAge(int i)` koja za *i*-tu značajku vraća 0 ako nije uspješno praćena, a 1 ako je
- `const math::Point2D& fThis(int i)` koja za redni broj značajke *i* vraća njenu lokaciju na trenutnom okviru
- `const math::Point2D& fPrev(int i)` koja za redni broj značajke *i* vraća njenu lokaciju na prethodnom okviru.

Ovisno o tome kako je postavljen parametar “initTracker_” obaviti će se ili inicijalizacija praćenja ili nastavak pozivom metode `process`.

4.2.2. Procjena esencijalne matrice

Za procjenu esencijalne matrice iz korespondentnih značajki korištena je gotova implementacija [2] algoritma za pet točaka. Osnovno sučelje za korištenje ovog programskog paketa je pozivom metode

```
int compute_pose_ransac(int n, v2_t *r_pts, v2_t *l_pts,
                        double *K1, double *K2,
                        double ransac_threshold, int ransac_rounds,
                        double *R_out, double *t_out)
```

gdje su značenja parametara sljedeća:

1. *n* - broj korespondentnih značajki

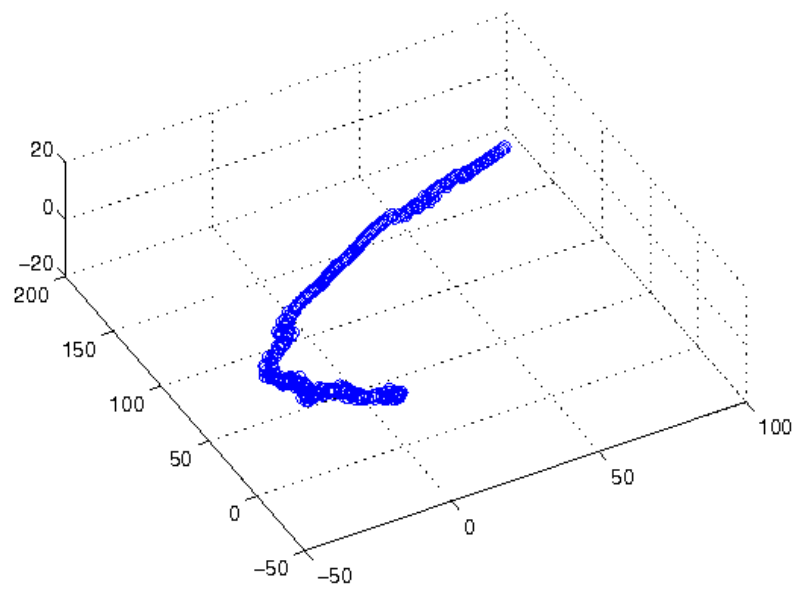
2. r_pts - točke iz desnog pogleda (mogu se dohvatiti pozivima metoda $fThis$)
3. l_pts - točke iz lijevog pogleda (mogu se dohvatiti pozivima metoda $fPrev$)
4. $K1$ - matrica parametara prve kamere (kamere prvog pogleda)
5. $K2$ - matrica parametara druge kamere (kamere drugog pogleda)
6. $ransac_threshold$ - prag tolerancije za RANSAC metodu uklanjanja nepogodnih značajki
7. $ransac_rounds$ - broj iteracija koje RANSAC metoda može izvršiti prije nego dojaviti najbolji model
8. R_out - rezultat poziva metode - matrica rotacije
9. t_out - rezultat poziva metode - matrica translacije

Kao što se može vidjeti iz povratnih parametara metode (R i t), ona vrši dekompoziciju esencijalne matrice na rotacijsku matricu i translacijski vektor.

4.3. Izračun položaja kamere

Za izračun položaja kamere koriste ukupnu rotaciju i translaciju u svakom okviru napisana na skriptu u programskom alatu Matlab. Prvo su izračunate sve matrice R_A . Nakon toga, pomoću njih su izračunati svi translacijski vektori T_A .

Na kraju se za sve okvire redom iscertavaju vektori $-R_A T_A$. Rezultat se može vidjeti na slici 4.3..

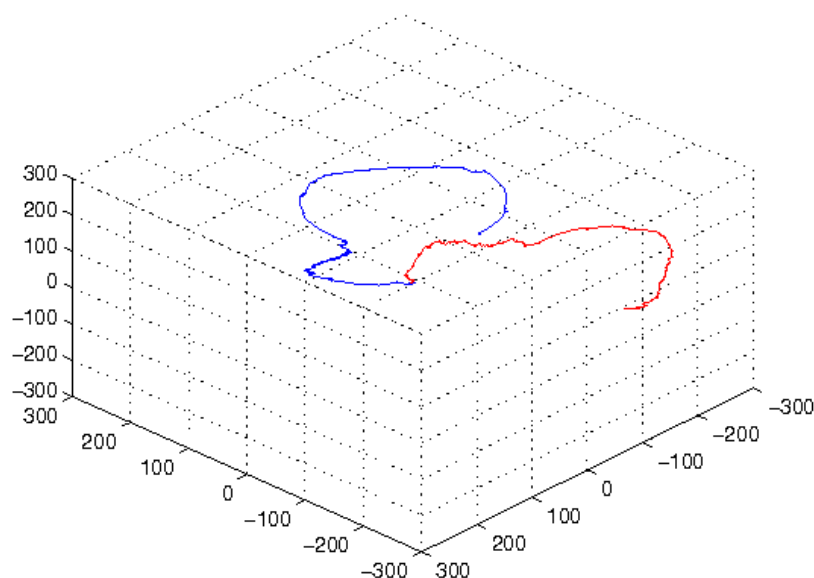


Slika 4.1: Tipičan rezultat estimacije kretanja koristeći apsolutnu rotaciju i translaciju

5. Testiranje programskog rješenja

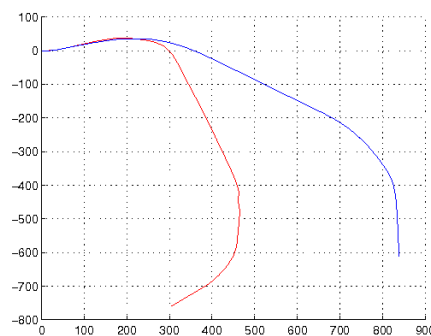
Tokom implementacije programskog rješenja, isprobane su različite kombinacije parametara. Neki od alterniranih parametara su broj značajki koje se pronalaze u trenutnom okviru pri inicijalizaciji praćenja, minimalan broj značajki uz koji se ne provodi ponovna inicijalizacija praćenja te postotak značajki koje su se pomakle manje od zadane granice t da se odredi da između 2 okvira nema kretanja.

5.1. Alterniranje broja značajki kojima se inicijalizira praćenje



Slika 5.1: Procjena puta temeljena na izračunu apsolutne rotacije i translacije koristeći 500 (plava) i 100 (crvena) značajki

Na slici 5.1. se mogu vidjeti rezultati procjene apsolutnih rotacijskih i translacijskih matrica za svaki okvir. Kako je taj proces računski zahtjevan, ne nalazi se u sklopu



Slika 5.2: Prvi zavoj (lijevo) i usporedba estimacije kretanja koristeći inicijalizaciju praćenja na 100 (crvena boja) i 500 (plava boja) značajki (desno)

glavne aplikacije koja vrši procjenu esencijalne matrice za svaka dva susjedna okvira, već je napisana kao zasebna Matlab skripta.

Zbog usporedbe sa dobivenim rješenjem u 3D prostoru, implementirano je rješenje koje koristi ograničenje kretanja cestovnog vozila opisanog u poglavlju 3.5.[8]. Rezultati ovakve implementacije se mogu vidjeti na slici 5.2 gdje je estimirani put pomoću inicijalizacije praćenja sa 500 značajki iscrtan plavom bojom, a onaj sa 100 značajki crvenom. Uočeno je da se sa manje značajki procjenjuju veći kutevi nego što je to slučaj sa više značajki.

5.2. Uklanjanje vanpopulacijskih značajki korištenjem histograma vrijednosti kuteva

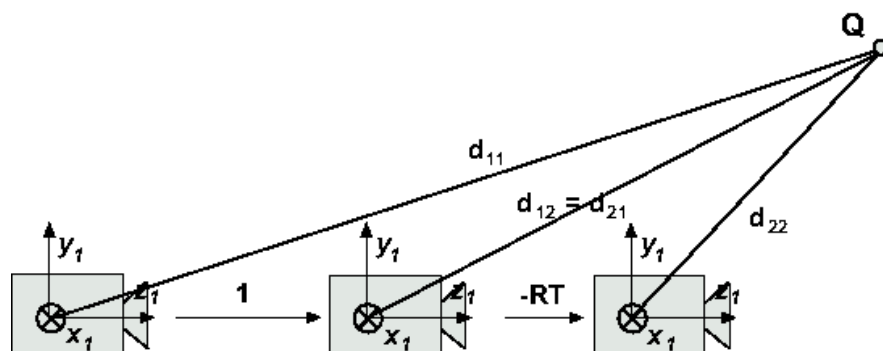
Tokom testiranja rješenja, implementiran je izračun histograma na način kako je opisano u [8]. Kako metoda koja obavlja procjenu esencijalne matrice vraća samo broj značajki koje odgovaraju modelu, ali ne i konkretne značajke, pokušaj je napravljen sa svim značajkama. Nakon izračuna kuteva za sve značajke po jednadžbi 3.5.1. provodi se odlučivanje o trenutnom kutu. Prvi pokušaj donošenja odluke je bio uzimanjem srednje vrijednosti svih kuteva. Zbog prevelikog šuma prevladavale su slučajne vrijednosti. Pretpostavi li se pravi kut iznosi 60 stupnjeva, dovoljna je tek nekolicina vanpopulacijskih značajki koje rezultiraju negativnim kutem da se srednja vrijednost značajno pomakne od stvarne. Zbog toga je isprobano i histogramsko glasanje. Intervali kuteva su po 10 stupnjeva. Iako je davalo nešto bolje rezultate, i to samo u određenim situacijama, niti ovo se nije pokazalo kao iskoristivo rješenje procjene kuta.

6. Moguća poboljšanja

6.1. Triangulacija točaka

Implementirano rješenje pretpostavlja da se cestovno vozilo kreće konstantnom brzinom. Iz toga slijedi da je udaljenost između svaka dva uzastopna okvira jednaka 1. Ovo u stvarnosti nije uvijek zadovoljeno. Zbog toga je potrebno na neki način odrediti relativne udaljenosti među okvirima u odnosu na početnu koja se može proizvoljno postaviti na jedan. Ovo je moguće postići triangulacijom točaka.

U postupku triangulacije može se postaviti udaljenost između prva dva okvira na 1. Pomoću koordinata 3D točke Q takve da ju je moguće pratiti kroz više okvira može se doći do njene udaljenosti od ishodišta koordinatnog sustava kamere. Potrebno je izračunati udaljenosti iste točke iz 2 uzastopna sustava kamere (uzevši u obzir da je prije ta dva postojao bar još jedan sustav kamere). Udaljenost kamere koja im prethodi i Q gledano iz sustava prve kamere označena je sa d_{11} . Udaljenost točke od ishodišta prve kamere označena je sa d_{12} . Iste udaljenosti u sustavu druge kamere označene su sa d_{21} i d_{22} . Inzistiranjem da vrijedi $d_{12} = d_{21}$ moguće je doći do skalarnog faktora kojim kad se pomnoži udaljenost sustava prve i druge kamere, dobije se udaljenost koja je realna u odnosu na postavljenu početnu udaljenost 1. U sustavu vizualne odometrije lako je moguća akumulacija pogreške pri procjeni 3D položaja točaka. To implicira da



Slika 6.1: Određivanje udaljenosti među sustavima kamera triangulacijom točaka

se sustav nikada nema priliku oporaviti od jednom počinjene greške. Da bi se riješio taj problem, moguće je implementirati svojevrсни vatrozid koji bi spriječio akumulaciju pogreške kako je predloženo u [4].

6.2. Ostala moguća poboljšanja

U implementiranom programskom rješenju se koristi prva estimacija esencijalne matrice. Ovu procjenu je moguće poboljšati koristi li se gradijentna optimizacija opisana u poglavlju 3.3..

Još jedno moguće poboljšanje je korištenje drugog algoritma za uklanjanje vanpopulacijskih značajki - MLESAC-a [9].

7. Zaključak

U sklopu ovog rada programski je implementirana procjena kretanja kamere na osnovu vizualnih podataka, točnije video sekvence. Koristeći ovakav način rekonstrukcije kretanja vozila može se s lakoćom pretpostaviti položaj pojedinih objekata čije je pojavljivanje ograničeno položajem ceste. Primjer takvih objekata su prometni znakovi. Mogućnost pretpostavke gdje će se oni pojaviti u slici olakšava proces njihovog praćenja i označavanja.

U procesu implementacije i testiranja očekivane su poteškoće kada vozilo nailazi na zavoje pod velikim kutem zbog toga što je tada teško pratiti značajke koje se najčešće ne nalaze na više od dva uzastopna okvira i njihova udaljenost je relativno velika. Uspriko očekivanju, povećanje broja značajki koje se prate nije previše utjecalo na performanse u opisanom slučaju. Neovisno o inicijalnom broju značajki, njihov broj ubrzo padne ispod granice uspješne procjene. Pokazalo se učinkovitije analizirati situacije u kojima treba nanovo inicijalizirati traženje novih značajki u trenutnom okviru.

Histogramsko glasanje za vrijednost kuta rotacije je iskušano kao metoda uklanjanja vanpopulacijskih značajki, ali nije polučilo pouzdane rezultate.

Procjena kretanja uz korišteno ograničenje gibanja cestovnog vozila daje dobre rezultate (iako nešto manje osjetljivo na promjenu kuta od 3D rekonstrukcije) i iskoristivo je kao bliska aproksimacija. U budućnosti bi trebalo iskoristiti GPS podatke o točnom kretanju vozila te poboljšati inicijalnu procjenu esencijalne matrice koristeći gradijentnu optimizaciju.

LITERATURA

- [1] *KLT: An Implementation of the Kanade-Lucas-Tomasi Feature Tracker.*
- [2] *Bundler: Structure from Motion for Unordered Image Collections.*
- [3] *An Invitation to 3-D Vision - From Images to Geometric Models*, poglavlje Geometry of Two Views. Springer, 2000.
- [4] Oleg Naroditsky David Nistér i James Bergen. Visual odometry for ground vehicle applications. *Journal of Field Robotics, Volume 23*, siječanj 2006.
- [5] A.A. Argyros M.I.A. Lourakis. Sba: A software package for generic sparse bundle adjustment. *ACM Transactions on Mathematical Software (TOMS)*, 36(1), ožujak 2009.
- [6] David Nistér. An efficient solution to the five-point relative pose problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, stranice 756–770, lipanj 2004.
- [7] M. Sarhadi R. Bowden, T.A. Mitchell. Reconstructing 3d pose and motion from a single camera view. *British Machine Vision Conference*, 2, 1998.
- [8] Davide Scaramuzza i Roland Siegwart. 1-point-ransac based visual odometry for vehicle mounted cameras by exploiting non-holonomic constraint. *International Journal of Computer Vision*, 2010.
- [9] P.H.S. Torr i A. Zisserman. Mlesac: A new robust estimator with application to estimating image geometry. *Journal of Computer Vision and Image Understanding*, stranice 138–156, 2000.
- [10] P.H.S. Torr, P.A. Beardsley, i D.W. Murray. Robust vision. U *British Machine Vision Conference*, 1994.

- [11] Mark Maimone Yang Cheng i Larry Matthies. Visual odometry on the mars exploration rovers. Technical report, California Institute of Technology, 2003.
- [12] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, 2000.

Monokularna vizualna odometrija za cestovna vozila

Sažetak

Zbog jednostavnosti instalacije i relativno niske cijene potrebne opreme, korisno je moći procijeniti kretanje vozila na temelju vizualnih podataka u obliku videa. Takav pristup estimaciji kretanja je primijenjen u ovom radu gdje je izvor potrebnih informacija kamera montirana na krov automobila. Konstrukcija ovog rješenja ima primjenu u procjeni položaja i kretanja objekata čija je pozicija usko vezana uz položaj ceste. Primjer takvih objekata su prometni znakovi. Implementirana procjena kretanja se oslanja na podatke dobivene iz korespondentnih značajki pronađenih u 2 ili više uzastopnih okvira. Ti podaci su iskorišteni za procjenu relativne rotacije i translacije među sustavima pojedinih kamera. Za opisani postupak nije potrebno nikakvo prethodno poznavanje ceste. Kako je rješenje rabljeno isključivo za procjenu kretanja cestovnog vozila, koristeći ograničenja koje takvo kretanje nameće, postignuto je značajno ubrzanje u rekonstrukciji puta. Podaci korišteni za evaluaciju rješenja su snimke iz prethodno kalibriranih kamera na županijskih cesta u Hrvatskoj.

Ključne riječi: vizualna odometrija, epipolarna geometrija, esencijalna matrica, epipolarno ograničenje, kalibracija kamera, RANSAC, gradijentna optimizacija

Monocular visual odometry for road vehicles

Abstract

Due to the simplicity of the installation process and the low pricing of the equipment needed, it can be very useful to be able to estimate motion using visual data only. Such an approach has been implemented in this work where the visual information source was the camera mounted on top of a car. Constructing such a solution to the problem of motion estimation can be used in estimating both position and motion of certain objects whose pose is limited by the positioning of the road, such as traffic signs. The implemented motion estimation in this work relies on the data extracted from point correspondences in two or more consecutive frames. This data is then used to reconstruct the relative rotation and translation between neighboring frames. Since this implementation has the sole purpose of estimating motion of road vehicles, it has been possible to enforce certain constraints that are the result of a particular kind of restricted motion in question. This resulted in an performance increase in the path reconstruction stage. The data used for evaluation are videos taken from previously calibrated cameras on county roads in Croatia.

Keywords: visual odometry, epipolar geometry, essential matrix, epipolar constraint, camera calibration, RANSAC, gradient optimization