

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

Tin Mišić

**Praćenje objekata aktivnim stereo
vizualnim sustavom**

Zagreb, travanj 2023.

Ovaj rad izrađen je u Zavodu za automatiku i računalno inženjerstvo pod mentorstvom prof. dr. sc. Ivana Markovića i neposrednim vodstvom Karla Koledića, mag. ing., i predan je na natječaj za dodjelu Rektorove nagrade u akademskoj godini 2022./2023.

SADRŽAJ

1. Uvod	1
2. Pristupi praćenju gibajućih objekata stereo vizijom	2
2.1. Stereo vizija	2
2.1.1. Disparitet	4
2.2. Aktivni stereo sustav	5
2.2.1. Horopter	6
2.2.2. Filter nultog dispariteta	8
2.2.3. Log-polarno mapiranje	8
2.2.4. Pregled algoritma praćenja aktivnim stereo sustavom	10
3. Implementacija aktivnog stereo sustava	12
3.1. Oblikovanje sustava i specifikacije	12
3.1.1. Hardverske komponente	12
3.1.2. Komponente programske potpore	14
3.2. Cjevovod sustava i upravljanje	16
3.2.1. Inicijalizacija sustava i upravljanje	17
3.2.2. Određivanje pozicije objekta	17
4. Praćenje gibajućeg objekta	19
4.1. Eksperimentalna postava	19
4.1.1. Opis postave	19
4.1.2. Eksperimenti na kartezijskim slikama	20
4.1.3. Eksperimenti na log-polarnim slikama	21
4.2. Rezultati	22
4.2.1. Eksperiment 1: Gibanje na kartezijskoj slici bez pozadinske smetnje	22

4.2.2.	Eksperiment 2: Gibanje na kartezijskoj slici s pozadinskom smetnjom	23
4.2.3.	Eksperiment 3: Gibanje na log-polarnoj slici bez pozadinske smetnje	25
4.2.4.	Eksperiment 4: Gibanje na log-polarnoj slici s pozadinskom smetnjom	26
4.3.	Usporedba pristupa	28
4.3.1.	Izvođenje u stvarnom vremenu	28
4.3.2.	Estimacija pozicije	28
4.3.3.	Učinkovitost virtualnih horoptera	29
5.	Zaključak	30
6.	Zahvale	31
	Literatura	32

1. Uvod

Praćenje gibajućih objekata nalazi brojne primjene u područjima robotike, autonomnih vozila i nadzora. Ono nam također omogućuje modeliranje kognitivnog procesa pažnje. Zbog toga su mnogi pristupi praćenju s biološkom motivacijom jako atraktivni.

Cilj ovoga rada jeste dati pregled pristupa praćenju korištenjem stereo vida, posebice onih sustava koji koriste motore kako bi pomjerali kamere. Cilj ovakvih pristupa jeste zadržati praćeni objekt u središtu slike, gdje se željene operacije nad slikom mogu vršiti na malenom području.

Poseban naglasak stavlja se na pristupe koji koriste virtualne horoptere i log-polarno mapiranje slike. Log-polarno mapiranje je vrlo slično ljudskom vidu, gdje je središte slike izoštreno, a rubovi mutni. Osim biološke sličnosti, log-polarno mapiranje ima i korisne osobine pri obradi slike.

U radu je također oblikovan i ispitan aktivni stereo sustav koji koristi web kamere i servo motore, a radi na osobnom računalu. Ispitan je rad oblikovanog sustava u stvarnom vremenu, utjecaj pozadine na praćenje gibajućeg objekta, gubljenje fokusa na objekt i točnost estimacije pozicije objekta. Podaci su dobiveni eksperimentima u kojima je gibajući objekt praćen duž vidnog polja sustava.

Nakon uvoda slijedi pregled različitih pristupa praćenju objekata stereo vizijom. Prvo je ukratko opisan klasični stereo sustav, a nakon toga se poseban naglasak stavlja na aktivne stereo sustave. Objašnjeni su pojmovi horoptera, filtera nultog dispariteta i log-polarnog mapiranja slike te je dan pregled algoritma praćenja korištenjem virtualnih horoptera.

Nakon toga slijedi opis implementiranog sustava, gdje su opisane hardverske komponente i komponente programske podrške. Poblježe je opisan tijek rada samog programa i neke ključne funkcije.

U posljednjem poglavlju prije zaključka opisana je eksperimentalna postava kojom su ispitane ranije navedene osobine sustava te su opisani rezultati eksperimenata. Na kraju je dana usporedba različitih pristupa - prednosti jednoga nad drugim te neki univerzalni nedostaci. Predložena su i neka rješenja za navedene nedostatke.

2. Pristupi praćenju gibajućih objekata stereo vizijom

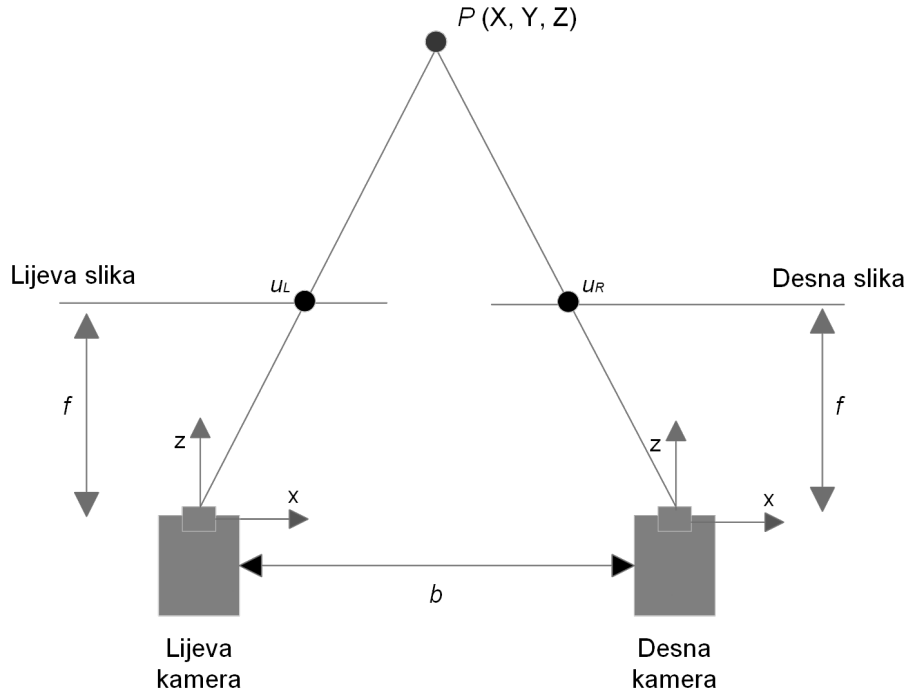
Cilj praćenja objekta je identifikacija i praćenje predmeta interesa tijekom njegova gibanja kroz slijed slika. Praćenje gibanja nekog objekta omogućuje nam modeliranje njegovog gibanja u prostoru i time prilagodbu ponašanja našeg sustava. Zbog toga je rješenje problema praćenja gibajućih objekata od velike važnosti u sustavima koji koriste računalni vid, kao što su autonomna vozila i robotski sustavi. U sustavima za autonomna vozila praćenje gibanja nam omogućava prilagodbu na gibanja ostalih sudionika u prometu, te predstavlja dodatnu mjeru sigurnosti u okruženju s pješacima. Robotski sustavi mogu prilagoditi svoje gibanje u dinamičkom prostoru, gdje poznavanje same karte prostora nije dovoljno.

Praćenje nekog objekta predstavlja određenu vrstu pažnje, jer svoj fokus usmjeravamo na objekt koji izdvajamo od ostatka scene po nekim njegovim osobinama. Te osobine od ostatka scene možemo izdvojiti ekstrakcijom značajki koje su svojstvene objektu, prepoznavanjem samog objekta primjenom strojnog učenja ili čak jednostavnom detekcijom promjene intenziteta piksela na sceni (uz pretpostavku da je ostatak scene statičan). Ovdje se javljaju mnogi izazovi, kao što su prikrivanje objekta, njegovo izobličenje i promjene u osvjetljenju. Često je potrebna kombinacija različitih metoda kako bi se osiguralo dosljedno praćenje objekta u zahtjevnim situacijama.

2.1. Stereo vizija

Praćenje objekta iz samo jedne perspektive ograničava nas na poznavanje njegova gibanja samo u dvodimenzionalnoj projekciji prostora. Drugim riječima, teško možemo odrediti gibanje objekta u trodimenzionalnom prostoru bez neke reference na skalu objekta u prostoru. Postoje pristupi određivanja dubine prostora monokularnim sustavima (sustavi koji koriste samo jednu kameru) korištenjem neke unaprijed poznate skale. Korištena je poznata visina kamere i/ili ravnine ispred kamere (npr. cesta) kako

bi se odredila skala prostora [1], [2], [3]. Uz to, korištena je poznata skala kalibracijom preko šahovnice [4], poznatih značajki određenih objekata i njihovih veličina [5], te naravno korištenje umjetnih neuronskih mreža [6].



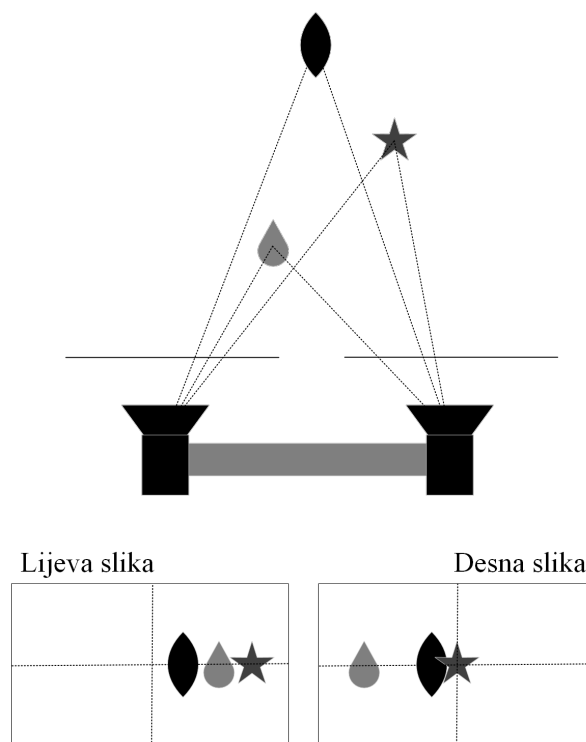
Slika 2.1: Klasični stereo sustav

Stereo kamere javljaju se kao rješenje na problem određivanja dubine promatrane scene. To su sustavi od dvije paralelno postavljene kamere, udaljene jedna od druge za neku poznatu udaljenost (engl. *baseline*). Pošto dvije kamere promatraju scenu iz različitih perspektiva, moguće je odrediti točnu trodimenzionalnu poziciju neke promatrane točke ako su nam poznate vrijednosti (u_L, u_R) na lijevoj i desnoj slici na koje je ona projicirana. Dijagram klasičnog stereo sustava vidljiv je na slici 2.1. Projekcijske ravnine dvaju kamera su obično paralelne u klasičnim stereo sustavima, što dosta olakšava izračun dubine promatrane točke P . Dodatnu motivaciju u korištenju stereo kamera je sličnost stereo vida s prirodnom pojavom stereopsije - percepcije dubine i trodimenzionalne strukture pomoću binokularnog vida.

Primjene stereo vida su mnogobrojne u mnogim različitim područjima, neka od kojih su robotika, autonomna vozila, medicinsko snimanje, proizvodni procesi te zabava.

2.1.1. Disparitet

Od velike je važnosti pojam dispariteta. Točka P je projicirana na projekcijske ravnine desne i lijeve kamere u koordinatama (u_L, v_L) , odnosno (u_R, v_R) ¹. Uz pretpostavku da su projekcijske ravnine, odnosno kamere i njihove osi paralelne, vrijednosti v_L i v_R su jednake, tj. točka P je projicirana na jednaku visinu u obje slike. Ova pretpostavka se može ostvariti rektifikacijom stereo slika - njihovom transformacijom tako da su svi parovi projekcija neke točke projicirani na istu visinu. Stoga nam vrijednosti v_L i v_R nisu od velike koristi pri određivanju dubine promatrane scene.



Slika 2.2: Ilustracija dispariteta raličitih predmeta

Disparitet predstavlja razliku horizontalnih pomaka u_L i u_R između dvaju stereo slika. Što je predmet bliži, to je njegov disparitet veći, dok disparitet predmeta s velikom udaljenošću teži u nulu². Disparitet predmeta različite udaljenosti od stereo sustava ilustriran je na slici 2.2. Poznavanjem dispariteta promatranog predmeta i intrinzičnih osobina kamere (fokus f) vrlo lako možemo odrediti dubinu, odnosno trodimenzionalni položaj predmeta.

¹Vrijednosti u_L i u_R predstavljaju horizontalni pomak u odnosu na središte slike, dok v_L i v_R predstavljaju vertikalni pomak u odnosu na središte.

²Ovo ne vrijedi za stereo sustave u kojima kamere nisu postavljene paralelno, kao što su kasnije opisani aktivni stereo sustavi.

2.2. Aktivni stereo sustav

Pod aktivnim stereo sustavima misli se na sustave gdje svaka od kamera ima dva stupnja slobode - rotacija oko x osi te rotacija oko y osi kamere, za ukupno četiri stupnja slobode. Glavna prednost ovakvih sustava jeste upravljanje pogledom, skup različitih mehanizama koji osiguravaju da promatrani objekt leži u središtu slika obje očiju [7]. U većini slučajeva to se vrši aktiviranjem parom motora za svako oko. Štoviše, može se postaviti ograničenje kojim osiguravamo da je zakretaj oko x osi obje kamere jednak. Ovime sustav ima tri stupnja slobode, što pojednostavljuje kinematiku sustava, praćenje objekta i izračune. Jedna od motivacija za korištenje ovakvog sustava jeste sličnost sa životinjskim vizualnim sustavom, koji upravljanjem pogledom postavlja pažnju na određeni dio promatrane scene.

Prednosti u odnosu na klasični stereo sustav

Postavljanjem objekta u središte slike dobivamo sljedeće prednosti [8]:

- Pošto nam je objekt koji pratimo u središtu slike, možemo koristiti lokalizirane algoritme vizualne obrade i stereo algoritme koji primaju samo ograničen raspon dispariteta.
- Pošto oči prate gibajući objekt, njegovo gibanje po slici je usporeno, dok je gibanje pozadine ubrzano - što dovodi do zamućenja pozadine. Ovime se signal objekta naglašava u odnosu na pozadinu.
- Pošto praćenje vršimo s dvije kamere, možemo iskoristiti koncept horoptera, koji će biti objašnjen kasnije. Njime možemo izolirati područja u prostoru po kojima se giba objekt i time izostaviti sve što je izvan tih područja. Jedan od primjera takve primjene jeste [9].
- Samim time što su kamere pokretne dobivamo na povećanju vidnog polja sustava. Klasični stereo sustavi imaju manje vidno polje od aktivnih, što ograničava područje po kojemu pratimo određeni objekt.
- Određivanje pozicije objekta svodi se na jednostavan izračun inverzne kinematike sustava - objekt na koji smo fokusirani nalazi se na sjecištu pravaca koji su okomiti na projekcijske ravnine pojedinih kamera.

Nedostatci u odnosu na klasični stereo sustav

Nedostatci sustava su sljedeći:

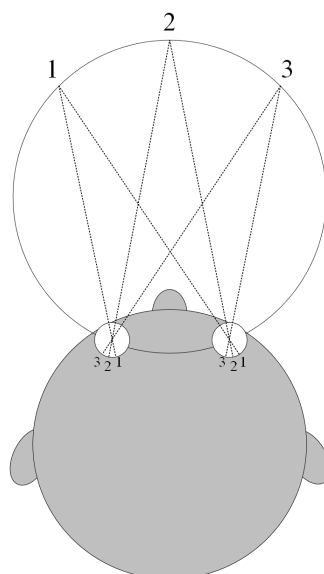
- Upravljanje motorima koji prate objekt mora biti u stvarnom vremenu. Sustav se oslanja na to da je praćeni objekt u središtu slike nakon inicijalne fiksacije. Stoga sve obrade slika i akcije motora moraju biti dovoljno brze da prate objekt.
- Kalibracija sustava i rektifikacija slika su otežane. Iako su nam transformacije između koordinatnih sustava dvaju kamera poznate, svaka drukčija konfiguracija zahtijeva drukčiju kalibraciju i rektifikaciju. Stereo algoritmi se oslanjaju na brzinu pretraživanja dobivenu rektifikacijom slika, stoga ovdje gube na efikasnosti. Zbog toga su aktivni stereo sustavi bolje prilagođeni problemima praćenja, dok su klasični stereo sustavi bolji za određivanje dubine čitave promatrane scene - što je od velike koristi u npr. algoritmima istovremene lokalizacije i mapiranja.

2.2.1. Horopter

Horopter je skup točaka u prostoru koje, kada su projicirane na lijevu i desnu sliku stereo sustava, imaju iste koordinate [10]. Drugim riječima, točke u prostoru koje leže na horopteru imaju disparitet približan ili jednak nuli. Pojam horoptera predstavio je François d'Aguilon u drugoj od svojih knjiga o optici iz 1613. g., dok je geometrija horoptera definirana sredinom 19.st. [10]. Horopter je već duže vrijeme izučavan u oftalmologiji, a u drugoj polovici 20.st. nalazi primjene u robotici i obradi i analizi slike.

Za analizu i definiciju geometrije horoptera korištena je geometrijska algebra [10]. U [11] je koncept horoptera kao krivulje u prostoru proširen na tri dimenzije, čime se dobivaju takozvane horopter-sfere. Autori Fredrik Bergholm i Antonio Francisco su u [12] dokazali da: "*Ako "priroda" konstruira korespondencije binokularnih zraka što je jednostavnije moguće, ona odabire dijelove krugova ili krivulja drugog reda (npr. elipse)*". Također su pokazali geometrijsku povezanost između teorijskog i empirijskog horoptera.

Teorijski horopter naziva se i Vieth-Müller-ov krug. On je definiran s tri točke: središta očiju i točkom na koje su oči fokusirane. Empirijski horopter je drugačijeg oblika, a devijacija među njima naziva se Hering-Hillebrand devijacija [13]. Dijagram Vieth-Müller-ovog kruga nalazi se na slici 2.3. Za potrebe ovog rada dovoljan je Vieth-Müller-ov krug kao definicija horoptera, iako bi za veće udaljenosti trebalo uzeti u obzir i Hering-Hillebrand devijaciju [12].



Slika 2.3: Teorijski horopter, tj. Vieth-Müller-ov krug

Virtualni horopteri

U [8] predstavljeni su virtualni horopteri, a dalje su korišteni u [14]. Iako postoje mnogi razvijeni aktivni stereo sustavi koji se bave upravljanjem pogledom([15], [16], [17], [18], [19], [20]), posebno naglašavamo one koji koriste virtualne horoptere, zbog njihove jednostavnosti i intuitivnosti.

Virtualni horopteri su horopteri koji nastaju ne stvarnom akcijom motora, već "virtualnim" pomjeranjem kamera. Ovo virtualno pomjeranje možemo postići jednostavnim horizontalnim pomakom slike. Mali pomaci slike odgovaraju malim rotacijama kamere [8], čime mijenjamo i sam oblik horoptera i stvaramo nove, virtualne horoptere. Zbog jednostavnosti operacije horizontalnog pomaka, lako i brzo možemo stvoriti više ovakvih inkrementalnih horoptera s nekoliko malih pomaka lijevo, odnosno desno.

Ako se objekt koji pratimo pomjerio van stvarnog horoptera, među virtualnim horopterima pronalazimo onaj koji najbolje opisuje pomak samog objekta. Nakon što smo odabrali najbolji virtualni horopter, iz njegovog horizontalnog pomaka lako izračunavamo kut za koji je potrebno pomjeriti kameru, čime transformiramo stvarni horopter i vraćamo fokus nazad na objekt. Ovime smo osigurali optimalan pomak kamere bez da smo stvarno pomicali kamere kako bismo ispitali koji bi to pomak bio.

Potrebno je naglasiti da horizontalni pomak slike vršimo samo za male pomake, te samo za jednu kameru(desnu ili lijevu). Pošto veliki horizontalni pomak ne odgovara rotaciji kamere, korištenje takvog horoptera bi unijelo velike greške. Pomak obje

kamere ili ne bi imao nikakvog učinka, ili bi pokvario oblik horoptera, sve to zbog stvarnog prostornog pomaka jedne kamere u odnosu na drugu.

2.2.2. Filter nultog dispariteta

Filter nultog dispariteta (engl. *Zero Disparity Filter, ZDF*) je jednostavni filter koji radi na binarnim stereo slikama. Ako nad stereo slikama obavimo detekciju vertikalnih rubova³, dobivamo binarnu sliku na kojoj su jedinice tamo gdje su rubovi, a nule bilo gdje drugo. Sada možemo napraviti jednostavnu AND operaciju nad lijevom i desnom slikom te dobiti binarnu sliku na kojoj su jedinice tamo gdje se rubovi na lijevoj i desnoj slici nalaze na istim koordinatama u slici. Ovime zapravo izdvajamo one rubove koji se nalaze na horopteru, jer je njihov disparitet nula. Nakon toga lako možemo izračunati centroidu rubova i tako odrediti gdje se nalazi objekt na horopteru[8].

Prednost filtera nultog dispariteta jeste u njegovoj jednostavnosti. Klasični stereo algoritmi zahtijevaju pretraživanje obje slike kako bi se pronašle jednake značajke i time odredio disparitet. Filter nultog dispariteta ne može odrediti disparitet svakog piksela kao što to mogu klasični stereo algoritmi, već samo područje horoptera, ali to čini u vrlo kratkom vremenu. Zato je on vrlo koristan u algoritmima praćenja, gdje je bitna brzina i samo određeni dio slike, a manje koristan u primjenama koje zahtijevaju dubinu cijele promatrane scene.

2.2.3. Log-polarno mapiranje

Log-polarno mapiranje (engl. *Log-polar mapping, LPM*) preslikava sliku u kartezij-skom sustavu u sliku čije su osi ekscentričnost (ξ), tj. udaljenost od centra slike i kut (η) određenog piksela u odnosu na x os. Logaritmom nad ξ dobivamo sliku koja je u središtu oštra, a prema krajevima sve više zamućenija. Primjer log-polarnog mapiranja je na slici 2.4. Ovakvo mapiranje vrlo je slično ljudskom vidu, gdje je vid najoštrij u središtu, gdje je *fovea centralis*. Osim što ima povezanosti s ljudskim vidom, log-polarno mapiranje ima i sljedeće prednosti[21]:

- Daje vrlo dobar omjer širokog vidnog polja, visoke vizualne rezolucije i male količine podataka. Fokusom na oštrinu u sredini možemo zanemariti detalje na rubovima, čime smanjujemo količinu informacija za obradu (čak i do 30 puta manja količina podataka)

³Vršimo detekciju vertikalnih rubova, a ne horizontalnih, jer se disparitet neće izraziti na horizontalnim rubovima koliko na vertikalnim.



Slika 2.4: Primjer slike u kartezijskom sustavu (lijevo) i slike u log-polarnom sustavu (desno). Desna slika je preslikana nazad u kartezijski sustav radi lakše interpretacije.

- Biološka vjerodostojnost - Mapiranje oponaša retino-kortikalno mapiranje u vizualnom sustavu sisavaca.
- "Invarijanca" na rotaciju i skaliranje - Kada se originalna slika skalira ili rotira oko centra, uzorci na log-polarnoj slici se samo transliraju.

Mapiranje određenog piksela u log-polarne koordinate vrši se na sljedeći način:

$$\xi = \log(\sqrt{x^2 + y^2})$$

$$\eta = \text{atan2}(y, x)$$

gdje su x i y koordinate piksela na kartezijskoj slici. Primjer log-polarne slike dobivene ovim postupkom je na slici 2.5.



Slika 2.5: Log-polarna slika dimenzija 100x100. Očito je znatno smanjenje veličine u odnosu na originalnu sliku koja je bila dimenzija 720x704 [22]

Potrebno je naglasiti operator detekcije rubova i operator translacije, koji su važni u postupku praćenja. Jednostavna translacija nad log-polarnom slikom nije jednaka tran-

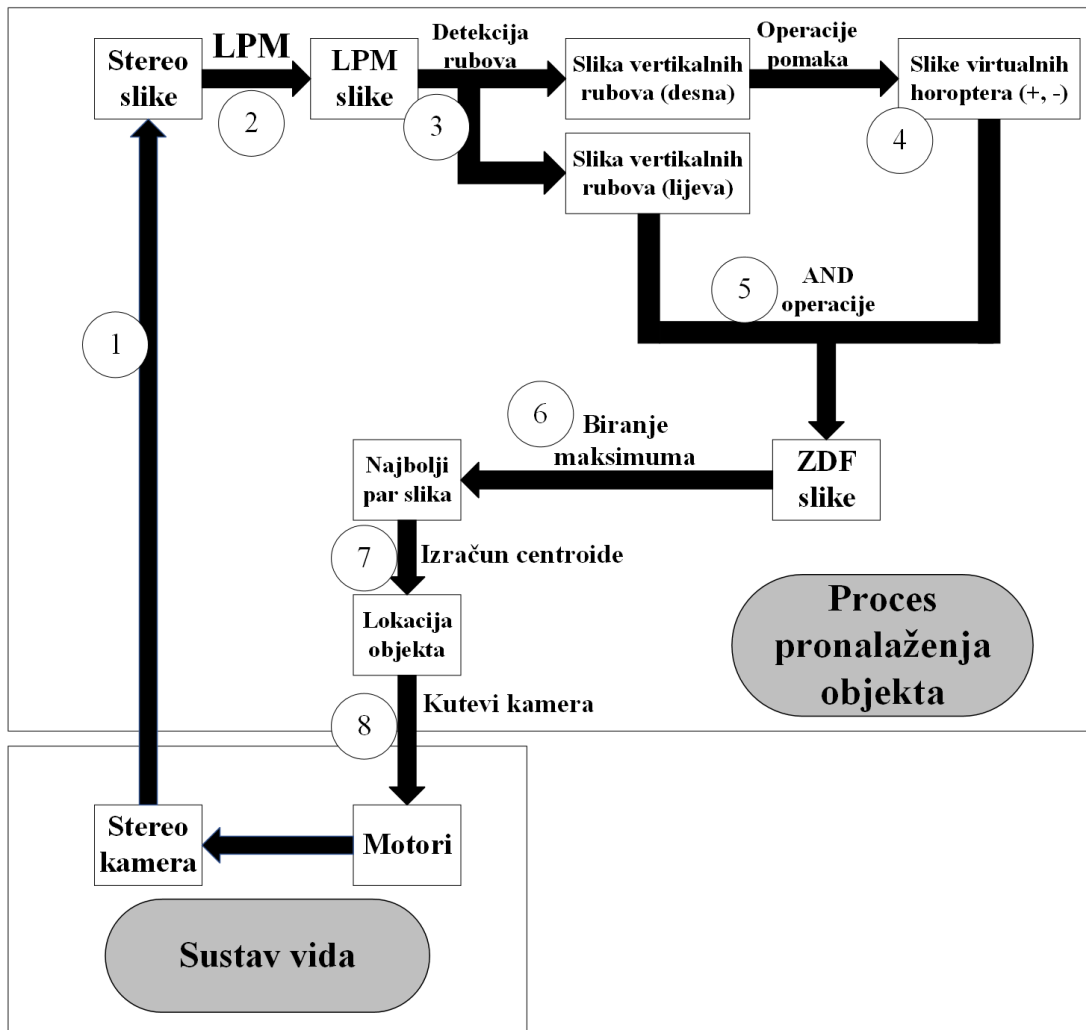
slaciji u kartezijskoj slici, a slično vrijedi i za detekciju rubova. Naivan pristup bio bi preslikati log-polarnu sliku nazad u kartezijski sustav, obaviti translaciju ili detekciju, te ponovno preslikati sliku u log-polarnu. Ovime ne dobivamo na smanjenju dimenzionalnosti slike i time na ubrzanju čitavog procesa, već samo još više usporavamo proces pretvorbama.

U [23] objašnjen je način stvaranja grafa povezanosti nad preslikanim područjima te izvođenje navedenih operacija nad samim grafom, umjesto nad slikom. Ovo je korišteno u [14] te pokazuje jako dobre rezultate. Ovakav način operacija nad log-polarnom slikom nije implementiran u ovome radu, te zbog toga ne ulazimo u detalje postupka. U ovome radu je korištena naivna metoda - iako nema ubrzanja smanjenjem veličine podataka, log-polarno mapiranje se koristi samo kao dokaz koncepta.

2.2.4. Pregled algoritma praćenja aktivnim stereo sustavom

U ovome dijelu napraviti ćemo pregled algoritma praćenja aktivnim stereo sustavom s log-polarnim mapiranjem i filterom nultog dispariteta. Prikaz čitavog procesa je na slici 2.6.

Par stereo slika uzima se od sustava za viziju (1). Nakon toga se vrši preslikavanje slika u log-polarne (2) te se na njima vrši detekcija vertikalnih rubova (3). Od desne stereo slike generiramo niz pomaknutih slika koje će uz originalnu lijevu graditi virtualne horoptere (4). Nakon toga nad originalnom lijevom slikom i nizom desnih pomaknutih slika vršimo AND operaciju - filter nultog dispariteta (5) i biramo onu s najvećom sumom (6), jer pretpostavljamo da objekt leži na tom virtualnom horopteru i daje najveću pobudu. Slijedi računanje centroide pomaknutog objekta (7) i kuteva za koje treba pomjeriti kamere (8). Kamere pomjeramo, uzimamo novi stereo par i krećemo ponovno od (1).



Slika 2.6: Dijagram procesa praćenja

3. Implementacija aktivnog stereo sustava

U ovome poglavlju napravljen je pregled implementacije sustava aktivnog stereo praćenja napravljenog za ovaj rad. Prvo će biti napravljen pregled hardverskih komponentata i komponentata programske potpore. Nakon toga detaljnije je pojašnjena programska implementacija procesa. Sav kod i 3D objekti dostupni su na <https://github.com/TinMisic/ActiveStereo>.

3.1. Oblikovanje sustava i specifikacije

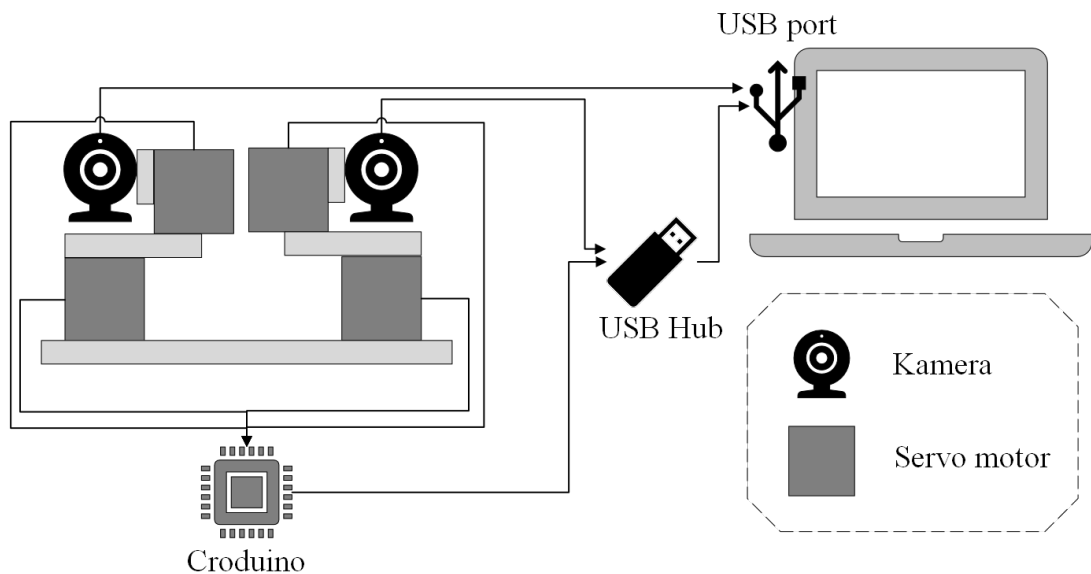
Sustav je oblikovan u operacijskom sustavu Linux, distribucija Pop!_OS 22.04 LTS. 3D objekti su modelirani u FreeCAD-u[24], a sav kod je napisan u jeziku Python, verzija 3.9.12.

3.1.1. Hardverske komponente

Za stereo kamere su korištene dvije obične web-kamere, Exis kamere proizvođača Trust [25] razlučivosti 640 x 480, dok su za motore korišteni servo motori RDS3135 [26]. Korišten je mikrokontroler Croduino Basic3¹ za upravljanje servo motorima. Za napajanje servo motora korištena je Litij-polimer baterija s dvije ćelije, a kasnije jedinica za napajanje. Sva obrada slike i upravljanje vrši se na osobnom računalu, dok Croduino služi za slanje signala servo motorima. Dijagram sustava je na slici 3.1.

Croduino je spojen s osobnim računalom serijskom vezom putem USB-a, a njime je moguće upravljati protokolom Firmata[27], koji je namijenjen za komunikaciju između mikrokontrolera i računala domaćina, preko programskog sučelja. U program-

¹Tijekom pisanja rada više nije bila dostupna ova pločica. Soldered Electronics, prethodno E-Radionica, sada proizvode slične pločice pod nazivom Dasduino, <https://soldered.com/categories/dasduino-arduino/>.



Slika 3.1: Dijagram aktivnog stereo-vizualnog sustava

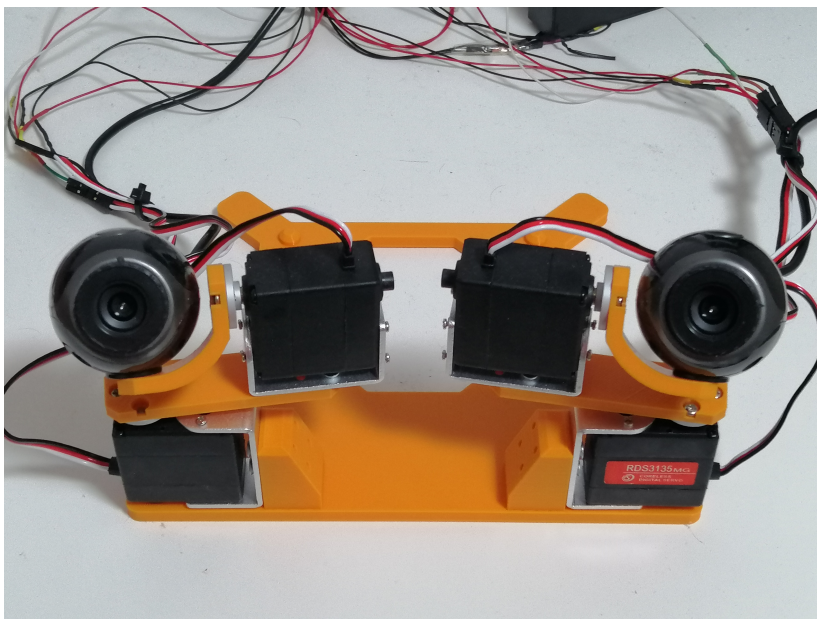
skom jeziku Python korišten je paket `pyfirmata`[28], verzija 1.1.0. Ovako se sav kod izvršava na osobnom računalu, u jednom programskom jeziku.

Kamere su spojene izravno na računalo, tj. jedna izravno na jedan USB priključak (engl. *USB port*), a druga zajedno s Croduinom na USB čvorište (engl. *USB Hub*), pa onda na drugi USB priključak. Sustav je spojen na ovakav način zbog ograničenog broja USB priključaka na osobnom računalu i zato što kamere moraju biti spojene na različite priključke kako bi mogle istovremeno snimati scenu². Kalibracija stereo kamera obavljena je korištenjem paketa [29].

Motora i kamere su povezani plastičnim dijelovima koji su modelirani u FreeCAD-u te isprintani 3D printerom. Dijelovi su modelirani tako da koriste minimalnu količinu plastike. Također je osigurano da osi oba motora zadužena za rotacije jedne kamere prolaze kroz središte projekcijske ravnine kamere, tj. senzora. Ovo olakšava izračun pozicije jedne kamere u odnosu na drugu, jer je pomak između njih uvijek jednak (u ovome slučaju 18cm), dok se relativna rotacija lako može odrediti iz kuteva na servo motorima. Kamere, motori i plastični dijelovi vidljivi su na slici 3.2.

Sustav ne sadrži nikakve enkodere ili potenciometre (osim onih u samim servo motorima) preko kojih možemo odrediti stvarnu poziciju servo motora. Pretpostavljeno je da su motori zakrenuti onoliko koliko je od njih traženo, tj. da vrijednosti zakreta u memoriji računala odgovaraju stvarnim zakretima. U redu je napraviti ovu pretpostavku, jer motori nisu pod velikim opterećenjem koje bi uzrokovalo neželjeni pomak

²Ovo je zbog ograničenja brzine prijenosa podataka na jednome USB priključku.



Slika 3.2: Izgled stvarnog sustava

njihovih osovina.

3.1.2. Komponente programske potpore

Slijedi detaljniji opis pojedinih ključnih komponenata programske potpore. Naglašeni su samo najvažniji razredi i funkcije pojedinih modula. Osim navedenih modula korišteni su standardni Python moduli, a među njima su najvažniji OpenCV[30] (verzija 4.6.0.66.) i NumPy[31] (verzija 1.21.5). OpenCV je korišten za sve operacije računalnog vida, dok je NumPy korišten za jednostavne manipulacije slikama i linearnu algebru.

Modul camera

Modul `camera` definira razred `Camera` koji nasljeđuje razred `threading.Thread`. Njime se pokreće nova dretva koja prima snimak jedne kamere. Konstruktor uzima putanju do datoteke s intrinzičnom matricom kamere i indeks kamere preko kojega se pokreće objekt za video snimak modula `opencv`. Pri pokretanju dretve pokreće se objekt video snimka i kamera kreće sa snimanjem. Najnovija slika se nakon uklanjanja distorzije sprema u podatkovni član `buffer`, odakle joj se može pristupiti pozivom metode `get_frame` iz glavne dretve.

Otvaranje dva objekta video snimke u glavnoj dretvi stvaralo bi kašnjenje između dvaju slika preveliko za rad sustava u stvarnom vremenu. Zbog toga je potrebno po-

krenuti svaki objekt u zasebnoj dretvi, kako bi se osiguralo paralelno uzimanje slika.

Iz glavne dretve je moguće pokrenuti više dretvi `Camera` (ako to hardver dopušta) i istovremeno primati snimke obje kamere bez prevelikih kašnjenja između dva snimka. Glavna dretva pribavlja slike pozivom ranije navedene metode `get_frame`, te je dužna pozvati metodu `stop` pojedine dretve nakon što joj snimci više nisu potrebni. Metoda `stop` postavlja događaj zaustavljanja koji prekida glavnu petlju dretve `Camera` i sigurno raspušta objekt video snimke.

Modul `util`

Modul `util` sadrži razrede i funkcije potrebne u radu sustava. Definirani su razredi mapiranja slike, njene translacije i detekcije vertikalnih rubova. Neke od važnijih funkcija su one za generiranje pomaknutih slika korištenih za virtualne horoptere, odabir najboljeg horoptera i ažuriranje kuteva na servo motorima.

Apstraktni razred `ImgMap` oblikovan je kao oblikovni obrazac `Strategija` i on definira mapiranje nad ulaznom slikom. Članske metode su `map`, koja vrši mapiranje nad ulaznom slikom, te `inv` koja vrši inverzno mapiranje ulazne slike. Konkretni razredi koji nasljeđuju `ImgMap` su `IdentityMap` i `LogPolarMap`. `IdentityMap` je jednostavna funkcija identiteta na ulaznu sliku, dok `LogPolarMap` vrši log-polarno mapiranje funkcijom `cv2.warpPolar` iz modula `opencv`. U konstruktoru `LogPolarMap` dodatno je potrebno definirati oblik ulazne i izlazne slike.

Apstraktni razred `EdgeDetector` također je `Strategija`, a sadrži samo metodu `detect`. Konkretni razredi koji su implementirani su `SobelCartesian` i `SobelPseudoLPM`. `SobelCartesian` obavlja detekciju vertikalnih rubova Sobelovim operatorom nad slikom u kartezijskom sustavu. `SobelPseudoLPM` je naivna implementacija detekcije vertikalnih rubova na log-polarnoj slici. On prvo vrši transformaciju log-polarne slike nazad na kartezijsku, pronalazi vertikalne rubove, te vraća sliku nazad u log-polarnu. Ovime nije ostvarena ušteda na obradi podataka koja bi bila ostvarena operacijom nad samom log-polarnom slikom, već samo služi kao dokaz koncepta.

Apstraktni razred `ShiftOp` također je `Strategija` koja definira horizontalni pomak ulazne slike. Deklarirana je samo funkcija `shift` koja obavlja pomak ulazne slike za dani broj piksela. Konkretni razredi koji nasljeđuju `ShiftOp` su `ShiftCartesian` i `ShiftPseudoLPM`. `ShiftCartesian` obavlja jednostavni horizontalni pomak nad kartezijskom slikom, dok `ShiftPseudoLPM` radi na način sličan `SobelPseudoLPM`, pretvarajući sliku u kartezijsku, obavljajući pomak, i vraćajući ju nazad u log-polarnu

sliku.

Funkcija `generateShifts` generira željeni broj pomaka lijevo/desno za željeni broj koraka. Uz originalnu sliku potrebno je predati i objekt tipa `ShiftOp`.

Funkcija `getMaxZDF` vraća pomak za koji je ostvarena najveća suma elemenata pomaknute slike. Ovako se određuje horopter koji je najprikladniji. Kao ulaz se prima mapa čiji su ključevi pomaci, a vrijednosti pomaknute slike, a kao izlaz vraća ključ slike s najvećom sumom.

Modul `servo_controller`

U modulu `servo_controller` definirana je samo funkcija za izračun kuteva za koje se moraju zakrenuti motori kako bi određena točka bila u središtu slike. Kao ulaz se uzimaju centroide objekta na lijevoj i desnoj slici, te intrinzične matrice lijeve i desne kamere. Jednostavnom trigonometrijom se računa kut zakreta na svakoj osi, poznavanjem koordinata centroide i fokusa kamere (izraženog u pikselima). Funkcija osigurava da je vertikalni zakret servo motora jednak, čime se osigurava koplanarnost projekcijskih pravaca dvaju kamera.

Modul `motion_detection`

Najvažnija funkcija u modulu `motion_detection` jeste `centroidOfDiff` koja računa centroidu na osnovu razlike dvaju uzastopnih slika. Ovime se ostvaruje jednostavna detekcija pokreta, koja je bitna u postupku inicijalizacije sustava.

3.2. Cjevovod sustava i upravljanje

U ovome dijelu opisan je rad glavnog modula, `object_tracking`. U većini prethodno spomenutih pristupa pretpostavljeno je da su kamere već fokusirane na neki objekt, nakon čega odmah kreće praćenje. U ovoj implementaciji inicijalno fokusiranje na objekt vrši se jednostavnom detekcijom pokreta. Nakon što je primijećeno kretanje, sustav prelazi u stanje praćenja. Ako se u stanju praćenja kamere ne pomjere nakon određenog vremena, objekt se smatra izgubljenim te se ponovno prelazi u stanje inicijalizacije.

3.2.1. Inicijalizacija sustava i upravljanje

Pri pokretanju modula pokreću se dretve dvaju kamera. Nakon toga su definirani mapiranje, detektor rubova i operator pomaka. Slijedi inicijalizacija servo motora i njihovo postavljanje na neutralnu poziciju (kamere su paralelne i svi kutevi na motorima su nula). Nakon toga se ulazi u glavnu petlju programa.

Ako je zastavica za inicijalizaciju postavljena, ulazi se u novu petlju u kojoj vršimo detekciju pokreta. Ako je razlika između trenutne slike i prethodne dovoljno velika da se smatra pokretom, te ako je taj pokret primijećen na obje kamere, postavljaju se kutevi na motorima te se izlazi iz inicijalizacijske petlje. Sada bi kamere trebale biti usmjerene prema području scene gdje je primijećeno kretanje. Servo motori se postavljaju na zadane kuteve jednostavnim pozivom funkcije `write` pojedinog pin-a objekta `board` koji odgovara Croduino pločici. Funkciji `write` prosljeđujemo vrijednost kuta izraženu u stupnjevima. Na ovaj način se ostvaruje jednostavno upravljanje motorima preko protokola Firmata.

3.2.2. Određivanje pozicije objekta

Računanje centroide i kuteva

Centroidu projekcije objekta na sliku računamo koristeći momente slike. Moment diskretne dvodimenzionalne funkcije $I(x, y)$ (u ovome slučaju to je slika) reda $(i + j)$ definiran je kao:

$$M_{ij} = \sum_x \sum_y x^i y^j I(x, y)$$

M_{00} daje površinu binarne slike, tj. sumu svih sivih razina crno-bijele slike. M_{10} predstavlja moment prvoga reda distribucije po x osi, a analogno vrijedi za y os i moment M_{01} . Centroidu površine dobivamo na sljedeći način:

$$\{\bar{x}, \bar{y}\} = \left\{ \frac{M_{10}}{M_{00}}, \frac{M_{01}}{M_{00}} \right\}$$

Nakon što znamo centroidu objekta, želimo saznati za koje relativne kuteve trebamo zakrenuti kamere tako da centroida objekta bude u središtu slike. Njih lako možemo izračunati ako poznamo fokuse na x i y osima pojedine kamere. Te fokuse lako možemo izlučiti iz intrinzične matrice kamere dobivene njenom kalibracijom. Jednako lako računamo kuteve oko x i y osi kamere:

$$\phi_y = \frac{c_x - \bar{x}}{f_x}$$

$$\phi_x = \frac{c_y - \bar{y}}{f_y}$$

gdje je $\{\bar{x}, \bar{y}\}$ centroida objekta, $\{c_x, c_y\}$ središte slike, a f_x i f_y fokusi kamere na x i y osima, izraženi u pikselima. Kuteve računamo za lijevu i desnu sliku te dobivamo četiri kuta potrebna za rotaciju motora. Srednjom vrijednošću $\bar{\phi}_x$ između ϕ_{xL} i ϕ_{xR} osiguravamo da su projekcijski pravci obiju kamera koplanarni.

Odabir najprikladnijeg horoptera

Kao izlaz filtera nultog dispariteta dobivamo binarne slike na kojima su označeni rubovi koji imaju jednaku poziciju na slici. Kao najprikladniji virtualni horopter biramo onaj čija je suma, tj. moment M_{00} najveći, jer pretpostavljamo da upravo ti rubovi pripadaju objektu na koji smo fokusirani i koji leži na tom virtualnom horopteru.

Nakon toga nam je bitan samo virtualni pomak kojim smo dobili taj virtualni horopter, te će on ulaziti u stvarni pomak kamere.

3D pozicija objekta na osnovu orijentacije kamera

Poznavanje orijentacije kamera i dimenzija sustava lako možemo izračunati 3D poziciju objekta bez korištenja kompleksnih stereo algoritama. Pošto smo osigurali da su projekcijski pravci kamera koplanarni, oni se moraju sjeći tamo gdje se nalazi objekt³. Sjecište dvaju pravaca određujemo na osnovu ϕ_{yL} i ϕ_{yR} .

Izračun kreće na ravnini koja je za $\bar{\phi}_x$ rotirana oko x osi. Sjecište P uzimamo kao sjecište dvaju pravaca definiranih z osima kamera (osi koje su okomite na njihove projekcijske ravnine) i točkama u kojima se nalaze kamere (središte projekcijske ravnine). Ova točka se nalazi u koordinatnom sustavu jedne od kamera. Točku dalje rotiramo oko x osi za kut $\bar{\phi}_x$. Nakon toga ju pomjeramo u koordinatni sustav čitavog sustava translacijom.

³Naravno, oni se ne moraju sjeći, ali to je samo slučaj ako je objekt jako daleko ili u beskonačnosti.

4. Praćenje gibajućeg objekta

U ovome poglavlju pokazana je funkcionalnost implementiranog sustava. Napravljena su četiri eksperimenta kako bi se ocijenila efikasnost i točnost pojedinih metoda. Demonstracija praćenja i snimci eksperimenata nalaze se u YouTube listi <https://youtube.com/playlist?list=PLaB-Uh5PTKZ6NRGsafHZIQWY19qwmFHi7>. Prvo je dan opis eksperimentalne postave i eksperimenata te nakon toga prikaz rezultata i diskusija.

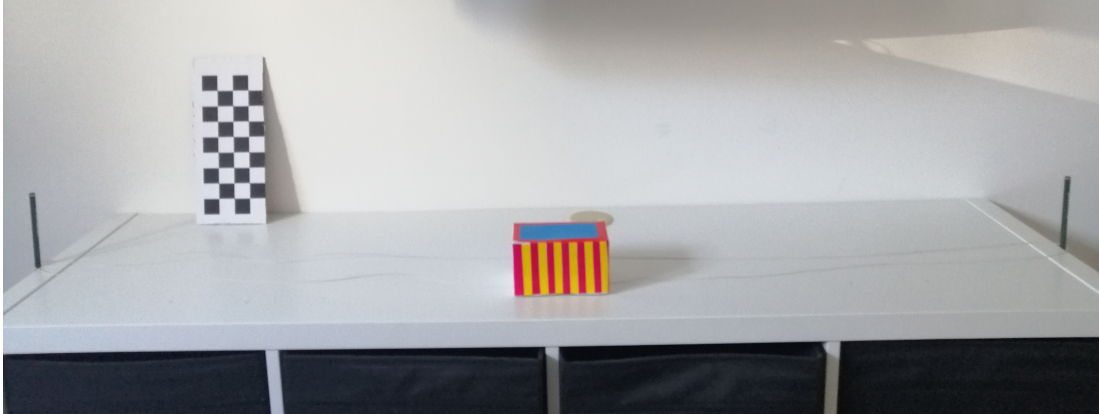
4.1. Eksperimentalna postava

4.1.1. Opis postave

Ukupno su urađena četiri eksperimenta:

- Praćenje gibajućeg objekta na kartezijskoj slici
 - Bez pozadinske smetnje
 - S pozadinskom smetnjom u obliku mirujućeg uzorka
- Praćenje gibajućeg objekta na log-polarnoj slici
 - Bez pozadinske smetnje
 - S pozadinskom smetnjom u obliku mirujućeg uzorka

Gibajući objekt je prizma s uzorkom koji sadrži mnogo vertikalnih rubova. Objekt kreće od sredine postolja, giba se desno do krajnje desne pozicije, onda lijevo preko početne pozicije do krajnje lijeve, te nazad na početnu poziciju. Same krajnje pozicije i pozicija sustava u odnosu na početnu poziciju prizme se razlikuju između kartezijskih i log-polarnih inačica eksperimenata. Razlike su napravljene kako bi se dobili što jasniji rezultati, ali učinak uvedenih razlika nije zanemaren. U inačicama eksperimenata sa smetnjom kao pozadinska smetnja korišten je dio kalibracijskog uzorka, kako bi se pokazao utjecaj vertikalnih rubova u pozadini. Slika postave iz perspektive sustava je na slici 4.1.



Slika 4.1: Eksperimentalna postava s objektom na početnoj poziciji i uzorkom u pozadini

Svrha različitih eksperimenata je ispitati sljedeće:

- Koje su prednosti i nedostaci uporabe log-polarnih slika u odnosu na kartezijske?
- Je li sustav uporabljiv u stvarnom vremenu?
- Kakav utjecaj ima pozadina i smetnje u pozadini na praćenje objekta?
- Koliko često određena inačica sustava gubi objekt tijekom praćenja?
- Koliko je točna estimirana trajektorija objekta u odnosu na stvarnu?

4.1.2. Eksperimenti na kartezijskim slikama

U oba eksperimenta objekt se kreće sljedećom putanjom u koordinatnom sustavu aktivnog stereo sustava (koordinate su izražene u metrima):

1. Početak (0.0, 0.1, 1.0)
2. Krajnja desna pozicija (0.4, 0.1, 1.0)
3. Krajnja lijeva pozicija (-0.4, 0.1, 1.0)
4. Kraj (0.0, 0.1, 1.0)

U inačici eksperimenta bez pozadinske smetnje pozadina je potpuno prazna, dok se u inačici s pozadinskom smetnjom kalibracijski uzorak nalazi na poziciji (-0.7, 0.0, 1.2).

4.1.3. Eksperimenti na log-polarnim slikama

U eksperimentu bez pozadinske smetnje objekt se giba sljedećom putanjom:

1. Početak (0.0, 0.1, 0.55)
2. Krajnja desna pozicija (0.4, 0.1, 0.55)
3. Kraj (0.0, 0.1, 0.55)

U eksperimentu s pozadinskom smetnjom objekt se giba sljedećom putanjom:

1. Početak (0.0, 0.1, 0.55)
2. Krajnja desna pozicija (0.3, 0.1, 0.55)
3. Kraj (0.0, 0.1, 0.55)

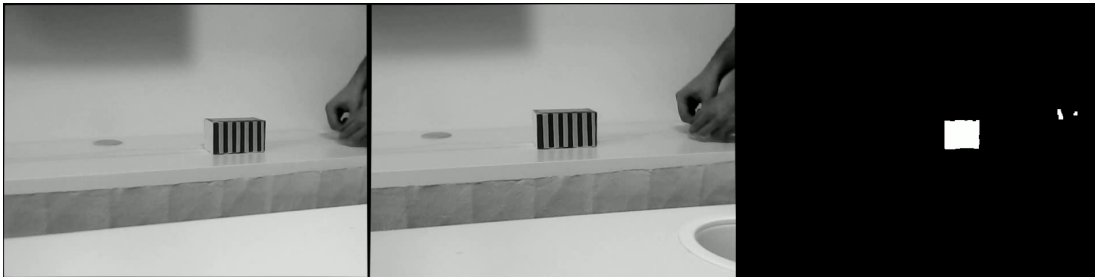
Kalibracijski uzorak koji predstavlja smetnju nalazi se na poziciji (0.3, 0.0, 0.75).

4.2. Rezultati

4.2.1. Eksperiment 1: Gibanje na kartezijskoj slici bez pozadinske smetnje

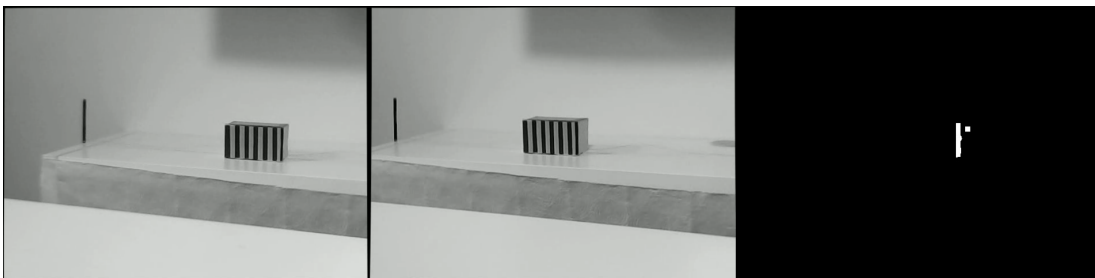
Snimak eksperimenta je na sljedećoj poveznici: <https://youtu.be/LMisw3QWEts>

Praćenje gibanja objekta se odvija u stvarnom vremenu i sustav bez problema prati objekt nakon mijenjanja smjera gibanja. Iako je većina pozadine bez većih vertikalnih rubova koji bi mogli zbuniti sustav, smetnje svejedno postoje, kao što je vidljivo na slici 4.2. Unatoč smetnji, sustav se oporavlja i nastavlja s praćenjem.



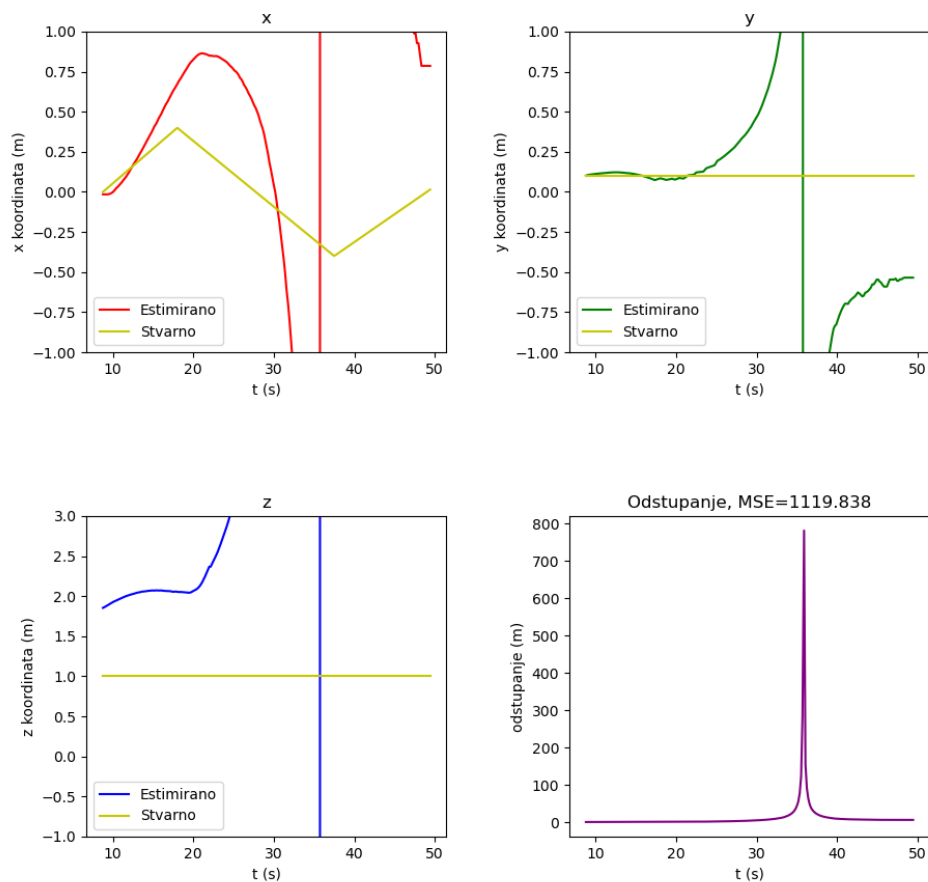
Slika 4.2: Eksperiment 1: Smetnja je vidljiva kao odziv na izlazu filtera nultog dispartiteta(desno). Slika lijeve kamere je lijevo, a desne u sredini.

Nakon povratka promjene smjera iz krajnje lijeve pozicije postepeno se gubi obris objekta, slika 4.3. Najvjerojatniji uzrok je pomjeranje virtualnih horoptera na neoptimalan način, što je uzrokovalo sve manji broj preklapanja vertikalnih rubova. Nakon nekog vremena između krajnje lijeve točke i početne točke gibanja gubi se objekt i praćenje se ne oporavlja do kraja gibanja.



Slika 4.3: Eksperiment 1: Postepeno gubljenje objekta i smanjenje odziva filtera nultog dispartiteta(desno). Slika lijeve kamere je lijevo, a desne u sredini.

Na slici 4.4 nalaze se grafovi koji uspoređuju estimiranu poziciju po svakoj osi sa stvarnom pozicijom, te graf odstupanja estimirane pozicije od stvarne. Srednja



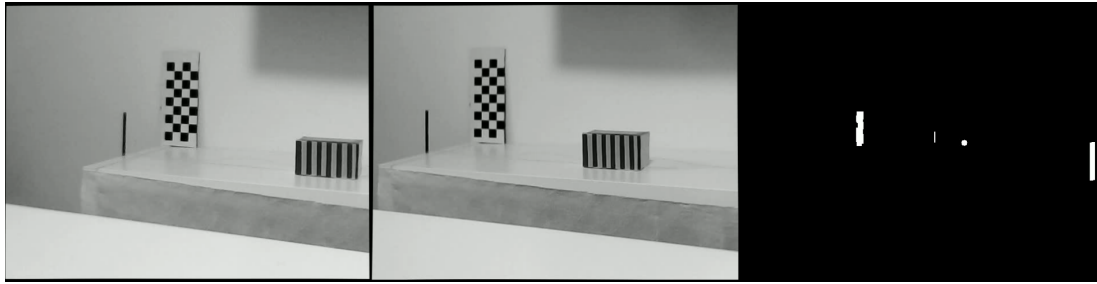
Slika 4.4: Eksperiment 1: Prikazi estimiranih i stvarnih pozicija, te grafa odstupanja po vremenu uz srednju kvadratnu pogrešku.

kvadratna pogreška je relativno visoka zbog odstupanja koje nastaje pri postepenom gubljenju opisanom ranije.

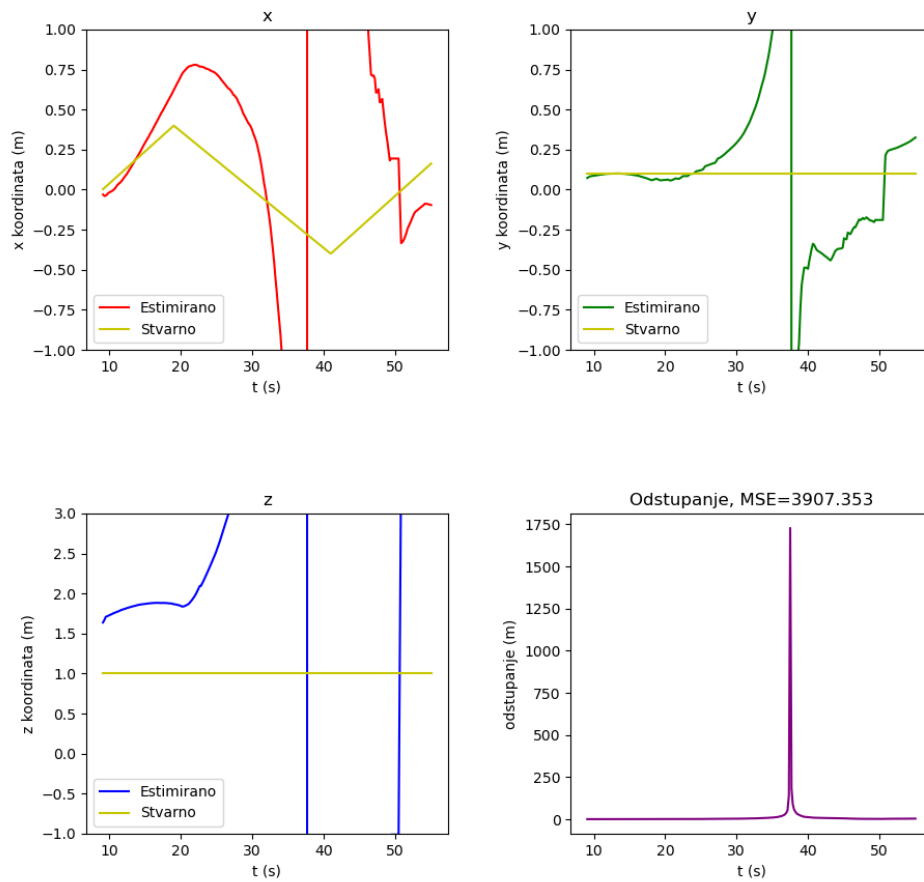
4.2.2. Eksperiment 2: Gibanje na kartezijskoj slici s pozadinskom smetnjom

Snimak eksperimenta je na sljedećoj poveznici: <https://youtu.be/qrpOhCGjYy0>

Praćenje gibanja se odvija u stvarnome vremenu, kao i u prvome eksperimentu. Male pozadinske smetnje postoje, ali su dovoljno male i moguće je oporaviti praćenje. Zanimljiv je trenutak kada kalibracijski uzorak ulazi u vidno polje (slika 4.5). Zbog toga fokus ostaje između gibajućeg objekta i mirujuće smetnje što kasnije uzrokuje gubljenje objekta. Sustav uspijeva oporaviti praćenje prije završetka gibanja.



Slika 4.5: Eksperiment 2: Pozadinska smetnja i njen utjecaj na filter nultog dispariteta (desno).



Slika 4.6: Eksperiment 2: Prikazi estimiranih i stvarnih pozicija, te grafa odstupanja po vremenu uz srednju kvadratnu pogrešku.

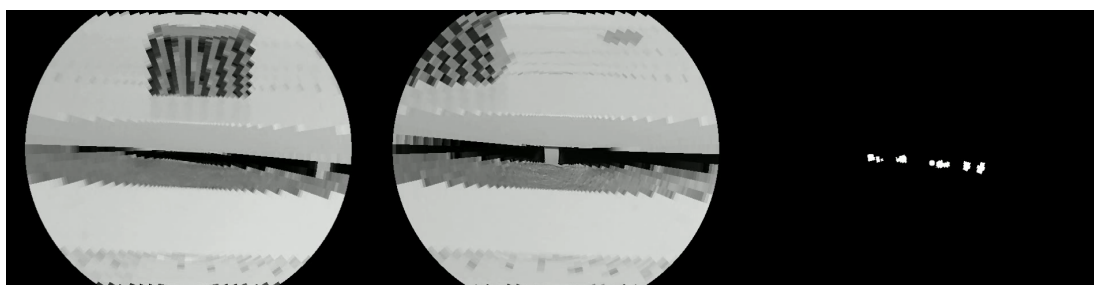
Na slici 4.6 nalaze se grafovi koji uspoređuju estimiranu poziciju po svakoj osi sa stvarnom pozicijom, te graf odstupanja estimirane pozicije od stvarne. Srednja kvadratna pogreška je još veća u usporedbi s prvim eksperimentom zbog smetnje u pozadini.

4.2.3. Eksperiment 3: Gibanje na log-polarnoj slici bez pozadinske smetnje

Snimak eksperimenta je na sljedećoj poveznici: <https://youtu.be/OUQ3FyO-Ink>

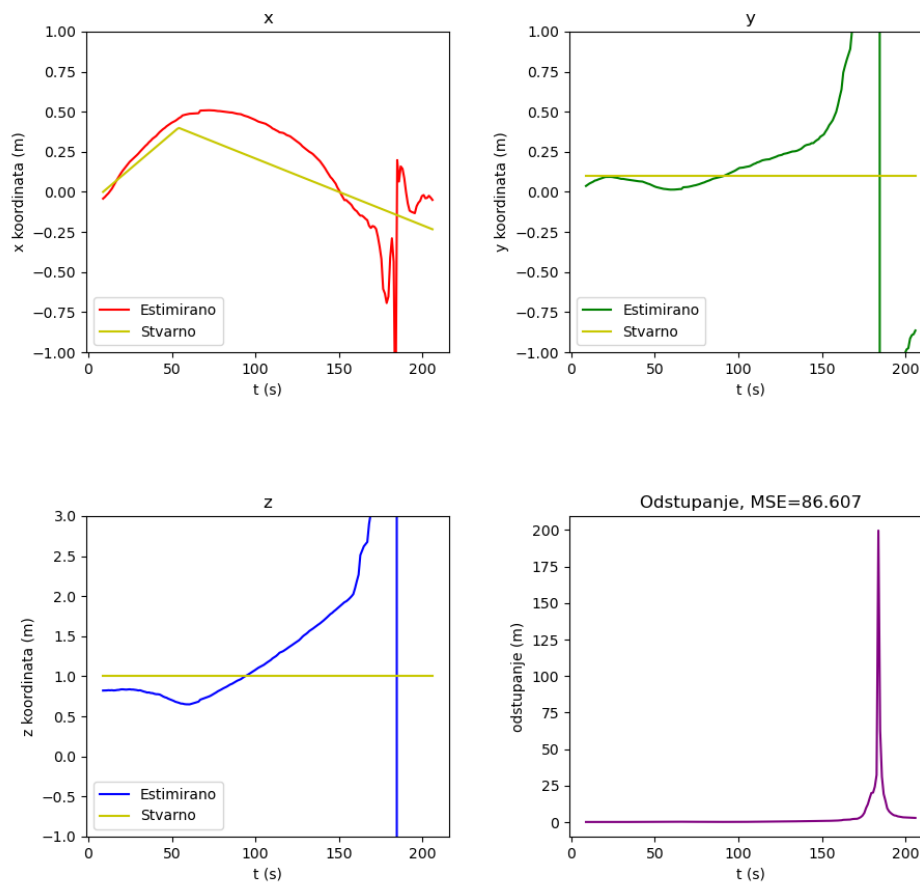
Sustav ne radi u stvarnom vremenu, zbog velikog broja pretvaranja log-polarne slike u kartezijsku i obrnuto. Pretvaranje je potrebno za svaku operaciju pomicanja i detekcije rubova, što se često javlja u generiranju virtualnih horopera. Rješenje je predloženo u odjeljku 4.3.

Treba napomenuti kako je inačica s log-polarnim slikama izrazito osjetljiva na pozadinske smetnje, te je zbog toga putanja objekta pomjerena bliže sustavu. Ove smetnje često nastaju ne zbog stvarnih vertikalnih rubova u pozadini, već zbog lažnih vertikalnih rubova nastalih mapiranjem log-polarne slike nazad u kartezijsku, što je dio detekcije vertikalnih rubova. Primjer lažnih rubova koji su uzrokovali gubitak praćenja objekta je na slici 4.7. Uzrok lažnih rubova i njihovo rješenje predloženo je u odjeljku 4.3.



Slika 4.7: Eksperiment 3: Smetnja u obliku lažnih vertikalnih rubova i gubitak objekta koji je van središnjeg oštrog dijela.

Sustav ostvaruje veliku točnost u odnosu na kartezijske slike, sve do pomjeranja fokusa na lažne vertikalne rubove. Ovoj točnosti vrlo vjerojatno pridonosi manja udaljenost objekta od sustava, ali potrebna su dodatna ispitivanja kako bi se utvrdila stvarna uloga blizine u povećanoj točnosti. Prikaz estimiranih i stvarnih pozicija nalazi se na slici 4.8. Srednja kvadratna pogreška je za red veličine manja nego u kartezijskim inačicama eksperimenta.



Slika 4.8: Eksperiment 3: Prikazi estimiranih i stvarnih pozicija, te grafa odstupanja po vremenu uz srednju kvadratnu pogrešku.

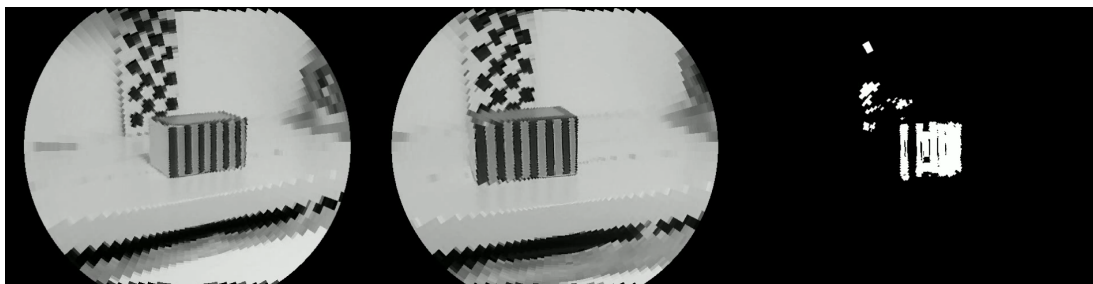
4.2.4. Eksperiment 4: Gibanje na log-polarnoj slici s pozadinskom smetnjom

Snimak eksperimenta je na sljedećoj poveznici: <https://youtu.be/jrD0EFVjSvY>

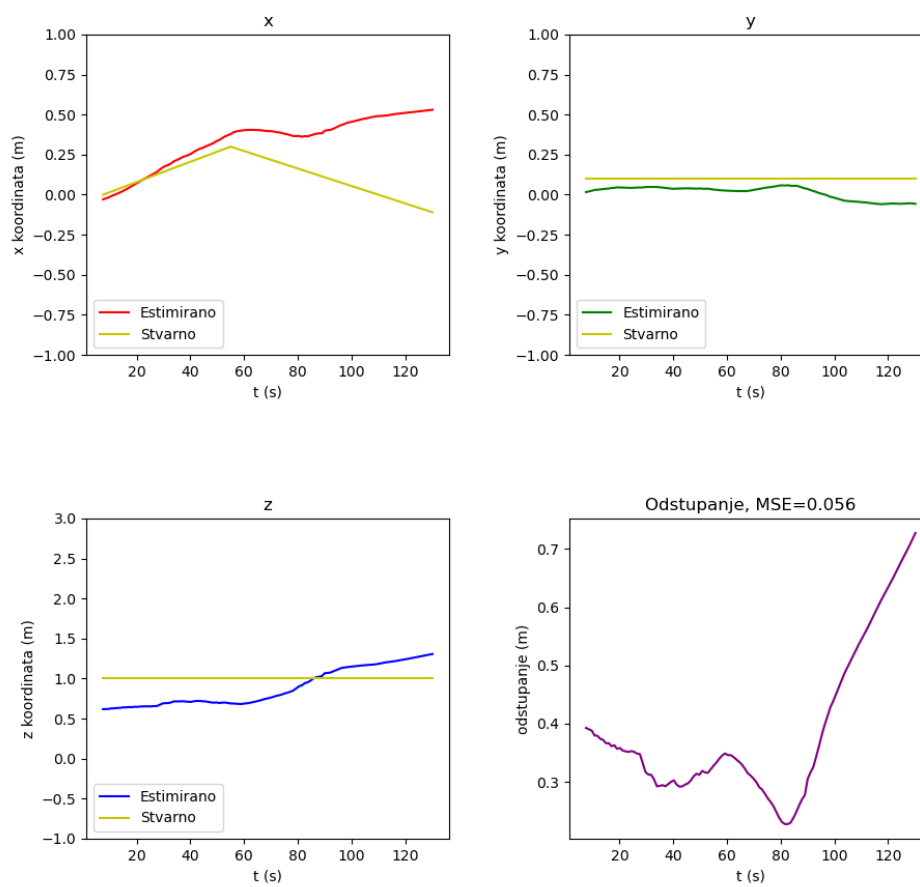
Sustav ne radi u stvarnom vremenu iz istih razloga kao i u trećem eksperimentu. Isto vrijedi i za detekciju lažnih vertikalnih rubova. Mogući uzrok i rješenje predloženi su u odjeljku 4.3.

Pri prvom pojavljivanju kalibracijskog uzorka vidljivo je filtriranje smetnji van područja veće razlučivosti, što je uzrokovano log-polarnim mapiranjem(slika 4.9). Unatoč tome, pozadinska smetnja prevladava i gubi se fokus na objektu. Uzrok gubitka fokusa je najvjerojatnije zbog toga što se smetnja s vremenom uspije naći u oštrm dijelu slike, te zbog lažnih vertikalnih rubova u pozadini.

Kao i u trećem eksperimentu, točnost je znatno veća nego u inačicama s kartezijskim



Slika 4.9: Eksperiment 4: Smanjen odziv pozadinske smetnje zbog log-polarnog mapiranja.



Slika 4.10: Eksperiment 4: Prikazi estimiranih i stvarnih pozicija, te grafa odstupanja po vremenu uz srednju kvadratnu pogrešku.

skim slikama. Prikaz estimiranih i stvarnih pozicija nalazi se na slici 4.10.

4.3. Usporedba pristupa

4.3.1. Izvođenje u stvarnom vremenu

Pristup s korištenjem kartezijskih slika je moguće izvoditi u stvarnom vremenu, dok pristup s log-polarnim slikama prati samo jako spora gibanja, te je efektivno neuporabljiv u stvarnom vremenu. Ova sporost nije uzrokovana samim log-polarnim slikama, već operacijama horizontalnog pomaka i detekcije vertikalnih rubova koje su u ovome radu implementirane na "naivan" način. Ovakav "naivan" pristup za svaku operaciju zahtjeva pretvaranje log-polarne slike u kartezijsku, vršenje određene operacije, te pretvorbu nazad u log-polarnu sliku. Ovo je vrlo neučinkovito, što se još više primjećuje kada u stvaranju virtualnih horoptera zahtijevamo 45 različitih horizontalnih pomaka, kao što je slučaj u prethodnim eksperimentima.

Još jedna negativna posljedica "naivnog" pristupa je u Sobelovom detektoru vertikalnih rubova. Pretvaranje log-polarne slike u kartezijsku je diskretno i stvara lažne vertikalne rubove koje Sobelov detektor označava i ti lažni rubovi dalje utječu na odabir optimalnog virtualnog horoptera.

Oba ova problema mogu se riješiti pristupom spomenutim u [23] i implementiranim u [14]. Ovakav pristup na početku procedure stvara graf povezanosti polja(čvora) log-polarne slike koji ostaje konstantan tijekom rada, jer je sagrađen na osnovu dimenzija originalne kartezijske slike i željene log-polarne slike te ne ovisi o konkretnoj slici koja pristiže tijekom rada procedure. Sve operacije horizontalnog pomaka i detekcije rubova vrše se na grafu, a konkretne vrijednosti na poljima grafa se određuju preko tablice za pretraživanje (engl. *Lookup table, LUT*) koja se također inicijalizira na početku procedure. Ovime bi pristup s log-polarnim slikama bio ostvariv u stvarnom vremenu, te lažni vertikalni rubovi ne bi postojali.

4.3.2. Estimacija pozicije

Pristupi s log-polarnim mapiranjem daju veću točnost u estimaciji pozicije gibajućeg objekta, ali nije u potpunosti jasno koliku ulogu u tome ima stvarna blizina samog objekta te je potrebno još ispitivanja. Bolja estimacija pozicije mogla bi se postići dodavanjem enkodera ili potenciometara u sustav, čime bi bilo moguće točnije odrediti kuteve motora i time poziciju objekta.

4.3.3. Učinkovitost virtualnih horoptera

Najveći nedostatak oba pristupa nalazi se u njihovoj velikoj osjetljivosti na rubove u pozadini. Kako bi bilo koji od eksperimenata dao ikakve razumljive rezultate, pozadina je trebala biti potpuno monotona i bez ikakvih rubova. Za oba pristupa je pri implementaciji bilo potrebno empirički podesiti broj virtualnih horoptera i broj piksela za koji se pomiče desna slika. Također je bilo potrebno "napuhati" binarne slike s rubovima morfološkom operacijom dilatacije kako bi ti rubovi bili dovoljno široki da ih filter nultog disparteta može pronaći. Ovdje blagu prednost ima pristup s log-polarnim mapiranjem, ali ni on nije otporan na pozadinske smetnje.

Glavni uzrok ovoga je korištenje vertikalnih rubova za identifikaciju objekta. Oni su previše jednostavni kako bi identificirali objekt te zbog toga mnogi rubovi u pozadini mogu "odglumiti" objekt i time pokvariti praćenje. Čak i sam objekt može pokvariti estimaciju svoje pozicije ako ima mnoštvo vertikalnih rubova koji se na više načina mogu preklopiti, kao što je slučaj u prethodnim eksperimentima. S vremenom se pogrešna preklapanja nakupljaju i na kraju se horopter toliko pomjeri da objekt više nije na njemu.

Rješenje ovoga problema bi zadržalo ideju virtualnih horoptera kao alternativu stvarnom pomicanju kamera, ali za identifikaciju objekta ne bi koristilo vertikalne rubove. Umjesto rubova bi se koristili opisnici kao što su ORB, SIFT ili SURF kako bi se bolje izdvojio objekt od pozadine. Pretraga ovih opisnika između lijeve i desne slike bi se radila kao u klasičnim stereo algoritmima, ali samo na malom području oko centra slike, po ideji virtualnih horoptera. Također je atraktivna ideja korištenja konvolucijskih neuronskih mreža u inicijalizaciji praćenja objekta te njegovog pretraživanja u uskom području oko središta slike tijekom samog praćenja, umjesto ranije navedenih opisnika. Oba ova alternativna pristupa identifikacije objekta bila bi dosta otpornija smetnjama u pozadini.

5. Zaključak

U ovome radu napravljen je pregled pristupa praćenju gibajućih objekata aktivnim stereo sustavima koji koriste virtualne horoptere. Oblikovan je i ispitani aktivni stereo vizualni sustav koji koristi web kamere i servo motore.

Za rad u stvarnome vremenu najbolji se pokazao pristup s kartezijskim mapiranjem slika, ali log-polarna inačica sustava bi također mogla postići rad u stvarnom vremenu uz prikladnu implementaciju operatora pomaka i detektora rubova. Obje inačice pokazuju veliku osjetljivost na pozadinske smetnje, što je najvjerojatnije uzrokovano korištenjem jednostavnih vertikalnih rubova kao identifikatora predmeta. Predloženo je korištenje opisnika kao što su ORB, SIFT ili SURF te korištenje konvolucijskih neuronskih mreža kao rješenje za ovaj problem.

Točnost estimacije pozicije objekta je veća kod pristupa s log-polarnim mapiranjem, ali nije u potpunosti jasna uloga blizine samog objekta u ovoj točnosti te su potrebna dodatna ispitivanja. Također je pokazana sposobnost log-polarnog mapiranja u filtriranju dijela slike koji nije u fokusu.

Pokazana je učinkovitost metode virtualnih horoptera kao alternative stvarnom pomjeranju kamera. Nedostatci implementiranog sustava ne nalaze se u metodi virtualnih horoptera, već u implementaciji log-polarnog mapiranja i korištenju vertikalnih rubova.

Sljedeći korak bio bi implementacija grafa povezanosti za operacije nad log-polarnim slikama. Ovime bi se riješili problemi sporosti ove inačice sustava i pojave lažnih rubova. Također bi bilo dobro ispitati rad sustava s opisnicima ili neuronskim mrežama. Točnost estimacije pozicije mogla bi se poboljšati dodavanjem enkodera ili potenciometara.

6. Zahvale

Zahvaljujem mentoru prof. dr. sc. Ivanu Markoviću na podršci i pruženoj prilici. Njegovi savjeti i usmjerenja su uvijek od velike pomoći te iznimno cijenim dane prilike za istraživanje onih pitanja koja me istinski zanimaju.

Zahvaljujem mag. ing. Karlu Kolediću na pruženoj podršci i posuđenoj opremi. Bez njegove pomoći ne bih uspio završiti ovaj rad na vrijeme.

Zahvaljujem se kolegama i prijateljima sa studija na mnogim savjetima i povratnoj informaciji za ideje koje sam predstavljao. Posebno se zahvaljujem Antunu Weberu za uslugu 3D printanja dijelova potrebnih za oblikovani sustav.

Konačno, zahvaljujem svojoj obitelji i najboljem prijatelju na konstantnoj podršci i motivaciji tijekom studija i rada. Bez njih ništa od ovoga ne bi bilo moguće.

LITERATURA

- [1] A. T. Nishitani and D. F. Wolf, “Solving the monocular visual odometry scale problem with the efficient second-order minimization method,” in *2015 12th Latin American Robotics Symposium and 2015 3rd Brazilian Symposium on Robotics (LARS-SBR)*, 2015, pp. 126–131.
- [2] S. Song, M. Chandraker, and C. C. Guest, “High accuracy monocular sfm and scale correction for autonomous driving,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 4, pp. 730–743, 2016.
- [3] D. Zhou, Y. Dai, and H. Li, “Ground-plane-based absolute scale estimation for monocular visual odometry,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 2, pp. 791–802, 2020.
- [4] Y. Li, S. Wang, D. Yang, and D. Sun, “Metric online monocular slam by using a known reference,” in *2016 12th World Congress on Intelligent Control and Automation (WCICA)*, 2016, pp. 3002–3006.
- [5] A. Sharma, I. Wadhwa, and R. Kala, “Monocular camera based object recognition and 3d-localization for robotic grasping,” in *2015 International Conference on Signal Processing, Computing and Control (ISPCC)*, 2015, pp. 225–229.
- [6] S. Guo, J. Guo, and C. Bai, “Semi-direct visual odometry based on monocular depth estimation,” in *2019 IEEE International Conference on Unmanned Systems (ICUS)*, 2019, pp. 720–724.
- [7] D. H. Ballard, “Animate vision,” *Artificial Intelligence*, vol. 48, no. 1, pp. 57–86, 1991. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/0004370291900804>
- [8] S. Rougeaux, N. Kita, Y. Kuniyoshi, S. Sakane, and F. Chavand, “Binocular tracking based on virtual horopters,” in *Proceedings of IEEE/RSJ International Con-*

- ference on Intelligent Robots and Systems (IROS'94)*, vol. 3, 1994, pp. 2052–2057 vol.3.
- [9] J. Prado, L. Santos, and J. Dias, “Horopter based dynamic background segmentation applied to an interactive mobile robot,” in *2009 International Conference on Advanced Robotics*, 2009, pp. 1–6.
- [10] J. Selig, “The complex of lines from successive points and the horopter,” in *2008 IEEE International Conference on Robotics and Automation*, 2008, pp. 2380–2385.
- [11] E. Bayro-Corrochano and D. Gonzalez-Aguirre, “Human like vision using conformal geometric algebra,” in *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, 2006, pp. 1299–1304.
- [12] F. Bergholm and A. Francisco, “On the horopter and hering-hillebrand deviation,” in *Proceedings II Workshop on Cybernetic Vision*, 1996, pp. 166–171.
- [13] E. Hering, *The Theory of Binocular Vision: Ewald Hering (1868)*. Springer US, 2012. [Online]. Available: <https://books.google.hr/books?id=GDCgBQAAQBAJ>
- [14] N. Oshiro, N. Maru, A. Nishikawa, and F. Miyazaki, “Binocular tracking using log polar mapping,” in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems. IROS '96*, vol. 2, 1996, pp. 791–798 vol.2.
- [15] M. Tanaka, N. Maru, and F. Miyazaki, “3-d tracking of a moving object by an active stereo vision system,” in *Proceedings of IECON'94 - 20th Annual Conference of IEEE Industrial Electronics*, vol. 2, 1994, pp. 816–820 vol.2.
- [16] D. Coombs and C. Brown, “Real-time smooth pursuit tracking for a moving binocular robot,” in *Proceedings 1992 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1992, pp. 23–28.
- [17] J. Batista, P. Peixoto, and H. Araujo, “Real-time vergence and binocular gaze control,” in *Proceedings of the 1997 IEEE/RSJ International Conference on Intelligent Robot and Systems. Innovative Robotics for Real-World Applications. IROS '97*, vol. 3, 1997, pp. 1348–1354 vol.3.
- [18] E. Grosso and D. Ballard, “Head-centered orientation strategies in animate vision,” in *1993 (4th) International Conference on Computer Vision*, 1993, pp. 395–402.

- [19] K. Pahlavan, T. Uhlin, and J.-O. Eklundh, "Dynamic fixation [active vision]," in *1993 (4th) International Conference on Computer Vision*, 1993, pp. 412–419.
- [20] C. Brown, "Gaze controls with interactions and decays," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 20, no. 2, pp. 518–527, 1990.
- [21] V. Javier Traver and A. Bernardino, "A review of log-polar imaging for visual perception in robotics," *Robotics and Autonomous Systems*, vol. 58, no. 4, pp. 378–398, 2010. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0921889009001687>
- [22] "Marcus aurelius sculpture bust," <https://pixabay.com/photos/marcus-aurelius-sculpture-bust-6363862/>, pristupljeno 12. travnja, 2023.
- [23] R. Wallace, P. Ong, B. B. Bederson, and E. Schwartz, "Space variant image processing," *International Journal of Computer Vision*, vol. 13, pp. 71 – 90, 1994// 1994.
- [24] "Freecad," <https://www.freecad.org/>, pristupljeno 13. travnja, 2023.
- [25] "Exis webcam," <https://www.trust.com/en/product/17003-exis-webcam-black-silver>, pristupljeno 13. travnja, 2023.
- [26] "Robot servo rds3135,3.5nm,metal,270°,coreless,8.4v (mrm-rds3135)," https://top.in.hr/HTML.ashx?_command=showItem&itemKey=mrm-rds3135, pristupljeno 13. travnja, 2023.
- [27] "Firmata," http://firmata.org/wiki/Main_Page, pristupljeno 13. travnja, 2023.
- [28] "pyfirmata," <https://pyfirmata.readthedocs.io/en/latest/>, pristupljeno 13. travnja, 2023.
- [29] T. Batpurev, "python_stereo_camera_calibrate," https://github.com/TemugeB/python_stereo_camera_calibrate, 2022.
- [30] "Opencv," <https://opencv.org/>, pristupljeno 13. travnja, 2023.
- [31] "Numpy," <https://numpy.org/>, pristupljeno 13. travnja, 2023.

Praćenje objekata aktivnim stereo vizualnim sustavom

Sažetak

Autor: Tin Mišić

Praćenje gibajućeg objekta nalazi mnoge primjene u područjima robotike, autonomnih vozila i nadzora. Od svih metoda najviše se ističu one koje koriste aktivne stereo sustave s pokretnim kamerama. Poseban naglasak u radu se stavlja na pristupe aktivnim stereo sustavima koji koriste virtualne horoptere i log-polarno mapiranje slike. Oblikovan je aktivni stereo sustav koji koristi obične web kamere i servo motore. Kamere i motori su povezani plastičnim dijelovima isprintanim na 3D printeru, a čitav sustav radi na osobnom računalu. Ispitan je rad oblikovanog sustava u stvarnom vremenu, utjecaj pozadine na praćenje gibajućeg objekta, gubljenje fokusa na objekt i točnost estimacije pozicije objekta. Uspoređeni su pristupi koji koriste kartezijsko mapiranje slika te pristupi koji koriste log-polarno mapiranje. Na osnovu dobivenih rezultata pokazane su neke prednosti log-polarnog mapiranja u odnosu na kartezijsko mapiranje te nedostaci konkretne implementacije.

Ključne riječi: praćenje objekta, aktivni stereo sustav, virtualni horopteri, log-polarno mapiranje, filter nultog dispariteta

Object tracking using an active stereo visual system

Abstract

Author: Tin Mišić

Object tracking has many applications in robotics, autonomous vehicles, and surveillance. Among all methods, those that use active stereo systems with moving cameras stand out the most. Special emphasis in this paper is given to approaches of active stereo systems that use virtual horopters and log-polar image mapping. An active stereo system is designed that uses ordinary web cameras and servo motors. The cameras and motors are connected with plastic parts printed on a 3D printer, and the entire system runs on a personal computer. The system's real-time performance, ability to track a moving object against various backgrounds, consistency of focus on the object, and accuracy of the estimated object position were tested. Approaches using Cartesian image mapping and those using log-polar mapping were compared. Based on the obtained results, some advantages of log-polar mapping over Cartesian mapping were shown, as well as the drawbacks of the specific implementation.

Keywords: object tracking, active stereo system, virtual horopters, log-polar mapping, zero disparity filter