

Određivanje udaljenosti detektiranog vozila ispred vozila s ugrađenim sustavom stereo kamera

Miljković, Marko

Master's thesis / Diplomski rad

2020

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:200:447573>

Rights / Prava: [In copyright](#)/[Zaštićeno autorskim pravom.](#)

Download date / Datum preuzimanja: **2025-01-09**

Repository / Repozitorij:

[Faculty of Electrical Engineering, Computer Science and Information Technology Osijek](#)



**SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I
INFORMACIJSKIH TEHNOLOGIJA**

Sveučilišni studij

**ODREĐIVANJE UDALJENOSTI DETEKTIRANOG
VOZILA ISPRED VOZILA S UGRADENIM
SUSTAVOM STEREO KAMERA**

Diplomski rad

Marko Miljković

Osijek, 2020.

SADRŽAJ

1. UVOD.....	1
2. PROBLEM ODREĐIVANJA UDALJENOSTI OBJEKATA OD KAMERE	3
2.1. Kamera kao senzor u vozilu	4
2.1.1. CMOS kamera	7
2.2. Određivanje udaljenosti objekata zasnovano na korištenju slike dobivene s jedne kamere	8
2.3. Određivanje udaljenosti objekata zasnovano na korištenju stereo para slika	11
2.3.1. Triangulacija.....	11
2.3.2. Ispravljeni stereo par slika	14
2.3.3. Epipolarna geometrija.....	15
2.3.4. Određivanje disparitetne mape	16
2.4. Postojeća rješenja za određivanje udaljenosti objekata od kamere.....	17
3. RAZVOJ RJEŠENJA ZA ODREĐIVANJE UDALJENOSTI VOZILA ZASNOVANOG NA KORIŠTENJU STEREO PARA SLIKA I NJEGOVA IMPLEMENTACIJA NA REALNOJ ADAS PLATFORMI.....	22
3.1. Pregled tehnologija upotrjebljenih za rješavanje problema određivanja udaljenosti detektiranog vozila zasnovanog na korištenju stereo para slika	22
3.1.1. Vision SDK programska platforma	22
3.1.2. ADAS APLHA razvojna ploča.....	24
3.2. Koncept rješenja problema određivanja udaljenosti detektiranog vozila na temelju stereo para slika.....	27
3.3. Programsko rješenje problema određivanja udaljenosti detektiranog vozila zasnovanog na korištenju stereo para slika.....	28
3.3.1. Kalibracija stereo sustava kamera.....	29
3.3.2. Algoritam za detekciju vozila na ispravljenom stereo paru slika	33
3.3.3. Algoritam za izračun mape dispariteta zasnovan na korištenju stereo para slika	34
3.3.4. Početna verzija konačnog rješenja	35
3.3.5. Implementacija FGS filtera.....	37
3.4. Način pokretanja programskog rješenja problema određivanja udaljenosti detektiranog vozila zasnovanog na korištenju stereo para slika	39

4. TESTIRANJE RADA PREDLOŽENOG RJEŠENJA ZA ODREĐIVANJE UDALJENOSTI VOZILA ZASNOVANOG NA KORIŠTENJU STEREO PARA SLIKA..	43
4.1. Baza signala upotrjebljenih za testiranje predloženog rješenja za određivanje udaljenosti vozila zasnovanog na korištenju stereo para slika	43
4.2. Rezultati testiranja predloženog rješenja za određivanje udaljenosti vozila zasnovanog na korištenju stereo para slika	44
4.2.1. Prvo testno vozilo	45
4.2.2. Drugo testno vozilo	46
4.2.3. Treće testno vozilo.....	50
4.2.4. Četvrto testno vozilo.....	52
4.2.5. Srednja relativna pogreška i standardna devijacija procjene udaljenosti za prva tri testna vozila	54
4.2.6. Vremena i frekvencije izvođenja predloženih rješenja	57
4.3. Osvrt na dobivene rezultate	58
5. ZAKLJUČAK.....	60
LITERATURA.....	62
SAŽETAK.....	64
ABSTRACT	65
ŽIVOTOPIS.....	66
PRILOZI.....	67

1. UVOD

Automobilska industrija jedna je od najrazvijenijih industrija na svijetu. Kako bi se takav status održao, zahtijeva se konstantna prilagodba potrebama tržišta, uz razvoj i upotrebu najnovijih tehnologija. Suvremeni život, posebice u razvijenijim zemljama, posjedovanje automobila nameće kao potrebu, a ne luksuz. Globalizacija podrazumijeva svakodnevno prelaženje velikih udaljenosti, kako robe, tako i ljudi. Iz navedenih se razloga broj automobila, kao i drugih vozila, u prometu svakodnevno povećava. Povećanjem broja vozila, te povećanjem njihove snage, povećava se i mogućnost događanja nesreće. Većina je prometnih nesreća uzrokovana ljudskim pogreškama. Te pogreške uključuju gubitak kontrole nad vozilom, kasnu reakciju, pogrešnu reakciju i sl. Kako bi se pogreške uzrokovane ljudskim faktorom svele na minimum, u velikom se broju suvremenih vozila upotrebljava sustav za pomoć vozaču pri vožnji (engl. *Advanced Driver Assistance System*, ADAS). To je elektronički sustav, koji pomoću različitih senzora percipira okolinu vozila, obrađuje informacije dobivene sa senzora te na osnovu rezultata njihove obrade donosi odluku koju radnju treba poduzeti u idućem trenutku, kako bi pomogao vozaču u vožnji [1].

Jedan od senzora koji se vrlo često upotrebljava u ADAS-u je standardna CCD (engl. *Charge Coupled Device*) ili CMOS (engl. *Complementary Metal Oxide Semiconductor*) kamera. Slika dobivena s kamere može se upotrebljavati u različite svrhe, poput: detekcije različitih objekata (pješačka, vozila, znakova, prometnih traka i sl.), raspoznavanja tih objekata, određivanja udaljenosti istih i sl. Udaljenost određenog objekta može se preciznije odrediti pomoću dviju kamera. Sustav sastavljen od dviju kamera koje su montirane na istoj konstrukciji, te snimaju istu scenu, sa svrhom određivanja udaljenosti objekata na sceni od kamere, naziva se sustav stereo kamera. Jedna od prednosti stereo sustava kamera, u odnosu na neke druge senzore za procjenu udaljenosti, poput lasera, jest cijena, koja je znatno manja.

Cilj je ovog diplomskog rada bio razviti rješenje za određivanje udaljenosti vozila detektiranog pomoću stereo sustava kamera, te ga implementirati na realnu ADAS razvojnu platformu. Kako bi se izvelo sve zadano, najprije je bilo potrebno odraditi kalibraciju stereo sustava kamera. Takvom se kalibracijom dobiva ispravljeni stereo par slika. Potom, na ispravljenom stereo paru slika primijenio se algoritam za detekciju vozila. Na kraju, udaljenost detektiranih vozila dobila se primjenom kalibracijskih parametara koristeći vrijednosti u mapi dispariteta (engl. *disparity map*), dobivenoj spajanjem odgovarajućih elemenata slike (engl. *pixel*) ispravljenog stereo para slika. Vrijednosti dobivene uporabom ovog rješenja uspoređene su s

vrijednostima dobivenim uporabom rješenja kod kojeg se udaljenost detektiranog vozila računa iz dostupne slike sa samo jedne kamere.

U nastavku je detaljnije opisano rješenje zadatka diplomskog rada. U drugom poglavlju dan je teorijski opis problema određivanja udaljenosti objekata od kamere. Naglasak je stavljen na metode koje upotrebljavaju stereo sustav kamera, ali je objašnjen i način određivanja udaljenosti koristeći samo jednu kameru. Također, dan je sažeti pregled nekoliko znanstvenih radova koji se bave predmetnom tematikom. Treće poglavlje u potpunosti opisuje razvoj i implementaciju vlastitog rješenja zadatka diplomskog rada. Najprije su opisane upotrjebljene tehnologije, odnosno programski alati, programski jezici te sklopovska platforma. Potom je predstavljeno konceptualno rješenje zadatka, kao i programsko rješenje njegova te implementacija na realnoj ADAS platformi. Na kraju trećeg poglavlja prezentiran je način pokretanja programskog rješenja na korištenoj sklopovskoj ADAS platformi. Četvrto poglavlje predstavlja rezultate testiranja rada predloženog rješenja zadatka. Opisan je testni skup podataka, način provedbe testiranja, prezentirani su dobiveni rezultati, uspoređeni s rezultatima rješenja za određivanje udaljenosti detektiranog vozila iz dostupne slike sa samo jedne kamere, te je dana diskusija o rezultatima. Zadnje poglavlje predstavlja zaključak diplomskog rada.

2. PROBLEM ODREĐIVANJA UDALJENOSTI OBJEKATA OD KAMERE

S ciljem jasnije klasifikacije autonomne vožnje, te njene standardizacije, Međunarodno je udruženje inženjera u automobilskoj industriji (engl. *Society of Automotive Engineers*, SAE), u standardu J3016_201401, definiralo 6 razina autonomije vožnje (Slika 2.1. [2]).

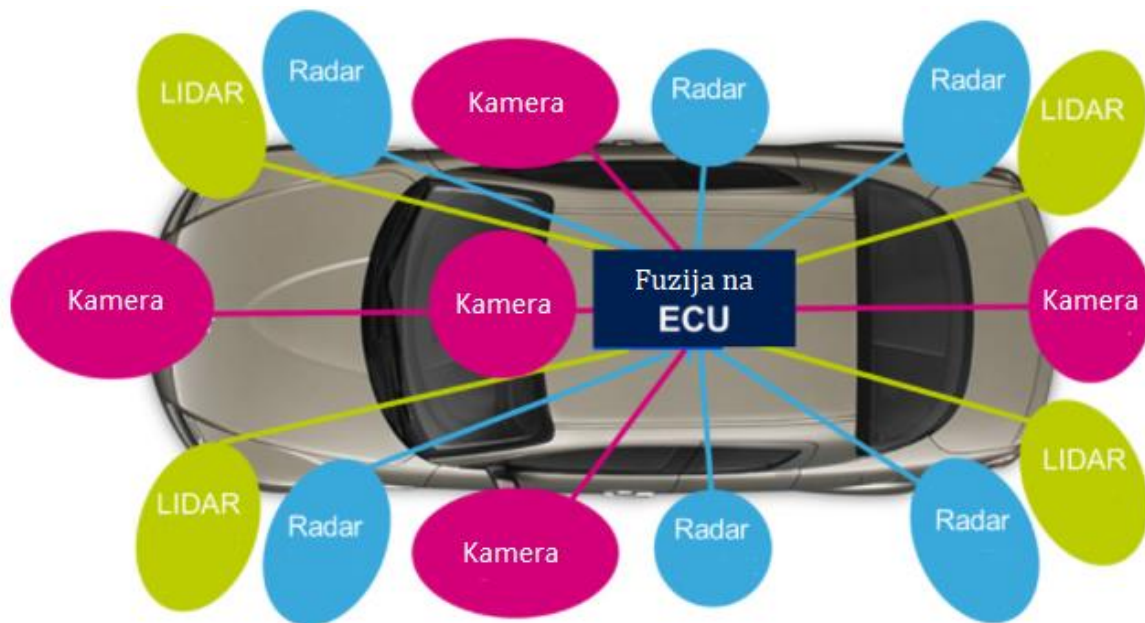
SAE razina	Naziv razine	Definicija	Skretanje i ubrzanje/ usporavanje	Praćenje voznog okruženja	Preuzimanje kontrole u slučaju potrebe	Modovi vožnje
Ljudski vozač prati vozno kruženje						
0	Nema automatizacije	Vozač u potpunosti kontrolira sve aspekte zadatka vožnje, čak i ako je vožnja unaprijedena sustavima upozorenja.	Vozač	Vozač	Vozač	-
1	Pomoć vozaču	Izvršavanje pojedinih sustava za pomoć vozaču: okretanje upravljača ili upravljanje s ubrzavanjem i usporavanjem u ovisnosti o okruženju, dok vozač izvodi sve ostale zadatke vožnje.	Vozač i sustav	Vozač	Vozač	Neki modovi vožnje
2	Djelomična automatizacija	Izvršavanje više sustava za pomoć vozaču u isto vrijeme: okretanje upravljača i upravljanje s ubrzavanjem i usporavanjem u ovisnosti o okruženju, dok vozač izvodi sve ostale zadatke vožnje.	Sustav	Vozač	Vozač	Neki modovi vožnje
Automatizirani sustav vožnje prati vozno okruženje						
3	Uvjetna automatizacija	Automatizacija zadatka vožnje u određenoj situaciji kontrolirajući sve aspekte vožnje, pri čemu vozač mora moći preuzeti upravljanje u slučaju potrebe.	Sustav	Sustav	Vozač	Neki modovi vožnje
4	Visoka automatizacija	Automatizacija zadatka vožnje u određenoj situaciji kontrolirajući sve aspekte vožnje i nastavak kontrole čak i ako vozač nije u stanju preuzeti upravljanje.	Sustav	Sustav	Sustav	Neki modovi vožnje
5	Potpuna automatizacija	Potpuna kontrola svih zadataka vožnje u svim situacijama koje bi inače mogao kontrolirati i vozač.	Sustav	Sustav	Sustav	Svi modovi vožnje

Sl. 2.1. Opis i karakteristike pojedine razine autonomije vožnje [2].

Kod razine 0 (nema automatizacije) vozač ima potpunu kontrolu nad vozilom, bez ikakve pomoći od strane ADAS-a. Razina 1 (pomoć vozaču) definirana je na način da ADAS pomaže u operacijama upravljanja vozilom, primjerice, sustav za pomoć pri kočenju (engl. *Anti-lock Braking System*, ABS). Razina 2 (djelomična automatizacija) nalaže da je barem jedan sustav u potpunosti automatiziran, poput upravljanja brzinom vozila i razmakom između vozila (engl. *Adaptive Cruise Control*), ali vozač cijelo vrijeme mora nadzirati sustav i okolinu. Ako se radi o razini 3 (uvjetna automatizacija), tada kritične funkcije, gledano s razine sigurnosti, u određenim uvjetima vožnje

obavlja sustav, ali vozač nadzire sustav, te po potrebi intervenira. Kod razine 4 (visoka automatizacija) vozila su dizajnirana za upravljanje kritičnim situacijama, te nadziranje stanja na cesti, tijekom cijelog puta, no, funkcije koje može izvesti sustav ne pokrivaju sve moguće situacije u prometu. Nije potreban nadzor vozača. Konačno, Razina 5 (potpuna automatizacija) nalaže da je vožnja potpuno automatizirana u svim mogućim situacijama [3]. Najviša certificirana razina autonomije komercijalnog vozila, koja je dostupna u trenutku pisanja ovog teksta, jest razina 3 [4].

Postoje tri glavna tipa senzora u vozilima: kamera, radar i lidar. Iako se za funkcije parkiranja vozila naširoko koriste ultrazvučni senzori, oni su od manjeg, ali ne i nebitnog, značaja za samu autonomnu vožnju. Vozila razine autonomije od 3 pa naviše, za obavljanje svojih funkcionalnosti zahtijevaju, odnosno zahtijevat će, fuziju svih triju navedenih tipova senzora, kojom upravlja fuzijska elektronska kontrolna jedinica (engl. *Electronic Control Unit*, ECU) (Slika 2.2.) [3].



Sl. 2.2. Primjer korištenja više tipova senzora u vozilu [3].

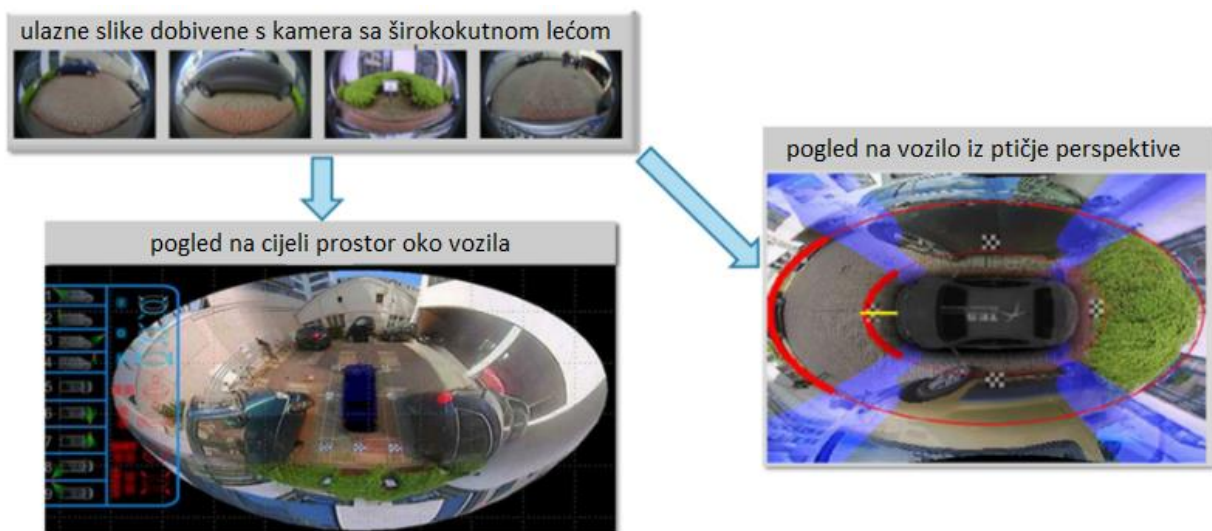
U nastavku rada fokus je stavljen samo na kamere jer su se one upotrebljavale za rješavanje zadatka ovog diplomskog rada.

2.1. Kamera kao senzor u vozilu

Kako je rečeno u uvodu, kamere su jedan od najčešće korištenih senzora u vozilima, poglavito zbog svoje niske cijene, ali i zbog razvoja sve naprednijih algoritama za obradu slike. One se mogu upotrebljavati za nadzor unutrašnjosti vozila, gdje služe za detekciju umora vozača, detekciju smjera gledanja (tj. pažnje) vozača, identifikaciju vozača, i sl. Također, kamere se mogu

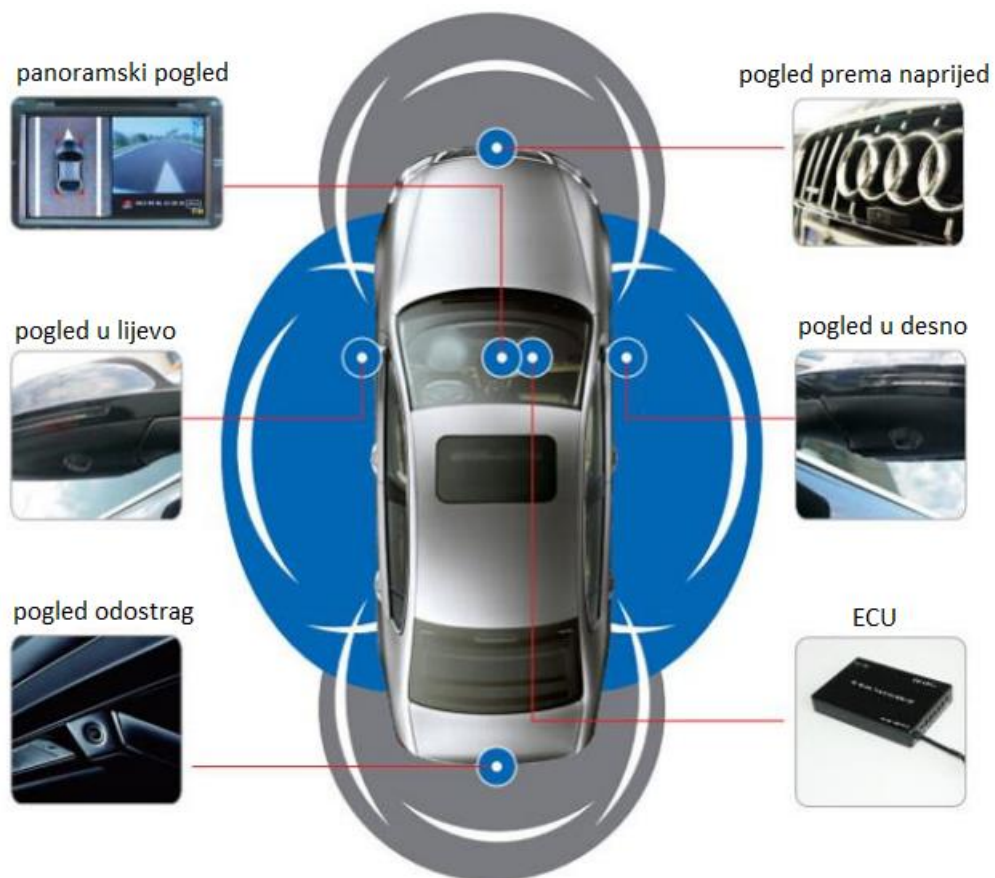
rabiti i za nadzor okoline vozila, a u tom se slučaju video dobiven s kamere koristi za obavljanje raznih funkcija, kao što su npr. detekcija i prepoznavanje prometnih znakova, detekcija i praćenje pješaka, biciklista, te drugih vozila, pružanje panoramskog pogleda, detekcija prometne trake, detekcija ruba ceste itd. [5].

Video dobiven s kamere, za razliku od podataka dobivenih s drugih senzora, nudi najviše detalja samom vozaču, ali je također vrlo koristan kao ulazni parametar za razne algoritme ADAS-a. Kamera montirana sa stražnje strane vozila, koja se koristi za dobivanje pogleda odostrag (engl. *rear view*) te kamere koje služe za dobivanje pogleda na cijeli prostor oko vozila (engl. *surround view*), omogućuju vozaču bolji uvid u okolinu vozila. Širokokutne leće (engl. *fisheye lens*) najčešće se upotrebljavaju, kako kod zadnjih kamera, tako i kod kamera koje se koriste za *surround view*. Ove kamere, iako nemaju mogućnost jasnog prikaza udaljenih objekata, omogućavaju mnogo veći horizontalni kut od kamera s normalnim lećama (engl. *normal lens*). Također, i kamera montirana sa stražnje strane vozila, i kamere koje se koriste za *surround view*, usmjerene su prema dolje. Kamera montirana sa zadnje strane vozila služi za prikaz područja iza vozila, kako bi pomogla prilikom parkiranja vozila, te kako bi se efikasnije uočile prepreke, ili ljudi, s ciljem izbjegavanja potencijalne nesreće. *Surround view* predstavlja sustav od, najčešće, četiri, ili šest kamera, postavljenih na način da daju uvid u cijelu okolinu vozila, odnosno, kako bi se ulazi dobiveni s kamera transformirali u pogled na vozilo iz ptičje perspektive (engl. *top view*), što je vidljivo na slici 2.3. [3].



Sl. 2.3. Primjer transformacije pogleda s četiriju različitih kamera montiranih na 4 strane automobila u jedinstven pogled na automobil iz ptičje perspektive [3].

Sustavi kamera okrenuti u smjeru vožnje unaprijed (engl. *forward-facing camera systems*) najčešće se nalaze u unutrašnjosti vozila, na prednjem dijelu, u visini vjetrobranskog stakla. Ove kamere služe za detekciju objekata koji se nalaze na srednjim i dalekim udaljenostima (100 – 250 metara). Video dobivenih s ovih kamera upotrebljava se za detekciju objekata, njihovu klasifikaciju, te određivanje njihove udaljenosti od kamera. Primjerice, pomoću algoritama za obradu ovih videa mogu se prepoznati pješaci i biciklisti, motorna vozila, prometne trake, rubovi ceste i sl. Kamere koje služe za detekciju objekata na srednjim udaljenostima od vozila upozoravaju vozača na raskrižja, pješake, naglo kočenje vozila ispred, te algoritmi za obradu videa dobivenih s ovih kamera omogućuju prepoznavanje prometnih traka i signalnih svjetala. Kamere koje služe za detekciju objekata na velikim udaljenostima upotrebljavaju se za prepoznavanje prometnih znakova, računanje udaljenosti vozila, ili pješaka, te navođenje po cesti. Glavna razlika između kamera koje služe za detekciju objekata na srednjim udaljenostima i kamera koje služe za detekciju objekata na velikim udaljenostima jest njihovo horizontalno vidno polje (engl. *Field of View*, FoV), gdje se za detekciju objekata na srednjim udaljenostima ono kreće u rasponu vrijednosti od 70 stupnjeva do 120 stupnjeva, a za detekciju objekata na većim udaljenostima



Sl. 2.4. Moguće lokacije montaže nekih od kamera u automobilu [5].

iznosi otprilike 35 stupnjeva [3]. Slika 2.4. prikazuje smještaj nekih od kamera koje se nalaze u automobilu [5].

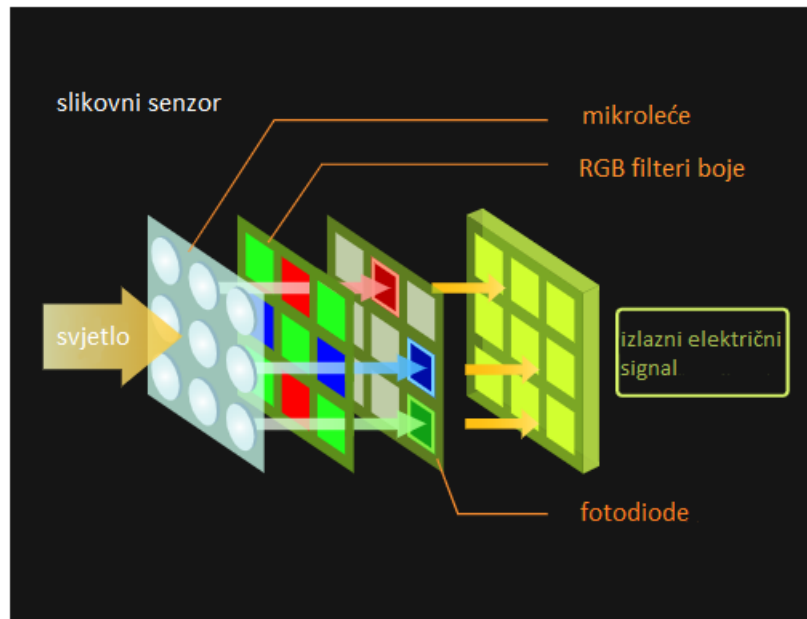
2.1.1. CMOS kamera

Princip rada CMOS senzora smišljen je još u prvoj polovici 60-ih godina 20. stoljeća, no nije komercijaliziran dok tehnologija mikrovlakana nije postala dovoljno napredna, što se dogodilo 90-ih godina 20. stoljeća. Slikovni senzori ugrađeni u današnje digitalne kamere, te mobilne telefone, uglavnom koriste, ili CCD, ili CMOS tehnologiju. I CCD i CMOS senzori poluvodički su uređaji koji služe kao „elektroničke oči“. Iako oba senzora koriste foto-diode, razlikuju se u procesu proizvodnje, te metodi čitanja izlaznog signala. CCD tehnologija bila je dominantna u početku zbog superiornije osjetljivosti i kvalitete slike, ali su razna poboljšanja CMOS tehnologije dovela do češće uporabe takve vrste senzora, od 2004. godine pa nadalje.

CCD senzor posjeduje polje kondenzatora, gdje svaki kondenzator skladišti električni naboj koji odgovara intenzitetu svjetline odgovarajućeg elementa slike. Kontrolni elektronički krug omogućuje svakom kondenzatoru slanje uskladištene količine naboja susjednom kondenzatoru, a zadnji kondenzator unutar polja šalje naboj pojačalu. Grupni stil prijenosa podataka karakterističan je za CCD senzor. Za razliku od CCD senzora, CMOS senzor posjeduje foto-diodu i CMOS tranzistorsku sklopku za svaki element slike, čime omogućava individualno pojačavanje signala određenog elementa slike. Uporabom matrice sklopki, signalu elementa slike može se pristupiti direktno i sekvencijalno, te mnogo većom brzinom nego što je slučaj kod CCD senzora. Uporaba pojačala za svaki element slike ima još jednu prednost, a to je smanjenje šuma prilikom čitanja električnog signala, dobivenog obradom registrirane svjetlosne zrake.

Proizvodnja CMOS slikovnih senzora jeftinija je od proizvodnje CCD senzora jer se postojeća oprema za proizvodnju poluvodiča može prenamijeniti za proizvodnju CMOS senzora. Za razliku od CCD senzora koji upotrebljavaju visokonaponske analogne elektroničke krugove, kod CMOS senzora koriste se manji digitalni krugovi, koji troše manje energije, te su u načelu otporni na *smear* (vertikalne bijele pruge na slici, koje se pojavljuju kao posljedica visokog intenziteta svjetline) i na *blooming* (oštećenje slike, poput bijelih točaka). Budući da se logički krug tijekom procesa proizvodnje može ugraditi na čip, CMOS senzori s elektroničkim krugom za obradu slike na čipu, razvijeni su za potrebe prepoznavanja objekata na slici i robotskog vida [6].

Princip rada CMOS slikovnog senzora prikazan je na slici 2.5. [6].



Sl. 2.5. Princip rada CMOS slikovnog senzora [6].

2.2. Određivanje udaljenosti objekata zasnovano na korištenju slike dobivene s jedne kamere

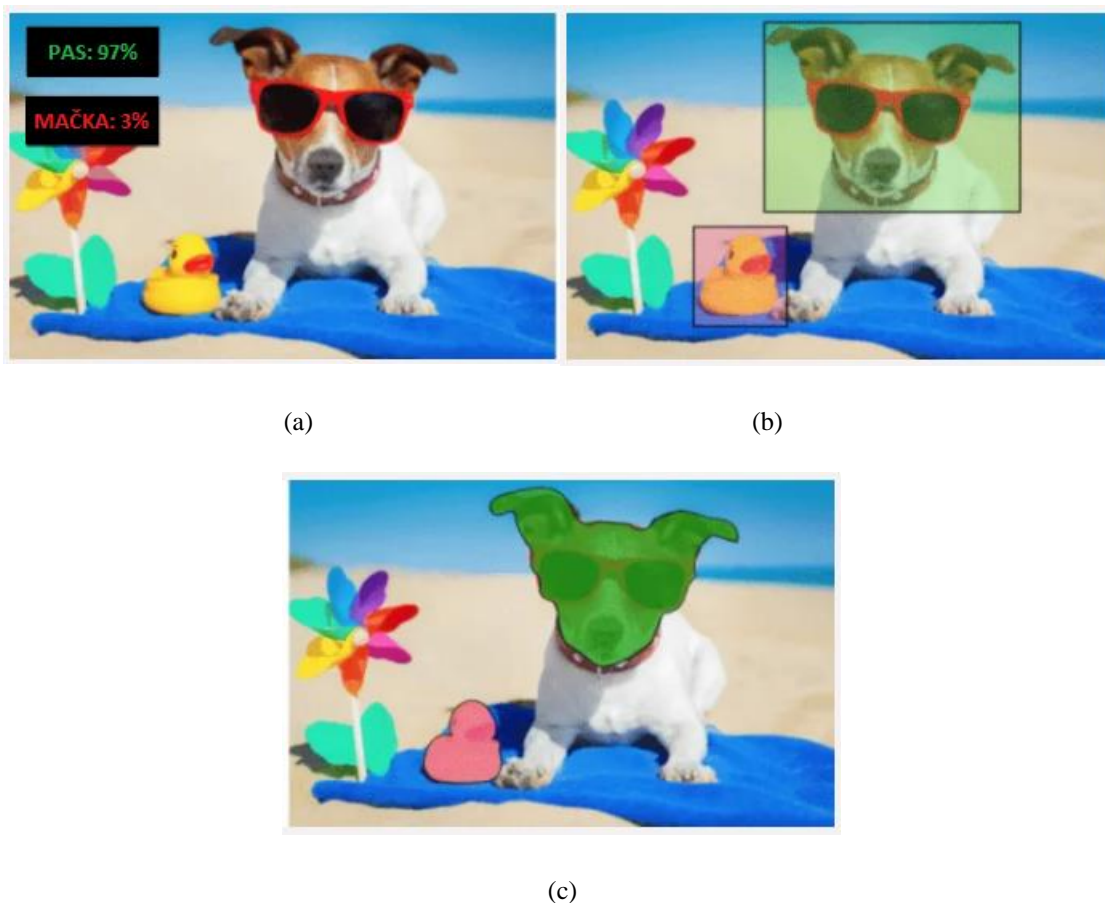
Kako je rečeno u uvodu, udaljenost objekata od CMOS kamere može se odrediti pomoću slika dobivenih s jedne kamere, ili pomoću stereo para slika. Za određenu primjenu, određivanje udaljenosti objekata zasnovano na korištenju slike dobivene s jedne kamere vrlo je privlačno, kako zbog niže cijene, tako i zbog manje zahtjevnog algoritma. Glavna mana ove metode lošija je procjena vrijednosti udaljenosti. Ako je preciznost rezultata metode bitna stavka, tada se, uz nešto višu cijenu i računalnu složenost, upotrebljava stereo sustav kamera [7]. U nastavku je objašnjena metoda izračuna udaljenosti objekata pomoću jedne kamere.

Postoji više metoda za izračun udaljenosti objekata zasnovanih na korištenju slike dobivene s jedne kamere. Osvrt na neke postojeće metode dan je u podpoglavlju 2.3. Metoda koja se koristila prilikom izrade rješenja zadatka ovog diplomskog rada objašnjena je u nastavku. Točnost dobivenih rezultata upotrijebljene metode, kako će se kasnije vidjeti, uvelike ovisi o preciznosti algoritma za prepoznavanje objekata. Prepoznavanje objekata na slici (engl. *object recognition*) općeniti je pojam koji opisuje nekoliko srodnih zadataka robotskog vida. Ti se zadaci mogu podijeliti u tri glavne kategorije:

- klasifikacija slike (engl. *image classification*);

- lokalizacija objekta (engl. *object localization*);
- detekcija objekta (engl. *object detection*).

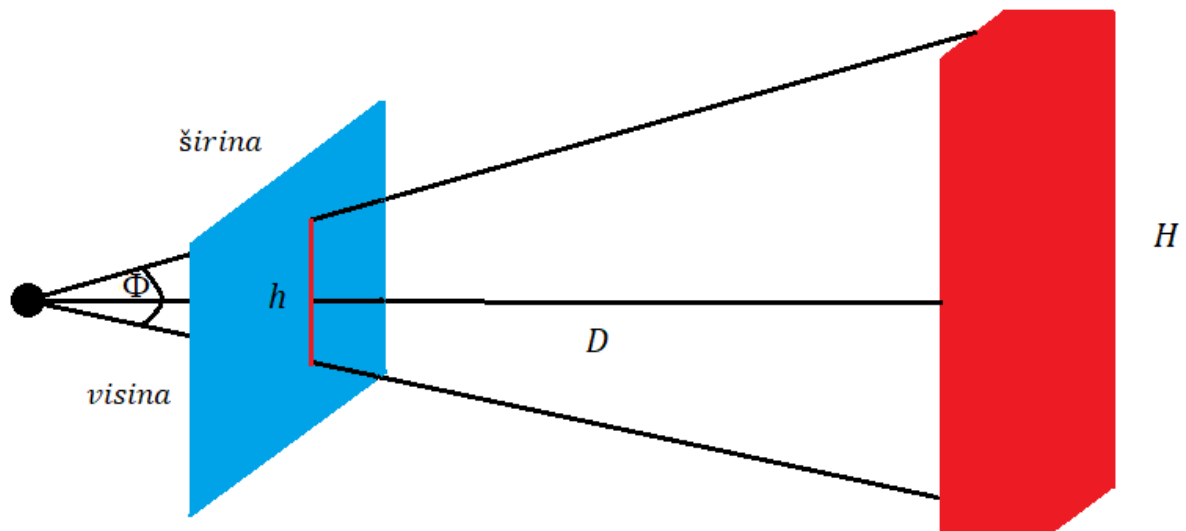
Klasifikacija slike predviđanje je klase jednog objekta na slici, s određenim stupnjem vjerojatnosti. Lokalizacija objekta označava prepoznavanje prisutnosti jednog, ili više objekata na slici, te određivanje njihove lokacije iscrtavanjem graničnog okvira (engl. *bounding box*). Detekcija objekta kombinira zadatke klasifikacije slike i lokalizacije objekta, tj. iscrtava granični okvir oko svakog prepoznatog objekta, te svakom objektu dodjeljuje klasu. Postoji još jedna nadopuna navedenih glavnih kategorija zadataka prepoznavanja objekata na slici, a to je segmentacija objekta (engl. *object segmentation*), koja naglašava svaki element slike prepoznatog objekta, umjesto šturog iscrtavanja graničnog okvira [8]. Slika 2.6. prikazuje razliku između klasifikacije slike, detekcije objekta, te segmentacije objekta [9].



Sl. 2.6. Primjeri: (a) klasifikacija slike, (b) detekcija objekta (c) segmentacija objekta [9].

Algoritam za izračun udaljenosti detektiranog vozila zasnovan na korištenju slike s jedne kamere, koji je upotrijebljen prilikom izrade rješenja ovog diplomskog rada, zasniva se na

definiranju prosječne visine objekta od interesa u stvarnom svijetu. Postavke na kojima se model algoritma zasniva prikazane su na slici 2.7.



Sl. 2.7. Postavke na kojima se zasniva model algoritma za određivanje udaljenosti detektiranog objekta korištenjem slike dobivene s jedne kamere.

Oznake na slici 2.7. predstavljaju sljedeće:

- H – prosječna visina objekta u stvarnom svijetu [m];
- D – udaljenost objekta od optičkog centra kamere [m];
- h - visina graničnog okvira objekta na slici [element slike];
- Φ – kut između optičke zrake koja spaja vrh objekta s optičkim centrom kamere i optičke zrake koja spaja dno objekta s optičkim centrom kamere [rad];
- *širina* - širina slike [element slike];
- *visina* – visina slike [element slike].

Pretpostavlja se da je kamera učvršćena te da je postavljena na visinu koja približno odgovara polovici visine H . Prema slici 2.7. može se izvesti izraz (2.1.) za izračun udaljenosti D .

$$D = \frac{H}{2 \operatorname{tg}\left(\frac{\Phi}{2}\right)} \quad (2.1.)$$

Kut Φ , potreban za izračun udaljenosti D , može se izračunati prema izrazu (2.2.),

$$\Phi = h \cdot \frac{\alpha}{height} \quad (2.2.)$$

gdje α predstavlja vertikalno vidno polje kamere (engl. *field of view*, FOV).

2.3. Određivanje udaljenosti objekata zasnovano na korištenju stereo para slika

Standardne CMOS kamere stvaraju slike projiciranjem svjetlosne zrake trodimenzionalne scene na dvodimenzionalnu plohu. Time se gubi jedna dimenzija. Ova dimenzija predstavlja dubinu slike, odnosno udaljenost dijelova scene od kamere. Dubina se može izračunati pomoću sustava sastavljenog od dviju kamera, koje su montirane na istoj konstrukciji, te snimaju istu scenu. Ovaj se sustav naziva sustav stereo kamera. Izračunavanje dubine slike dopušta kreiranje trodimenzionalnog modela scene, tzv. rekonstrukciju 3D scene. Algoritam za rekonstrukciju 3D scene mora najprije spojiti točke jedne slike s odgovarajućim točkama druge slike, a potom izvršiti triangulaciju za sve parove točaka, koristeći intrinzične i ekstrinzične parametre kamere, kako bi izračunao njihove koordinate u trodimenzionalnom prostoru.

2.3.1. Triangulacija

Kako bi se dobio bolji uvid u pojam triangulacije, najprije je potrebno objasniti proces kalibracije kamere. Kalibracija kamere proces je dobivanja intrinzičnih i ekstrinzičnih parametara kamere, kao i parametara izobličenja slike. Za proces rekonstrukcije trodimenzionalne scene bitnije je dobiti uvid u značenje intrinzičnih i ekstrinzičnih parametara. Intrinzični parametri obuhvaćaju žarišnu duljinu kamere f , te koordinate središta slike (u_c, v_c) . Parametri koji opisuju poziciju i orijentaciju kamere u odnosu na neki referentni koordinatni sustav nazivaju se ekstrinzični parametri kamere. U pogledu stereo vizije, ekstrinzični parametri predstavljaju položaj jedne kamere u odnosu na drugu.

Matrica koja sadrži intrinzične parametre kamere naziva se projekcijskom matricom i definirana je izrazom (2.3.).

$$\mathbf{P} = \begin{bmatrix} f & 0 & u_c \\ 0 & f & v_c \\ 0 & 0 & 1 \end{bmatrix} \quad (2.3.)$$

Projekcijska matrica $\tilde{\mathbf{P}}_R$ koja definira projekciju točke u prostoru prikazane u koordinatnom sustavu lijeve kamere, na sliku desne kamere, definirana je jednadžbom (2.4.).

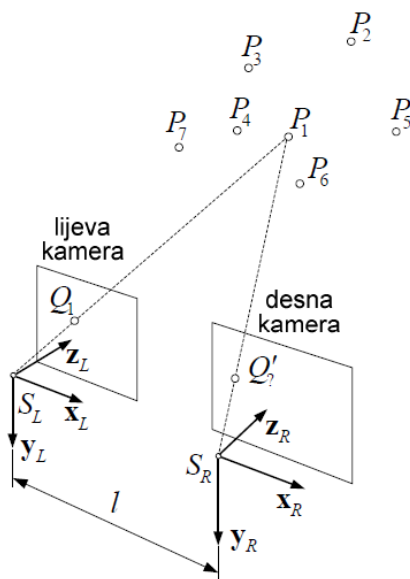
$$\tilde{\mathbf{P}}_R = [\mathbf{P}_R \cdot \mathbf{R}({}^R\boldsymbol{\phi}_L) \quad ; \quad \mathbf{P}_R \cdot {}^R\mathbf{t}_L] \quad (2.4.)$$

U izrazu (2.4.) matrica \mathbf{P}_R predstavlja projekcijsku matricu desne kamere, dok matrica $\mathbf{R}({}^R\boldsymbol{\phi}_L)$ označava rotacijsku matricu lijeve u odnosu na desnu kameru. Vektor ${}^R\mathbf{t}_L$ translacijski je vektor lijeve u odnosu na desnu kameru. Rotacijska matrica opisuje rotaciju, dok translacijski vektor opisuje translaciju lijeve kamere u odnosu na desnu, u trodimenzionalnom koordinatnom sustavu. Rotacijska matrica i translacijski vektor daju potpunu informaciju o položaju lijeve kamere u odnosu na desnu.

Sada je moguće definirati pojam triangulacije. Ukoliko se pretpostavi:

- da su kamere kalibrirane, tj. da su projekcijske matrice \mathbf{P}_L (projekcijska matrica lijeve kamere) i \mathbf{P}_R poznate;
- da je položaj jedne kamere u odnosu na drugu poznat, a time i matrica $\tilde{\mathbf{P}}_R$;
- da je poznato koja točka jedne slike odgovara kojoj točki druge slike.

tada je moguće izračunati poziciju točke u prostoru, s obzirom na koordinatni sustav svake od kamera. Taj se postupak naziva triangulacija. Triangulaciju je moguće vizualizirati pomoću slike 2.8. [10].



Sl. 2.8. Postavke na kojima se zasniva model triangulacije. [10].

Oznake na slici 2.8. predstavljaju sljedeće:

- $P_i, i = 1, 2, \dots, 7$ – koordinate točke u trodimenzionalnom prostoru [m];
- Q_1 – koordinate projekcije trodimenzionalne točke scene P_1 na sliku lijeve kamere [element slike];
- Q'_2 – koordinate projekcije trodimenzionalne točke scene P_1 na sliku desne kamere [element slike];
- S_L – ishodište koordinatnog sustava lijeve kamere (optički centar lijeve kamere) [m];
- S_R – ishodište koordinatnog sustava desne kamere [m];
- $\mathbf{x}_L, \mathbf{y}_L, \mathbf{z}_L$ – koordinatne osi koordinatnog sustava lijeve kamere [m];
- $\mathbf{x}_R, \mathbf{y}_R, \mathbf{z}_R$ – koordinatne osi koordinatnog sustava desne kamere [m];
- l – horizontalna udaljenost između optičkih centara lijeve i desne kamere [m].

Jednadžbe (2.5.) i (2.6.) definiraju optičke zrake lijeve, odnosno desne slike. Optička zraka, za danu točku prostora, predstavlja pravac koji spaja trodimenzionalnu točku scene s optičkim centrom kamere.

$$s_i \begin{bmatrix} {}^L\mathbf{m}_i \\ \dots \\ 1 \end{bmatrix} = \mathbf{P}_L \cdot {}^L\mathbf{p}_i \quad (2.5.)$$

$$s'_j \begin{bmatrix} {}^R\mathbf{m}'_j \\ \dots \\ 1 \end{bmatrix} = \tilde{\mathbf{P}}_R \begin{bmatrix} {}^L\mathbf{p}_i \\ \dots \\ 1 \end{bmatrix} \quad (2.6.)$$

Vektor ${}^L\mathbf{m}_i$ predstavlja koordinate projekcije promatrane točke u prostoru na koordinatni sustav lijeve slike, dok vektor ${}^R\mathbf{m}'_j$ predstavlja koordinate projekcije iste te točke na koordinatni sustav desne slike. Vektor ${}^L\mathbf{p}_i$ označava poziciju točke s obzirom na koordinatni sustav lijeve kamere, te je dobivanje njegove vrijednosti krajnji cilj triangulacije. Parametri s_i i s'_j predstavljaju pozicije trodimenzionalne točke na optičkoj zraci lijeve, odnosno desne slike.

Optičke zrake definirane jednadžbama (2.5.) i (2.6.) u idealnom se slučaju sijeku u promatranoj točki u prostoru, te se rješavanjem sustava jednadžbi (2.5.) i (2.6.) može izračunati vektor ${}^L\mathbf{p}_i$. U realnom se slučaju, zbog pogreške određivanja pozicija ${}^L\mathbf{m}_i$ i ${}^R\mathbf{m}'_j$, optičke zrake mogu mimoilaziti. U tom se slučaju triangulacija može provesti nekom od metoda optimizacije.

2.3.2. Ispravljeni stereo par slika

Ukoliko se koristi sustav od dvije kamere montirane na istu konstrukciju, tako da je njihov međusobni položaj nepromjenjiv, tada se odgovarajućim postupkom kalibracije kamera može doći do parametara koji omogućavaju transformaciju slika snimljenih kamerama, u slike koje bi se dobile sustavom kamera, za koji vrijedi:

- $\mathbf{P}_L = \mathbf{P}_R$, tj. intrinzični parametri kamera identični su,
- $\mathbf{R}({}^R\boldsymbol{\phi}_L) = \mathbf{I}^{3 \times 3}$, gdje $\mathbf{I}^{3 \times 3}$ predstavlja jediničnu matricu dimenzija 3×3 , tj. osi koordinatnih sustava lijeve i desne kamere, S_L i S_R paralelne su,
- ${}^R\mathbf{t}_L = [-l, 0, 0]^T$, gdje je l udaljenost između optičkih centara kamera.

Ta se transformacija naziva ispravljanje (engl. *rectification*), a za dobiveni se par slika koristi izraz ispravljeni stereo par slika.

Razlika između pozicije točke na lijevoj slici i pozicije točke na desnoj slici, koja predstavlja istu 3D točku, označena u jednadžbi (2.7.) s d_i , naziva se disparitet.

$${}^L u_i - {}^R u_i = d_i = f \frac{l}{{}^L z_i} \quad (2.7.)$$

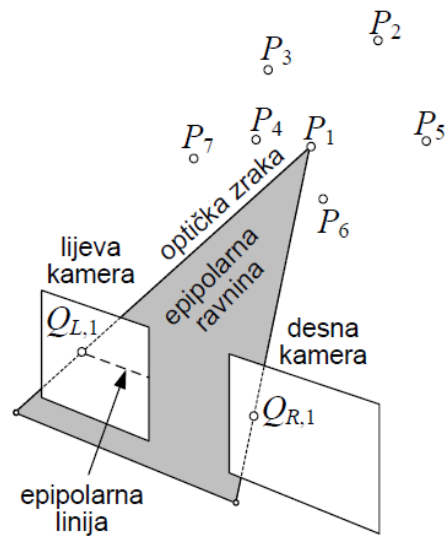
Oznaka ${}^L u_i$ predstavlja u koordinatu projekcije 3D točke na koordinatni sustav lijeve slike, a ${}^R u_i$ predstavlja u koordinatu projekcije te iste točke na koordinatni sustav desne slike. Udaljenost 3D točke u smjeru z koordinatne osi koordinatnog sustava lijeve kamere označena je s ${}^L z_i$. Disparitet je obrnuto proporcionalan s udaljenošću točke od ravnine slike. Izraz (2.8.) prikazuje jednadžbu za izračunavanje udaljenosti točke od koordinatnog sustava lijeve kamere u 3D prostoru, na temelju dispariteta.

$${}^L \mathbf{p}_i = \frac{l}{d_i} \begin{bmatrix} {}^L \mathbf{m}_i - \mathbf{m}_c \\ \dots \\ f \end{bmatrix} \quad (2.8.)$$

Vektor \mathbf{m}_c predstavlja koordinate središta lijeve slike. Tri točkice u jednadžbi (2.8.), zajedno s izrazom ${}^L \mathbf{m}_i - \mathbf{m}_c$, predstavljaju skraćeni zapis skalarnih koeficijenata vektora ${}^L \mathbf{m}_i - \mathbf{m}_c$.

2.3.3. Epipolarna geometrija

Projekcija optičkog centra jedne kamere na ravninu slike druge kamere, naziva se epipol.



Sl. 2.9. Postavke na kojima se zasniva model epipolarne geometrije [10].

Ravnina koja prolazi promatranom točkom u prostoru i centrima projekcija objiju kamere, naziva se epipolarna ravnina. Linije koje predstavljaju presjeka epipolarne ravnine s ravninama slika, nazivaju se epipolarne linije. Epipolarne linije također se mogu definirati kao linije na slici koje prolaze epipolom i projekcijom točke na sliku. Ova je geometrija prikazana na slici 2.9. [10].

Za svaku točku prve slike može se odrediti njoj odgovarajuća epipolarna linija na drugoj slici. Ta se epipolarna linija može promatrati kao projekcija optičke zrake prve kamere, koja prolazi optičkim centrom prve kamere i promatranom točkom, na sliku druge kamere. Budući da se 3D točka može nalaziti samo na toj optičkoj zruci, to znači da se njezina projekcija na drugu sliku može nalaziti samo na razmatranoj epipolarnoj liniji. Ta se činjenica naziva epipolarno ograničenje i igra značajnu ulogu u računalnom vidu.

Naime, kada se za neku točku prve slike traži točka na drugoj slici, koja odgovara istoj 3D točki scene, tada je pretraživanje druge slike ograničeno samo na epipolarnu liniju koja odgovara točki prve slike. To ograničenje smanjuje vrijeme pretraživanja i ujedno smanjuje vjerojatnost pogrešnog sparivanja točaka jer se značajno ograničava broj mogućih „kandidata“ na drugoj slici, za sparivanje s nekom točkom prve slike.

U slučaju ispravljenog stereo para slika, epipolarna linija na drugoj slici, koja odgovara nekoj točki prve slike, zapravo predstavlja isti onaj redak druge slike, u kojem se na prvoj slici nalazi željena točka [10].

2.3.4. Određivanje disparitetne mape

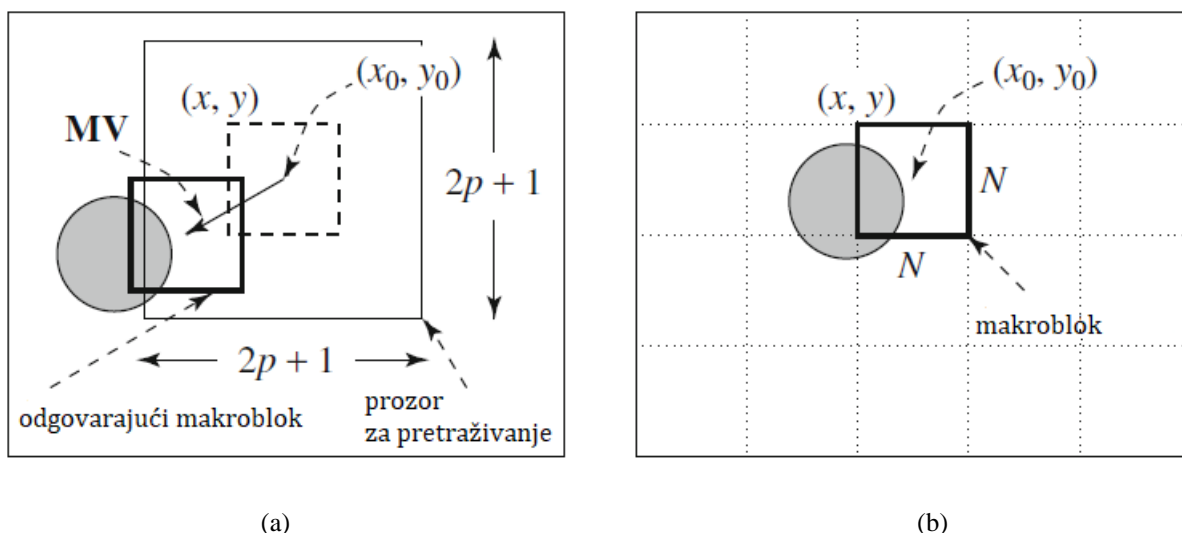
Glavni problem u stereo viziji predstavlja upravo određivanje mape dispariteta, odnosno pridruživanje točaka slike jedne kamere, odgovarajućim točkama na slici druge kamere. Problem se javlja kod slika koje imaju velike površine bez teksture, kao što su, primjerice, zidovi u prostori. Zahtjevi tehnologije takvi su da podatke o dubini slike treba dostaviti u što kraćem vremenu, odnosno u stvarnom vremenu (engl. *real time*), kako bi ih se moglo što efikasnije iskoristiti.

Jedan od algoritama za spajanje točaka, koji je korišten i prilikom rješavanja zadatka ovog diplomskog rada, jest algoritam spajanja blokova (engl. *block-matching algorithm*). Generalno, algoritam spaja makroblokove u slijedu okvira videa, u svrhu određivanja pomaka [11]. Taj se algoritam može primijeniti i kod stereo para slika, u svrhu računanja mape dispariteta.

Algoritam spajanja blokova radi na način da jednu sliku podijeli u makroblokove, te uspoređuje svaki makroblok s odgovarajućim makroblokom na drugoj slici, kao i njegovim susjedima. U slučaju stereo vizije samo se makroblokovi koji se nalaze na odgovarajućoj epipolarnoj liniji uzimaju u obzir, uz zadanu maksimalnu vrijednost pomaka. Stvara se vektor koji opisuje pomak makrobloka u odnosu na prvu sliku, a to je upravo disparitet. Za određivanje najboljeg odgovarajućeg makrobloka, minimizira se određena funkcija cilja, poput sume apsolutnih razlika vrijednosti intenziteta odgovarajućih elemenata slike unutar makrobloka (engl. *sum of absolute differences*) [12]. Slika 2.10. prikazuje općeniti princip rada algoritma spajanja blokova [11].

Oznake na slici 2.10. predstavljaju sljedeće:

- (x, y) – koordinate gornjeg lijevog vrha makrobloka na ciljnoj slici [element slike];
- (x_0, y_0) – koordinate središta makrobloka na ciljnoj slici [element slike];
- $2p + 1$ – duljina stranice prozora za pretraživanje, gdje je p pozitivan cijeli broj [element slike];
- N – duljina stranice makrobloka [element slike];
- \mathbf{MV} – vektor pomaka (engl. *Moving Vector*) odgovarajućeg makrobloka na referentnoj slici, u odnosu na poziciju makrobloka ciljne slike [element slike]



Sl. 2.10. Općeniti princip rada algoritma spajanja blokova: (a) referentna slika, (b) ciljna slika [11].

2.4. Postojeća rješenja za određivanje udaljenosti objekata od kamere

Osvrt na neke postojeće, najnovije (engl. *state-of-the-art*) metode izračuna udaljenosti objekata zasnovano na korištenju slika dobivenih s jedne kamere, te stereo para slika, dan je u nastavku.

U radu [13] predloženo je geometrijsko rješenje problema određivanja udaljenosti vozila, u stvarnom vremenu, pomoću slika dobivenih s jedne kamere. Predložena metoda upotrebljava geometrijske značajke prometnih traka, te je kompatibilna s algoritmima za detekciju prometnih traka i vozila, već integriranim u postojeće ADAS-e. Geometrija slike dobivene s jedne kamere, ovisi o odnosu detektiranih paralelnih linija i površina na slici. Rad počinje od pretpostavke da su ekstrinzični parametri kamere nepoznati, ali predlaže rješenje pronalaska visine kamere, koristeći horizont. Horizont predstavlja postepeno nestajuću liniju na cesti. On se pak određuje pomoću poznatih odnosa udaljenosti detektiranih paralelnih linija. Nakon toga, rekonstruira se površina ceste putem inverznog relacijskog mapiranja (engl. *Inverse Perspective Mapping*, IPM). IPM transformira sliku u sliku iz ptičje perspektive, kako bi se detektirane linije koje se sijeku, prikazale kao paralelne, kakve zaista i jesu. Određivanjem visine kamere i rekonstrukcijom površine ceste završava se inicijalizacija kamere, koja se odrađuje samo jednom. Kada je inicijalizacija kamere obavljena, udaljenost detektiranog vozila računa se reprojeksijom točke slike na pravac, koji siječe rekonstruiranu površinu ceste. Ovakvo rješenje uvodi dvije značajne novosti: upotrebljavaju se unakrsni omjeri širina prometnih traka kako bi se odredio horizont, te se inverzno perspektivno mapiranje, kao i visina kamere, određuje pomoću širine prometne trake, te poznatog horizonta. Za evaluaciju, vrijednosti dobivene putem lidara upotrjebljene su kao referentne (engl. *ground truth*),

te su uspoređene performanse algoritma s performansama lidara te s najnovijim algoritmima procjene udaljenosti pomoću slike dobivene s jedne kamere, koji se zasnivaju na dubokom učenju. Testiranje, izvršeno na tri javna skupa podataka (engl. *dataset*), a to su „Kitti“ [14], „nuScenes“ [15], te „Lyft level 5“ [16], pokazalo je da predloženo rješenje ne odstupa mnogo u vrijednostima korijena srednje kvadratne udaljenosti (engl. *Root Mean Square Error*, RMSE) između rezultata rješenja i referentnih vrijednosti, koje se kreću između 6,1 i 7,31. Metoda predložena u ovom radu daje točnije vrijednosti od ostalih algoritama na dvama od triju skupova podataka. Samo na „Kitti“ skupu podataka daje lošije rezultate, i to samo od jedne metode, koja se temelji na nadziranom dubokom učenju. Osim toga što daje točnije rezultate od navedenih algoritama, procesorski je najmanje zahtjevna te je pogodna za sve domene podatka. Mana predložene metode, uz manje točne rezultate dobivene na „Kitti“ skupu podataka, jest ta što je proces inicijalizacije kamere računski vrlo kompleksan. No, inicijalizacija se odrađuje samo jednom, te nakon toga procjena udaljenosti radi u stvarnom vremenu.

Postojeće metode za izračun udaljenosti vozila pomoću jedne kamere nailaze na problem velikih razlika u dobivenim rezultatima određivanja udaljenosti različitih tipova vozila. Jedan od glavnih razloga postojanja navedenog problema promjena je kuta vozila od interesa u odnosu na kameru (engl. *attitude angle*). U svrhu rješavanja ovog problema, autori rada [17] predlažu metodu koja se temelji na detekciji i segmentaciji ciljanog vozila. Model neuronske mreže za regresiju kuta upotrebljava se za dobivanje informacije o kutu pod kojim se ciljano vozilo nalazi, u odnosu na kameru. Za određivanje dimenzije ciljanog vozila, rabi se neuronska mreža za procjenu dimenzije. Potom, stvoren je vektorski geometrijski model neuronske mreže s 2D bazom, u sukladnosti s geometrijskim principom analize slike, kako bi se precizno rekonstruirao zadnji dio ciljanog vozila. Konačno, za izračun udaljenosti upotrebljava se modeliranje područja udaljenosti temeljeno na principu projekcije kamere. Testiranje rada algoritma odrađeno je na javnom „Kitti“ skupu podataka, a obavljeno je u tri faze. Prva faza obuhvaća testiranje točnosti sustava, gdje se uspoređuje prosječna pogreška procjene udaljenosti s različitim metodama za procjenu udaljenosti vozila zasnovano na korištenju slike dobivene s jedne kamere. U usporedbi s ostalim metodama, prosječna pogreška na gotovo svim testiranim udaljenostima manja je, te je točnost procjene udaljenosti veća. Drugu fazu testiranja predstavlja testiranje modula za izračun kuta pod kojim se ciljano vozilo nalazi, u odnosu na kameru. Rezultati procjene udaljenosti predložene metode uspoređuju se s rezultatima metode koja ne obuhvaća informaciju o navedenom kutu. Kao mjera uspješnosti upotrebljava se prosječna pogreška na različitim udaljenostima. Rezultati predloženog algoritma daju malo odstupanje u usporedbi s referentnim vrijednostima, te su točniji i imaju manje

standardno odstupanje pogreške gledajući sve testne udaljenosti, u usporedbi s algoritmom koji kut vozila, u odnosu na kut kamere, ne uzima u obzir. Treća faza testiranja je robusnosti predložene metode. Kako bi se testirala robusnost, upotrijebljene su slike vozila koja se nalaze pod različitim kutovima u odnosu na kameru. Pritom, prosječna stopa pogrešaka rezultata procjene udaljenosti različitih tipova vozila predloženog algoritma, uspoređena je s rezultatima još dviju metoda za procjenu udaljenosti vozila pomoću jedne kamere. U usporedbi s drugim metodama, predloženi algoritam daje manju vrijednost prosječne stope pogrešaka pri procjeni udaljenosti vozila koja se nalaze pored kamere, pri procjeni udaljenosti različitih tipova vozila, te je smanjeno odstupanje prosječne stope pogrešaka rezultata procjene udaljenosti vozila koja se nalaze ispred kamere, od rezultata procjene udaljenosti vozila koja se nalaze pored kamere. Dakle, sve faze testiranja pokazale su da metoda postiže superiorniji učinak u usporedbi s drugim postojećim metodama, za različite tipove vozila, kao i različite kutove pod kojima se vozila nalaze u odnosu na kameru. Moguća poboljšanja metode mogu se predstaviti u vidu točnije procjene udaljenosti vozila pri različitim uvjetima vožnje te točnije procjene udaljenosti vozila koja su djelomično zaklonjena.

Kako su algoritmi za procjenu udaljenosti pomoću stereo sustava kamera računski zahtjevniji od algoritama za procjenu udaljenosti pomoću jedne kamere, a jedan od glavnih zahtjeva ADAS-a jest rad u stvarnom vremenu, metoda [18] predlaže način smanjenja računске kompleksnosti takvih algoritama. Predložena metoda smanjuje računске zahtjeve cjenovno-obujamskog filtriranja (engl. *cost-volume filtering*, CVF) za spajanje odgovarajućih elemenata slike stereo para slika. Najprije se obavi CVF filtriranje nad svakim elementom slike te se iz toga dobije početna dubinska mapa. Nakon toga, pomoću histograma početnih vrijednosti dubinske mape, računaju se vrijednosti dubinske mape u smjeru osi okomite na površinu slike, tzv. glavne vrijednosti dubinske mape (engl. *principle depth values*). Potom, vrši se navođeno filtriranje pomoću dobivenih glavnih vrijednosti dubinske mape. Na kraju, obavlja se ispunjavanje pruga, te se primjenjuje medijalni filter s težinama (engl. *weighted median filter*), kako bi se ispunile rupe zaklonjenog prostora. Prilikom testiranja, kao mjere uspješnosti rada algoritma upotrijebljeni su vrijeme izračuna, te postotak neispravno spojenih elemenata slike (engl. *percentage of bad matching pixels*). Algoritam je testiran na javno dostupnom „Middlebury“ [19] skupu podataka, a rezultati su uspoređeni s rezultatima klasične CVF metode. Predložena metoda radi dvostruko brže, te ima za 0.1 posto manji postotak neispravno spojenih elemenata slike, u usporedbi s CVF algoritmom. Moguća poboljšanja metode mogu se predstaviti u vidu post-procesiranja, sa svrhom poboljšanja algoritma za ispunjenje rupa kod zaklonjenih područja slike.

Iako je u posljednje vrijeme napravljen velik napredak u preciznosti rezultata algoritama za izračun udaljenosti objekata zasnovanih na korištenju stereo para slika, postoji još mnogo neriješenih problema. Jedan je od problema izračun udaljenosti u zahtjevnim uvjetima, poput kiše, snijega, ili niskog osvjetljenja. Prijedlog rješenja navedenog problema iznesen je u radu [20]. Kako bi polučio glatke i točne rezultate u zahtjevnim uvjetima, predloženi algoritam od grube mape dispariteta pokušava dobiti glatku, odnosno precizniju. Najprije se izradi Gaussova piramida (engl. *Gaussian pyramid*) na temelju stereo para slika, te se generiraju tri sloja, a stopa smanjenja rezolucije (engl. *downsampling rate*) između svakog sloja postavi se na dva, tj. rezolucija novog sloja, u odnosu na prethodni, dvostruko je manja po širini i po visini. Na svakom se sloju određuju vrijednosti cijena-obujam pomoću metode ne-lokalne cijene spajanja (engl. *Non-Local matching Cost*, NLC), kako bi se smanjio utjecaj šuma u stereo paru slika. Potom, na temelju vrijednosti cijena-obujam, izračunavaju se sigurnosne vrijednosti (engl. *confidence score*) za svaki element slike, te se one mapiraju na vjerojatnost pojave stršćih vrijednosti (engl. *outliers*). Nakon dobivenih vjerojatnosti pojave stršćih vrijednosti, cijena spajanja njima se otežava. Za dobivanje vrijednosti dispariteta iz cijene spajanja, upotrebljava se modifikacija više-putnog Viterbi algoritma (engl. *Multi-Path Viterbi*, MPV), nakon kojeg se obavlja post-procesiranje putem otežanog medijalnog filtera (engl. *weighted median filter*). Na kraju, povećava se rezolucija (engl. *upsampling*) mape dispariteta grubog sloja, te se upotrebljava kao početna vrijednost glatkog sloja. Na temelju te početne vrijednosti, ograničava se prostor za pretragu glatkog sloja, kako bi se ubrzao algoritam, te povećala robusnost. U svrhu testiranja, algoritam je najprije implementiran na grafičkoj kartici (engl. *Graphics Processing Unit*, GPU) „Geforce TITAN X“, te je izmjereno vrijeme rada algoritma. Za rezoluciju slike 1280×960 elemenata slike, te raspon dispariteta $[0, 96]$, izmjereno je vrijeme od 40 milisekundi po slici, odnosno frekvencija rada algoritma za stereo sustav iznosi 25 okvira u sekundi (engl. *Frames Per Second*, FPS). Potom, izvršena je usporedba s polu-globalnim algoritmom za spajanje (engl. *Semi Global Matching*, SGM), te MPV-NLC algoritmom u uvjetima kiše, snijega, te lošeg osvjetljenja. Usporedba je vršena na vlastitom skupu podataka, s označenim stvarnim vrijednostima, na način da se dobivene vrijednosti mape dispariteta uspoređuju sa stvarnim vrijednostima. Korištene su tri mjere za usporedbu: preciznost, odziv, te F1 mjera (harmonijska sredina preciznosti i odziva). Rezultati testiranja govore da predloženi algoritam daje bolje rezultate od obaju usporednih algoritama, prema F1 mjeri, u uvjetima kiše i snijega, za oko 2 posto. Iz navedenog je vidljivo da predložena metoda uklanja veliku količinu grešaka u dobivenim rezultatima dispariteta u različitim vremenskim uvjetima.

Računanje cijene spajanja stereo para slika odnedavno je drastično poboljšano uporabom dubokih konvolucijskih neuronskih mreža (engl. *Convolutional Neural Network*, CNN). No, performanse takvog pristupa uvelike ovise o skupu podataka, te zahtijevaju veliku količinu istinitih podataka, što iziskuje mnogo vremena. Kako bi se ovaj problem riješio, autori rada [21] predlažu novo programsko okruženje (engl. *framework*) za učenje CNN-a, sa svrhom računanja cijene spajanja, nenadziranim učenjem (engl. *unsupervised learning*). Predložena metoda spaja učenje u domeni slike s epipolarnim ograničenjem. Iskorištenjem dosljednosti podudaranja stereo para slika odabiru se navodni pozitivni uzorci u svakom koraku treniranja, te se upotrebljavaju za treniranje neuronske mreže. Nadalje se predlaže shema propagacije pozitivnog uzorka, za dobivanje dodatnih trening uzoraka. Predložena nenadzirana metoda testirana je na dvjema arhitekturama neuronskih mreža, jednostavnoj i preciznoj CNN, te je uspoređena s najnovijim metodama, kako nadziranog, tako i nenadziranog učenja. Metode su uspoređivane nad javno dostupnim „Kitti“, „Middlebury“, „HCI“ [22], te „Yonsei“ [23] skupovima podataka. Bez uporabe istinitih disparitetnih vrijednosti, predložena metoda polučila je bolje rezultate od usporednih metoda, pri različitim razinama osvjetljenja, te uvjetima vožnje. Navedeno ide u prilog tome da predloženo programsko okruženje može biti od koristi budućim ADAS-ima.

3. RAZVOJ RJEŠENJA ZA ODREĐIVANJE UDALJENOSTI VOZILA ZASNOVANOG NA KORIŠTENJU STEREO PARA SLIKA I NJEGOVA IMPLEMENTACIJA NA REALNOJ ADAS PLATFORMI

U ovom je poglavlju detaljno predstavljeno rješenje zadatka diplomskog rada. Najprije je opisano sklopovlje i programsko okruženje na kojima je rješenje implementirano. Potom, opisan je koncept rješenja zadatka diplomskog rada, te je predstavljeno programsko rješenje zadatka diplomskog rada. Na kraju, predstavljen je način pokretanja programskog rješenja na sklopovskoj platformi.

Zadatak ovog diplomskog rada, kao što je navedeno u uvodnom poglavlju, razvoj je rješenja za određivanje udaljenosti detektiranog vozila, zasnovanog na korištenju stereo para slika. To rješenje potrebno je implementirati na realnu ADAS platformu, te njegove performanse usporediti s performansama rješenja za određivanje udaljenosti detektiranog vozila iz dostupne slike sa samo jedne kamere, koje također treba biti implementirano na ADAS sklopovskoj platformi. Kako je to izvedeno, objašnjeno je u nastavku.

3.1. Pregled tehnologija upotrjebljenih za rješavanje problema određivanja udaljenosti detektiranog vozila zasnovanog na korištenju stereo para slika

Kompletno programsko rješenje zadatka ovog diplomskog rada izvedeno je u Vision SDK (engl. *Software Development Kit*) programskoj platformi, te je implementirano na ADAS ALPHA razvojnoj ploči. Obje su navedene tehnologije detaljnije opisane u nastavku.

3.1.1. Vision SDK programska platforma

Vision SDK više-procesorska je razvojna programska platforma namijenjena „Texas Instruments (TI)“ obitelji ADAS-a na čipu (engl. *System on Chip*, SoC). Ovo programsko okruženje omogućava korisnicima stvaranje različitih tokova podataka ADAS aplikacije, poput snimanja videa, pred-procesiranja videa, algoritama za video analitiku, te prikaza videa. Vision SDK posjeduje primjere ADAS tokova podataka, koji rade na različitim procesorima i sklopovskim ubrzivačima na ADAS SoC-u, te demonstriraju korisniku kako efektivno koristiti različite SoC podsustave. Programska platforma bazirana je na programskom okruženju nazvanom „Links and Chains“, a korisničko sučelje (engl. *Application Programming Interface*, API) prema ovom programskom okruženju nazvano je „Link API“. SDK instalacijski paket uključuje sve alate

i komponente potrebne za izgradnju aplikacija, uključujući alate za kodiranje, osnovni ulazno/izlazni sustav (engl. *Basic Input/Output System*, BIOS), sustav za među-procesorsku komunikaciju (engl. *Inter-Process Communication*, IPC), programski paket koji omogućuje podršku za ARM i DSP TI procesore platformama koje se ne temelje na operacijskom sustavu („StarterWare“), upravljačke programe paketa za podršku ploči (engl. *Board Support Package*, BSP), stogove umrežavanja, programe za kodiranje/dekodiranje (engl. *codecs*), jezgre algoritama, itd. [24].

Slijede najbitnije Vision SDK karakteristike:

- pruža programsko okruženje i srednji programski sloj (engl. *middleware*) koji omogućava korisniku pokretanje različitih algoritama na DSP/EVE/A15 procesorima;
- pruža programsko okruženje i programski srednji sloj koje korisnici mogu koristiti u svojim sustavima;
- pruža spremnik, izravan pristup memoriji (engl. *Direct Memory Access*, DMA), upravljanje podatkovnim i kontrolnim tokovima kroz različite procesore unutar sustava;
- omogućava jednostavnu integraciju i vrednovanje korisničkih algoritama;
- pruža dobro definiran, dosljedan API između različitih podsustava;
- omogućava fleksibilnost u mijenjanju parametara razine sustava, bez potrebe mijenjanja cijele kodne baze;
- korisnik može posjedovati kodove/algoritme izvan SDK-a, te upotrebljavati samo sučelje prema SDK modulima koji su relevantni za te kodove/algoritme;
- programsko okruženje neće mijenjati API u komponentama (BSP, upravljački programi, BIOS, IPC, DSP/EVE biblioteke) koje su trenutno dostupne na „TDAxxx“ obitelji SoC-eva;
- Vision SDK primjenjiv je na sadašnjim i bit će primjenjiv na budućim ADAS SoC-evima proizvedenim od strane „Texas Instruments-a“;
- Vision SDK primjenjiv je na različitim evaluacijskim modulima (engl. *Evaluation Module*, EVM), odnosno različitim pločama;
- Vision SDK daje podršku za Linux operacijski sustav na A15 procesorima;
- omogućeni su korisni alati (engl. *utility tools*) za generiranje koda slučajeva upotrebe (engl. *use case*), kako bi se olakšao njihov razvoj, za spremanje i čitanje testnih podataka, putem TCP/IP protokola (engl. *Transmission Control Protocol/Internet Protocol*) sa strane računala, za kontrolu aplikacije kalibriranja i podešavanja, putem TCP/IP protokola, za

prebacivanje i pokretanje aplikacije s SD (engl. *Secure Digital*) kartice, ili četverostranog serijskog perifernog sučelja (engl. *Quad Serial Peripheral Interface*, QSPI).

Vision SDK ne nudi:

- automatsko raspoređivanje i izvođenje *vision* jezgara, već korisnik sam poziva dijelove API-a određenim redoslijedom, kako bi izveo algoritam;
- automatsko balansiranje opterećenjem različitih procesora, već je to zadatak korisnika;
- mogućnost izvođenja ugrađenih algoritama u svim testnim slučajevima, već su oni samo referentni;
- integraciju Auto-SAR (engl. *Automotive open System ARchitecture*) komponenti;
- nezavisnost od operacijskog sustava, već se programsko okruženje temelji na BIOS-u, dok je Linux podržan na A15 procesoru;
- među-procesorsku komunikaciju izvan „TDA2xx“ obitelji SoC-eva.

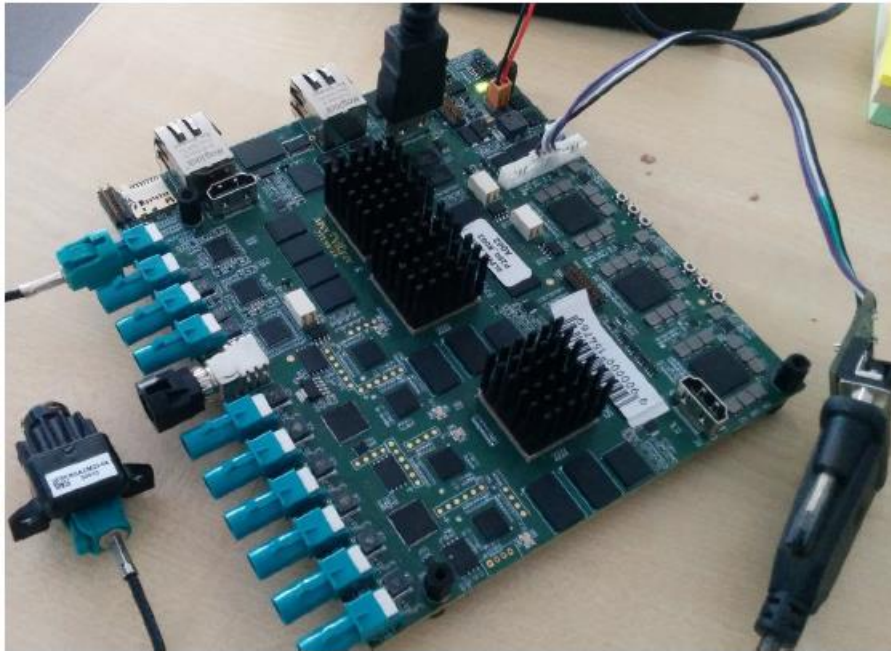
3.1.2. ADAS APLHA razvojna ploča

ADAS APLHA razvojna ploča (Slika 3.1.) dizajnirana je na način da pruža podršku za:

- osnovne i napredne sustave upozoravanja;
- aktivne sustave za upravljanje;
- polu-autonomne operacije.

Skup ciljanih aplikacija obuhvaća funkcionalnosti vezane za sustav kamera na vozilu:

- pogled odostraga (engl. *rear view*);
- prepoznavanje prometnih znakova (engl. *traffic sign recognition*);
- pogled oko vozila (engl. *surround view*);
- pomoć pri parkiranju i u prometu (engl. *parking and traffic assistance*);
- praćenje vozača unutar kabine (engl. *in-cabin driver monitoring*);
- automatska vožnja na autocesti (engl. *highway lane self-driving*);
- krstarenje prilagodljivom brzinom (engl. *adaptive cruise control*);
- itd.



Sl. 3.1. Fotografija ADAS ALPHA ploče [1].

ALPHA ploča upotrijebljena prilikom rješavanja zadatka ovog diplomskog rada sastoji se od tri SoC-a, oznake „TDA2SX“, koje je razvila tvrtka „Texas Instruments“, a koji su namijenjeni za različite stvari:

- „SoC #1“, ili „TDA2SX #1“, odnosno „SC SoC“ (engl. *Surround Camera*) upotrebljava se za pogled oko vozila;
- „SoC #2“, ili „TDA2SX #2“, odnosno „FFN SoC“ (engl. *Front view camera near angle stereoscopic view, Front view camera wide angle, Night vision camera*) upotrebljava se za pogled prema naprijed širokokutnim, odnosno uskokutnim kamerama, te za noćni pogled;
- „SoC #3“, ili „TDA2SX #3“, odnosno „FUS SoC“ (engl. *FUSion*) upotrebljava se za spoj signala dobivenih s kamera „FFN“ i „SC“ SoC-eva.

Najvažnije su značajke jednog SoC-a (sva tri SoC-a imaju iste značajke):

- dvije „ARM Cortex A15“ jezgre s frekvencijom do 1150 megahertza po jezgri;
- dvije „ARM Cortex M4“ jezgre s frekvencijom do 200 megahertza po jezgri;
- dvije „next-generation C66x fixed/floating-point DSP“ jezgre s frekvencijom do 750 megahertza;
- sklop za kodiranje/dekodiranje *Full HD* videa;
- sklopovski ubrzivač za obradu slike i grafike (engl. *graphics engine*);
- četverojezgreni EVE procesor;

- jednojezgreni VPE (engl. *Video Processing Engine*) procesor;
- 1,5 gigabajt DDR3 (engl. *Double Data Rate*) memorije;
- 32 megabajta brzo pristupne (engl. *flash*) memorije;
- EEPROM (engl. *Electrically Erasable Programmable Read-Only Memory*) memorija spojena preko I2C (engl. *Inter-Integrated Circuit*) veze;
- jedan HDMI (engl. *High-Definition Multimedia*) konektor;
- jedan JTAG (engl. *Joint Test Action Group*) konektor;
- jedan UART (engl. *Universal Asynchronous Receiver-Transmitter*) konektor;
- dva DCAN konektora;
- jedan ulaz za microSD karticu;
- pet I2C konektora;
- četiri SPI (engl. *Serial Peripheral Interface*) sučelja;
- jedno QSPI sučelje.

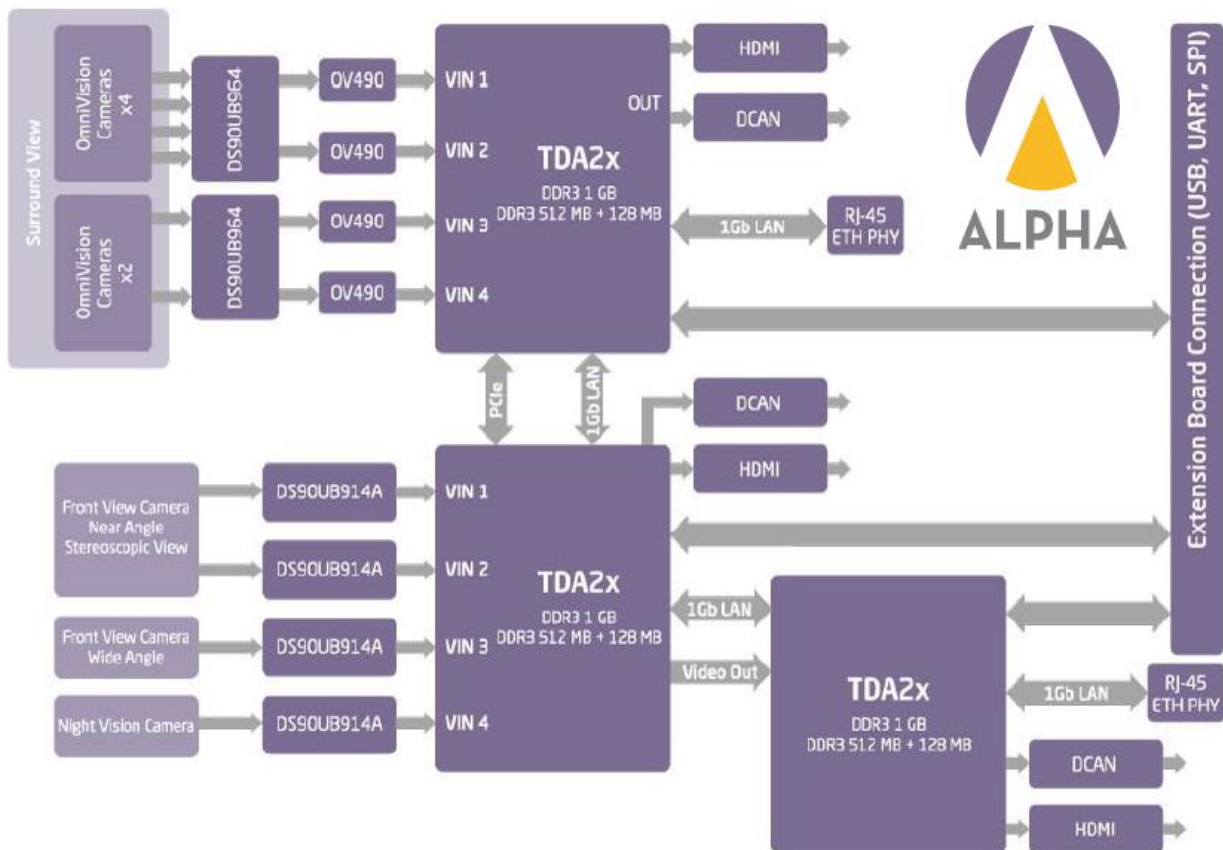
Svaki SoC moguće je pokrenuti u tri različita moda:

- *Debug* (JTAG) mod;
- SD mod;
- QSPI *Flash* mod.

Odabir moda vrši se podešavajući prekidače na samoj ploči. Na ploči se dodatno nalaze dva Ethernet konektora i 10 konektora za kamere (šest za „SC SoC“ i četiri za „FFN SoC“). SoC-ovi na ploči povezani su i međusobno, pri čemu su:

- svi SoC-ovi međusobno nanizano povezani Ethernet vezom od jednog gigabita po sekundi;
- „SC SoC“ i „FFN SoC“ povezani su PCIe (engl. *Peripheral Component Interconnect Express*) vezom brzine do pet gigabita po sekundi;
- „FFN SoC“ i „FUS SoC“ povezani su video kanalom, gdje „FFN SoC“ predaje signal „FUS SoC-u“.

Slika 3.2. prikazuje dijagram veza na ploči [1].



Sl. 3.2. Veze među pojedinim dijelovima ADAS ALPHA ploče [1].

Na ploči se nalaze tri ulaza za microSD karticu, pri čemu svaki ulaz odgovara pojedinom SoC-u. Kartice se upotrebljavaju kao jedan od načina pokretanja pojedinog SoC-a, na način da se na karticu prethodno postavi program koji se želi izvršavati.

3.2. Koncept rješenja problema određivanja udaljenosti detektiranog vozila na temelju stereo para slika

Cjelokupno rješenje zadatka diplomskog rada odrađeno je u Vision SDK programskoj platformi, te je rješenje implementirano na ADAS ALPHA razvojnoj ploči. Prvi korak razvoja rješenja bila je kalibracija stereo sustava kamera. Općeniti principi kalibracije kamere i dobivanja ispravljenog stereo para slika objašnjeni su u dijelovima 2.3.1. i 2.3.2.. Kalibracija kamera izvedena je pomoću programskog alata za kalibraciju kamere, ugrađenog u Vision SDK. Ovaj se alat zasniva na OpenCV programskoj biblioteci i pisan je u C++ programskom jeziku. Procesom kalibracije kamere dobiveni su kalibracijski parametri potrebni za ispravljanje stereo para slika, te računanje udaljenosti vozila od stereo sustava kamera. Potrebno je naglasiti da se kalibracija

obavlja prije pokretanja algoritma (*offline*), te ju je dovoljno obaviti samo jednom, nakon učvršćivanja konstrukcije na kojoj se kamere nalaze.

Nakon obavljene kalibracije stereo sustava kamera, počinje izrada algoritma za detekciju vozila na ispravljenom stereo paru slika. Općeniti princip rada algoritma za prepoznavanje objekata, čiji je podzadatak detekcija objekata, objašnjen je u podpoglavlju 2.2.. Kada je algoritam za detekciju vozila implementiran, prelazi se na izradu algoritma za izračun mape dispariteta na temelju stereo para slika. Prvi dio ovog algoritma uključuje ispravljanje stereo para slika na temelju kalibracijskih parametara. Drugi i glavni dio algoritma predstavlja izračunavanje disparitetne mape na temelju prethodno ispravljenog stereo para slika. Kada je i ovaj dio odrađen, spojene su funkcionalnosti algoritma za detekciju vozila i algoritma za dobivanje mape dispariteta na temelju stereo para slika, kako bi se mogla izračunati udaljenost detektiranog vozila, primjenom kalibracijskih parametara nad vrijednostima mape dispariteta koje se odnose na lokacije onih elemenata slike, koji se nalaze unutar graničnog okvira detektiranog vozila. Općeniti princip određivanja udaljenosti objekata na temelju stereo para slika objašnjen je u podpoglavlju 2.3.

Usporedbom sa stvarnim vrijednostima, te s rezultatima algoritma za određivanje udaljenosti zasnovanog na korištenju slike dobivene s jedne kamere, došlo se do zaključka da rezultati udaljenosti detektiranog vozila, dobiveni dosad opisanim konceptualnim rješenjem, nisu zadovoljavajući. Iz tog razloga, primijenjen je filter za brzo globalno zaglađivanje (engl. *Fast Global Smoothing filter*, FGS filter) nad disparitetnom mapom, predstavljen u radu [25], a kojeg upotrebljava i OpenCV biblioteka. Ovaj filter predstavlja efikasnu tehniku zaglađivanja dijelova slike, čuvajući rubove na njoj prikazanih objekata. Princip rada filtera temelji se na aproksimaciji rješenja memorijski i računski osjetljivih velikih linearnih sustava, definiranih preko d -dimenzionalne prostorne domene, rješavanjem slijeda jednodimenzionalnih podsustava. Odvojiva implementacija omogućava primjenu algoritma vremenski linearne tro-dijagonalne matrice, kako bi se izračunalo d Laplace-ovih matrica s tri točke, iterativno. Filter predstavlja kombinaciju najboljih karakteristika dviju paradigmi, tj. efektivno očuvanje rubova i zaglađivanje zasnovano na optimizaciji.

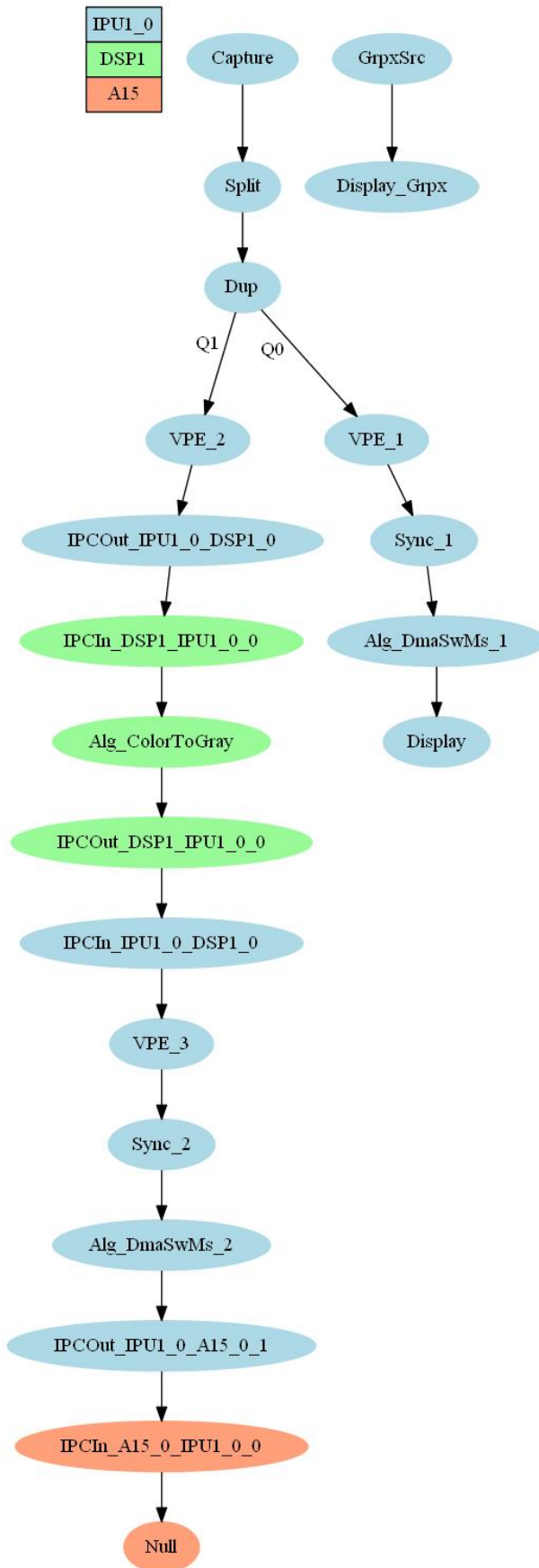
3.3. Programsko rješenje problema određivanja udaljenosti detektiranog vozila zasnovanog na korištenju stereo para slika

U ovom podpoglavlju detaljno je opisano programsko rješenje problema ovog diplomskog rada, redom kojim se ono razvijalo. Mapa s ispisima kodova svih samostalno razvijenih algoritamskih *link*-ova nalazi se u prilogu P.3.1.

3.3.1. Kalibracija stereo sustava kamera

Kako je navedeno u podpoglavlju 3.2., razvoj rješenja ovog diplomskog rada počeo je kalibracijom stereo sustava kamera. Kako bi se kalibracija provela, kalibracijskom je algoritmu potrebno predati slike lijeve i desne kamere, na kojima se nalazi kalibracijski panel. Dijagram toka slučaja upotrebe za pohranu i prikaz videa, iz kojeg se kasnije odvajaju slike za ulaz u kalibracijski algoritam, nalazi se na slici 3.3.

Capture link prihvaća dva videa sa stereo sustava kamera, te ih prosljeđuje *Split link*-u unutar jednog kanala, a potom ih razdvaja na dva zasebna kanala. *Dup link*, potom, duplicira ta dva kanala, te ih šalje u dva izlazna reda. Dupliciranjem se dijagram raščlanjuje na dvije grane. Grana koja počinje redom *Q0* služi za prikaz snimanog sadržaja na ekranu, dok grana koja počinje redom *Q1* služi za spremanje video sadržaja na računalo. U grani započetoj redom *Q0*, razdvojeni se okviri videa šalju u *VPE_1 link*, koji im mijenja format u YUV 420 *semi-planar* UV, kako bi odgovarali formatu zahtijevanom u *Alg_DmaSwMs_1 link*-u. Uz navedeno, *VPE_1 link* sužava okvire za dvostruko s originalne HD (engl. *High Definition*) rezolucije (1280×720 elemenata slike) na 640×720 elemenata slike rezoluciju. Prije nego okviri stignu do *Alg_DmaSwMs_1 link*-a, moraju se sinkronizirati, što obavlja *Sync_1 link*. Unutar *Alg_DmaSwMs_1 link*-a dva okvira spajaju se u jedan video okvir dvostruko veće rezolucije (1280×720 elemenata slike), te se prosljeđuju *Display link*-u koji ih prikazuje na ekran. Ekran na kojem se video prikazuje s pločom je spojen putem HDMI kabla. U grani započetoj redom *Q1*, razdvojeni se okviri videa šalju u *VPE_2 link*, koji im mijenja format u YUV 422 *interleaved* YUYV, kako bi odgovarali formatu zahtijevanom u *Alg_ColorToGray link*-u. *Alg_ColorToGray link* mijenja okvire videa u boji u crno-bijele (engl. *grayscale*) okvire, kako bi se video brže prenosio na računalo, budući da i kalibracijski algoritam radi s crno-bijelim slikama. Potom, okviri se prosljeđuju trećoj instanci *VPE link*-a, koja im mijenja format u YUV 420 *semi-planar* UV, kako bi odgovarali formatu zahtijevanom u drugoj instanci *Alg_DmaSwMs link*-a. Prije no što okviri stignu do *Alg_DmaSwMs_2 link*-a, kako je i ranije bio slučaj, moraju se sinkronizirati, što obavlja *Sync_2 link*. Unutar *Alg_DmaSwMs_2 link*-a dva se okvira spajaju u jedan video okvir dvostruko veće rezolucije (2560×720 elemenata slike), kako bi se očuvala potpuna informacija obaju video okvira. Na kraju, *Null link* zaprima takve spojene okvire videa, te ih pomoću mrežnog alata Vision SDK programske platforme, te putem Ethernet veze, prosljeđuje računalo u izvornom formatu.



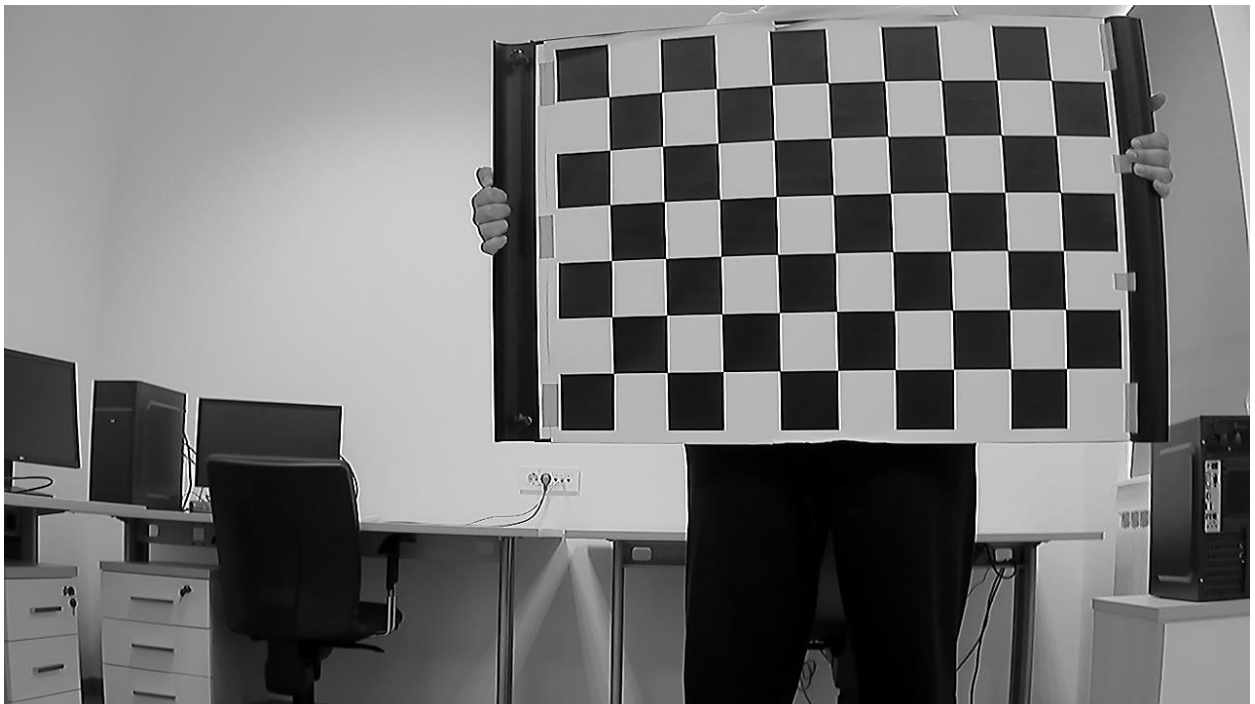
Sl. 3.3. Dijagram toka slučaja upotrebe za pohranu i prikaz videa.

Jednom kada je video, na kojem se nalazi kalibracijski panel iz pogleda obje kamere, zaprimljen na računalo, njegov je sadržaj reproduciran u „yuvplayer“ programu. Pomoću ovog programa, iz snimljenog videa izdvojeni su video okviri od interesa. Za uspješnu kalibraciju, kalibracijski je panel potrebno postaviti u horizontalni položaj, te snimiti pod različitim kutovima u odnosu na kamere, ali koji ne prelaze 45 stupnjeva. Također, kalibracijski se panel treba nalaziti na udaljenosti od kamere, na kojoj je zamišljeno da radi i konačni algoritam, te treba biti takve veličine, da kada je sniman na odgovarajućoj udaljenosti, zauzima otprilike pola veličine slike svake kamere [26]. Vodeći se ovim naputcima, iz videa je izdvojeno 30 okvira od interesa, jer se u dokumentaciji kalibracijskog alata preporuča da broj slika svake kamere bude veći od 10 [27]. Kako kalibracijski algoritam zaprima slike sa svake kamere zasebno, te u pgm (engl. *Portable Gray Map*), ove funkcionalnosti obavlja Python skripta, uz pomoć OpenCV programske biblioteke. Mapa sa slikama upotrijebljenim kao ulaz u kalibracijski algoritam dana je u prilogu P.3.2. Dvije odvojene slike lijeve i desne kamere, koje čine jedan stereo par slika, a koje su s još 58 slika služile kao ulaz kalibracijskog algoritma, prikazane su slikama 3.4. i 3.5.

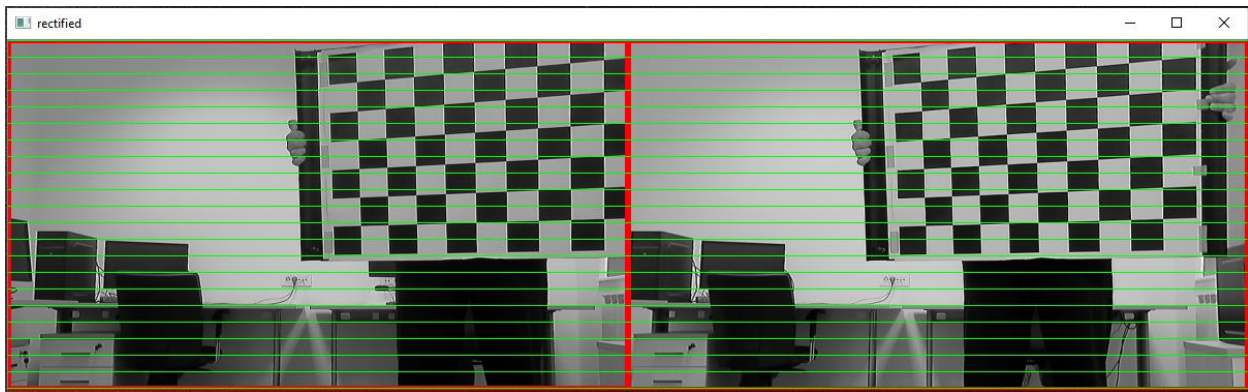
Nakon prikupljenog skupa slika s lijeve i desne kamere, ostao je još samo jedan korak prije kalibracije, a to je postavljanje parametara za kalibracijski algoritam. Ovi se parametri postavljaju u xml (engl. *EXtensible Markup Language*) datoteci, iz koje ih algoritam čita. Neki od bitnijih parametara su: širina panela izražena u broju vrhova u kojima se dva crna kvadrata dodiruju (9), visina panela izražena u broju vrhova u kojima se dva crna kvadrata dodiruju (6), duljina stranice kvadrata izražena u metrima (0,08), te granica reprojekcijske pogreške (engl. *reprojection error threshold*) (0,5). Ispisom rezultata kalibracijskog algoritma dobivena je vrijednost prosječne reprojekcijske pogreške za sliku lijeve kamere od 0,36, uz 13 odbačenih slika zbog prelaska granice reprojekcijske pogreške. Prosječna reprojekcijska pogreška za sliku desne kamere iznosi 0,4, a 11 je odbačenih slika zbog prelaska granice reprojekcijske pogreške. Korijen srednje kvadratne pogreške (engl. *Root Mean Square error*, RMS error) za ispravljanje stereo para slika iznosi 0,51. Od mnogih pokušaja s različitim skupovima slika, najbolji su rezultati dobiveni za navedeni skup slika, te su oni uzeti kao konačni. Ispravljeni par stereo slika 3.4. i 3.5., s iscrtanim epipolarnim linijama, prikazan je na slici 3.6.



Sl. 3.4. Slika s lijeve kamere koja služi kao ulaz u kalibracijski algoritam.



Sl. 3.5. Slika s desne kamere koja služi kao ulaz u kalibracijski algoritam.



Sl. 3.6. Ispravljeni stereo par slika 3.4 i 3.5., s iscrtanim epipolarnim linijama.

3.3.2. Algoritam za detekciju vozila na ispravljenom stereo paru slika

Nakon obavljene kalibracije stereo sustava kamera, počinje izrada algoritma za detekciju vozila na ispravljenom stereo paru slika. Ovaj algoritam od velike je važnosti za rad konačnog rješenja, jer informacija o detektiranom vozilu, odnosno informacija o njegovom graničnom okviru, određuje dio disparitetne mape za koji će se računati udaljenost. Dijagram toka slučaja upotrebe za detekciju vozila na ispravljenom stereo paru slika, nalazi se u prilogu P.3.3.

Alg_RemapMerge link odrađuje ispravljanje odvojenih video okvira, na temelju prethodno dobivenih kalibracijskih parametara. Tim se ispravljenim video okvirima rezolucija smanjuje na 640×360 elemenata slike u *VPE_orgdisp link*-u, te se oni spajaju u jedinstveni video okvir pomoću *Alg_DmaSwMs link*-a. Takav se video okvir prosljeđuje *link*-u za pred-obrađu slike, koji se naziva *Alg_ImagePreProcess*. Nakon toga, video okvir šalje se u *Alg_ImgPyramid link*, gdje se stvara piramida slike (engl. *image pyramid*), koja predstavlja hijerarhijsku reprezentaciju video okvira, od visoke do niske rezolucije, sa svrhom izdvajanja željenih značajki i smanjenja količine šuma [28]. Značajke se, na temelju piramide slike, izdvajaju u *Alg_FPCompute link*-u, te se na temelju njih vrši detekcija objekata, pomoću *Alg_ObjectDetection link*-a. *Merge_objectDetect link* spaja dva ulazna reda, od kojih jedan sadrži informaciju o detektiranim objektima, a drugi piramidu slike, u jedan izlazni red. Informacija sadržana u tom izlaznom redu sinkronizira se, te se šalje algoritmu da klasifikaciju, naziva *Alg_ObjectClassification link*, koji u ovom slučaju klasificira vozila. Na kraju, informacija o detektiranom vozilu, zajedno s ispravljenim parom video okvira, prosljeđuje se *Alg_ObjectDraw link*-u, koji iscrtava granične okvire na ispravljenom paru video okvira, u slučaju da su vozila detektirana, te takvu sliku šalje za prikaz na ekran. Vozilo detektirano na ispravljenom paru video okvira prikazano je na slici 3.7.



Sl. 3.7. Vozilo detektirano na ispravljenom paru video okvira.

3.3.3. Algoritam za izračun mape dispariteta zasnovan na korištenju stereo para slika

Algoritam za izračun mape dispariteta na temelju ispravljenog stereo para slika najbitniji je dio konačnog rješenja, jer se pomoću njega stvara mapa dispariteta slike lijeve kamere, na temelju čijih se vrijednosti, primjenom kalibracijskih parametara, računa udaljenost objekata scene, a posljedično i detektiranog vozila. Dijagram toka slučaja upotrebe za izračun mape dispariteta na temelju ispravljenog stereo para slika, dan je u prilogu P.3.4.

Nakon ispravljanja ulaznog para video okvira, dijagram se dijeli na dvije grane. Grana koja počinje redom $Q0$ služi za prikaz ispravljenog stereo para video okvira na dijelu ekrana, dok grana koja počinje redom $Q1$ služi za izračun mape dispariteta na temelju ispravljenog stereo para video okvira, koji se potom prikazuje na drugom dijelu istog ekrana. Izračun mape dispariteta počinje *Alg_Census link*-om, koji obavlja *census* transformaciju. *Census* transformacija za svaki element slike računa binarni niz, koji se naziva *census* potpis, uspoređujući monokromatsku vrijednost elementa slike s monokromatskim vrijednostima njegovog susjedstva. *Census* potpis sadrži informaciju o tome imaju li susjedni elementi slike manju monokromatsku vrijednost od referentnog elementa slike, ili ne [29]. Takva se transformirana slika šalje *Alg_DisparityHamDist link*-u, koji obavlja izračun mape dispariteta video okvira dobivenog s lijeve kamere, na temelju Hamming-ove udaljenosti odgovarajućih elemenata slike video okvira dobivenih s lijeve i desne kamere. Početna se disparitetna mapa, zajedno s ispravljenim stereo parom video okvira, prosljeđuje algoritmu za post-procesiranje, odnosno *Alg_StereoPostProcess link*-u. Ovdje se mapa dispariteta dodatno filtrira na temelju mape vjerojatnosti ispravnog sparivanja odgovarajućih elemenata slika dobivenih s lijeve i desne kamere (engl. *confidence map*), te se elementima disparitetne mape dodjeljuje boja koja odgovara udaljenosti točke trodimenzionalne scene, koju taj element predstavlja, od kamere. Pritom toplije boje označavaju manju udaljenost točke trodimenzionalne scene, koja odgovara određenom elementu mape dispariteta, od kamere, a hladnije veću. Takva se mapa dispariteta u boji prosljeđuje na ekran, gdje se iscrtava zajedno s ispravljenim stereo parom video okvira. Obojana disparitetna mapa, zajedno s odgovarajućim ispravljenim parom video okvira, prikazana je na slici 3.8.



Sl. 3.8. Obojana disparitetna mapa prikazana s odgovarajućim ispravljenim parom video okvira.

3.3.4. Početna verzija konačnog rješenja

Spajanjem algoritma za detekciju vozila na ispravljenom stereo paru slika s algoritmom za izračun mape dispariteta zasnovanim na korištenju stereo para slika, dobila se početna verzija konačnog rješenja zadatka diplomskog rada. Dijagram toka slučaja upotrebe početne verzije konačnog rješenja dan je u prilogu P.3.5.

Ispravljeni se stereo par video okvira udvaja, te se u desnoj grani, koja počinje redom $Q0$, odvija detekcija vozila, dok u lijevoj grani, koja počinje redom $Q1$, algoritam izračunava disparitetnu mapu na temelju ulaznog ispravljenog para video okvira. Informacija o lokaciji i veličini graničnog okvira, koja se dobije detekcijom vozila, iz *Alg_A_ObjectDraw link-a* prosljeđuje se u *Alg_A_StereoPostProcess link*. Glavna je razlika *Alg_A_ObjectDraw link-a* u odnosu na *Alg_ObjectDraw link*, postojanje dodatnog izlaznog reda u kojeg se prosljeđuje informacija o lokaciji i veličini graničnog okvira. Slično, glavna razlika *Alg_A_StereoPostProcess link-a* i *Alg_StereoPostProcess link-a* dodatni je ulazni red koji zaprima informaciju o lokaciji i veličini graničnog okvira. Na temelju te informacije, izračunava se udaljenost detektiranog vozila, primjenom kalibracijskih parametara nad vrijednostima mape dispariteta. Ispis programskog koda za računanje i ispisivanje u terminal udaljenosti detektiranog vozila, na temelju prosječne i medijalne vrijednosti dispariteta unutar granica graničnog okvira, dan je na slici 3.9.

Linija ***Kod***

```
1:      /*žarišna udaljenost lijeve kamere (f) i udaljenost između žarišta
2:      lijeve i desne kamere (l)*/
3:      const float f = 1261.639469;
4:      const float l = 0.156498;
5:
6:      /*izračun udaljenosti detektiranog vozila na temelju srednje
7:      vrijednosti dispariteta unutar granica graničnog okvira te ispis
8:      u terminal*/
9:      float D;
10:     uint8_t d;
11:     d = (float)suma[numObjLeft]/(float)br[numObjLeft];
12:     D = f*l/d;
13:     Vps_printf("Depth mean[%d]: %3.2f", numObjLeft, D);
14:
15:     /*izračun udaljenosti detektiranog vozila na temelju medijalne
16:     vrijednosti dispariteta unutar granica graničnog okvira te ispis
17:     u terminal*/
18:     d = (float)pDm->inputArray[numObjLeft][(br[numObjLeft]/2)];
19:     D = f*l/d;
20:     Vps_printf("Depth median[%d]: %3.2f", numObjLeft, D);
```

Sl. 3.9. Kod za računanje udaljenosti detektiranog vozila, na temelju prosječne i medijalne vrijednosti dispariteta unutar granica graničnog okvira te ispis u terminal.

Kao što je prikazano na slici 3.9., udaljenost se na temelju vrijednosti dispariteta i kalibracijskih parametara računa prema jednadžbi (2.7.), a samo se oznake za udaljenost razlikuju, tako da je na slici oznaka D , a u jednadžbi je odgovarajuća oznaka l_z_i . Varijabla *numObjLeft* predstavlja broj detektiranih vozila, odnosno broj graničnih okvira, na video okviru dobivenom s lijeve kamere. Oznaka *br* predstavlja jednodimenzionalno polje, gdje svaki element polja predstavlja broj disparitetnih vrijednosti unutar odgovarajućeg graničnog okvira. Jednodimenzionalno polje, kod kojeg svaki element polja predstavlja sumu disparitetnih vrijednosti unutar odgovarajućeg graničnog okvira, u ispisu koda na slici 3.9. naziva se *suma*. Za izračun medijalne vrijednosti dispariteta upotrebljava se pokazivač na strukturu, koja sadrži dvodimenzionalno polje sa sortiranim vrijednostima dispariteta, unutar odgovarajućeg graničnog okvira. Ovaj se pokazivač naziva *pDm*, dok je ime dvodimenzionalnog polja *inputArray*.

Rezultati ove metode uspoređeni su s rezultatima metode za izračun udaljenosti detektiranog vozila zasnovane na korištenju slike dobivene s jedne kamere. Algoritam za izračun udaljenosti na temelju graničnog okvira iscrtanog na video okviru dobivenom s lijeve kamere, čiji

je princip rada dan jednadžbama (2.1.) i (2.2.), također je implementiran unutar *Alg_A_StereoPostProcess* link-a, a njegov je programski kod prikazan na slici 3.10.

Linija Kod

```
1:      /*izračun kuta pod kojim je cijeli objekt vidljiv kameri*/
2:      const float alpha = 0.5411;
3:      float ef;
4:      ef = (pBb->Height[i])*0.5411/height;
5:
6:      /*izračun udaljenosti detektiranog vozila i ispis u terminal*/
7:      const float H = 1.5;
8:      float ef2 = ef/2;
9:      D = H/(2*tan(ef2));
10:     Vps_printf("Single cam distance[%d]: %3.2f", numObjLeft, D);
```

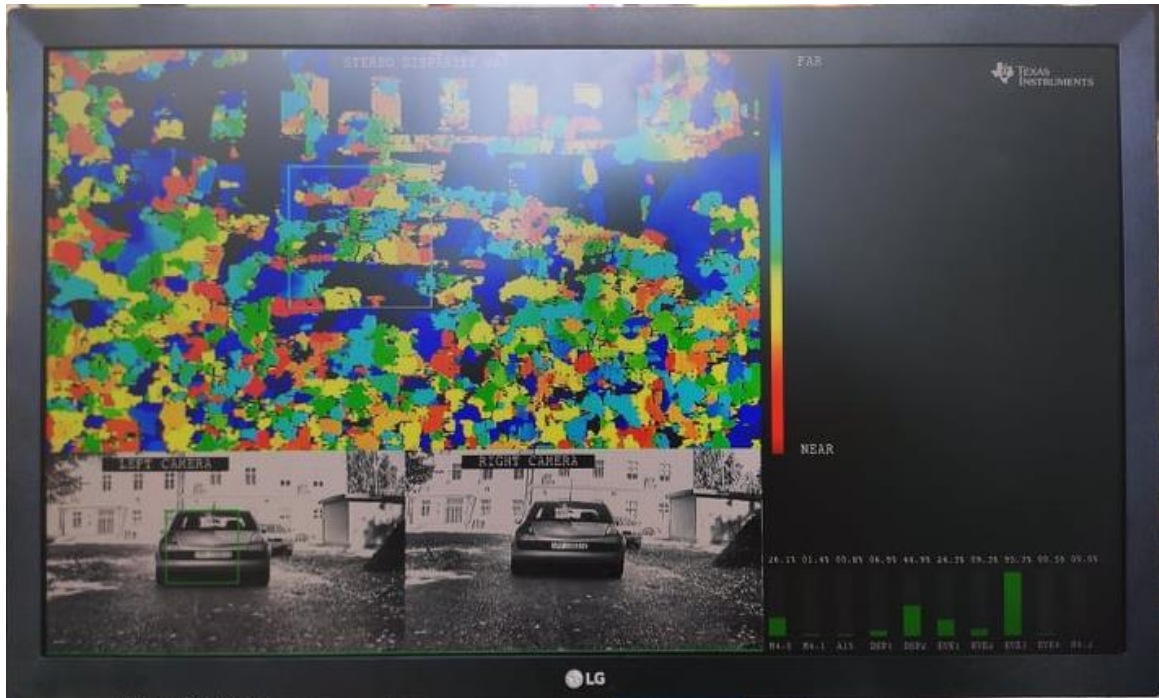
Sl. 3.10. Kod za računanje udaljenosti detektiranog vozila zasnovan na korištenju video okvira dobivenog s lijeve kamere te ispis u terminal.

Postoji jedna razlika u oznakama između jednadžbi (2.1.), odnosno (2.2.) i ispisa koda na slici 3.10., a to je oznaka za visinu graničnog okvira. Visina graničnog okvira u ispisu koda označava se s $pBb \rightarrow Height[i]$, gdje je pBb pokazivač na strukturu koja sadrži visinu graničnog okvira, a i redni broj detektiranog okvira. Podatak o prosječnoj visini automobila, koja iznosi 1,5 metara, preuzet je iz [30], usporedbom visina različitih modela automobila, dok je kut α dobiven mjerenjem.

Prikaz rezultata početne verzije konačnog rješenja zadatka diplomskog rada na ekranu, dan je na slici 3.11.

3.3.5. Implementacija FGS filtera

Usporedbom rezultata početne verzije konačnog rješenja zadatka diplomskog rada sa stvarnim vrijednostima te s rezultatima metode za određivanje udaljenosti detektiranog vozila zasnovane na korištenju slike dostupne sa samo jedne kamere, došlo se do zaključka da predloženo rješenje daje lošije rezultate. Objekti trodimenzionalne scene prikazani na mapi dispariteta, kako je vidljivo na slici 3.11., nisu bili jednolično obojani, odnosno vrijednosti dispariteta za isti objekt trodimenzionalne scene bile su raštrkane. Iz tog se razloga pribjegli implementaciji FGS filtera, koji je također implementiran u OpenCV programskoj biblioteci, a upotrebljava se za zaglađivanje površina slike koje predstavljaju isti objekt scene, uz očuvanje rubova tog objekta. Kako je programska biblioteka OpenCV računalno prezahtjevna za implementaciju na ADAS ALPHA



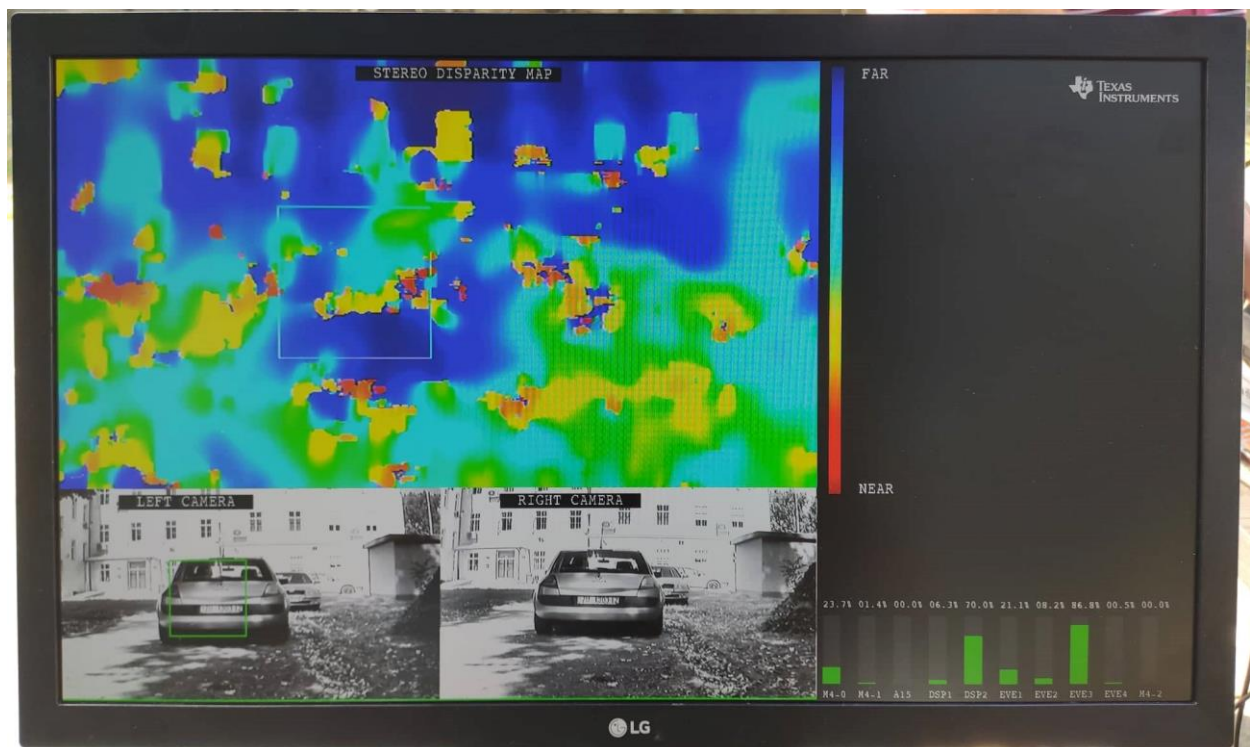
Sl. 3.11. Prikaz rezultata početne verzije konačnog rješenja zadatka diplomskog rada na ekranu.

ploču, upotrijebljeno je programsko rješenje FGS filtera izvedeno kombinacijom programskih jezika C i Matlab, a preuzeto iz [31]. Ovo se programsko rješenje temelji na istom radu na kojem se temelji i implementacija u OpenCV programskoj biblioteci [25].

Uz određene modifikacije, ovo je programsko rješenje u potpunosti izvedeno u C programskom jeziku, te upotrijebljeno pri rješavanju zadatka ovog diplomskog rada. FGS filter implementiran je u dva koraka. Prvi je korak bez navođenja, dok se u drugom koraku video okvir dobiven s lijeve kamere upotrebljava za navođenje. Dijagram toka slučaja upotrebe za određivanje udaljenosti detektiranog vozila zasnovanog na stereo paru slika, uz implementirani FGS filter bez navođenja, dan je u prilogu P.3.6.

Glavna je razlika između slučaja upotrebe prikazanog dijagramom toka u prilogu P.3.6. i slučaja upotrebe prikazanog dijagramom toka u prilogu P.3.5. način implementacije algoritma za post-filtriranje. U ovom je slučaju navedeni algoritam razdvojen na dva dijela, te je između njih ubačen algoritam s implementiranim FGS filterom. Prvi dio algoritma za post-filtriranje implementiran je unutar *Alg_A_StereoPost1 link*-a, a služi za filtriranje disparitetne mape na temelju vjerojatnosti ispravnog sparivanja odgovarajućih elemenata slika dobivenih s lijeve i desne kamere. Na takvu se mapu dispariteta primjenjuje FGS filter, implementiran u *Alg_A_FGSfilter link*-u. *Alg_A_FGSfilter link* postavljen je za izvršavanje na A15 procesoru, kako bi se ubrzao rad cjelokupnog rješenja. Na kraju, filtrirana je disparitetna mapa proslijeđuje *Alg_A_StereoDraw link*-

u, gdje se ona iscrtava te se na temelju nje i informacija o graničnom okviru izračunava udaljenost detektiranog vozila. Slika 3.12. prikazuje grafičke rezultate dobivene nakon implementacije FGS filtera bez navođenja.



Sl. 3.12. Prikaz rješenja problema određivanja udaljenosti detektiranog vozila zasnovanog na korištenju stereo para slika, s implementiranim FGS filterom bez navođenja.

Dijagram toka slučaja upotrebe za određivanje udaljenosti detektiranog vozila zasnovanog na korištenju stereo para slika, uz implementirani FGS filter koji video okvir dobiven s lijeve kamere upotrebljava za navođenje, dan je u prilogu P.3.7. Jedina je razlika ovog slučaja upotrebe, od slučaja upotrebe prikazanog dijagramom toka u prilogu P.3.6., ta što se video okvir dobiven s lijeve kamere, nakon ispravljanja prosljeđuje FGS filteru, koji je u ovom slučaju implementiran putem *Alg_A_FGSfilter2* link-a. Prikaz disparitetne mape filtrirane na ovakav način, zajedno s ispravljenim stereo parom video okvira, dan je na slici 3.13.

3.4. Način pokretanja programskog rješenja problema određivanja udaljenosti detektiranog vozila zasnovanog na korištenju stereo para slika

Za pokretanje programskog rješenja problema određivanja udaljenosti detektiranog vozila zasnovanog na korištenju stereo para slika, potrebno je najprije instalirati „Tera Term“ programski paket. Potom, SD karticu s prevedenim programskim kodom potrebno je staviti u odgovarajući



Sl. 3.13. Prikaz rješenja problema određivanja udaljenosti detektiranog vozila zasnovanog na korištenju stereo para slika, s implementiranim FGS filterom koji upotrebljava video okvir dobiven s lijeve kamere za navođenje.

ulaz za SD karticu na ADAS ALPHA ploči, povezan s „SC SoC-em“. Za prikaz grafičkog sadržaja treba spojiti monitor putem HDMI kabla, s odgovarajućim HDMI izlazom na ADAS ALPHA ploči, također povezanim s „SC SoC-em“. Izvršne se naredbe ploči zadaju putem računala, konkretnije putem „Tera Term“ programskog paketa. Kako bi ove naredbe dospjele do ploče, te bile interpretirane na odgovarajući način, ploča i računalo spajaju se pomoću UART kabla. Potrebno je opet pripaziti da se UART kabel na ploču spaja putem odgovarajućeg ulaza, povezanog s „SC SoC-em“. Zadnja stvar koja se tiče fizičkih postavki jest spojiti ADAS ALPHA ploču s napajanjem. Kada je ploča spojena na napajanje, potrebno je namjestiti postavke unutar „Tera Term-a“. Otvaranjem programa prikazuje se na zaslonu prozor „New connection“, gdje se odabire „Serial Port“, te odgovarajući USB (engl. *Universal Serial Bus*) ulaz. Potom, otvara se glavni prozor, unutar kojeg se odabire opcija „Setup“, potom „Serial port“ i na kraju se brzina prijenosa podataka podešava na 115200 baud-a. Ovime završava proces namještanja postavki „Tera Term-a“ te ako je sve pravilno odrađeno, trebao bi se prikazati izbornik unutar „Tera Term-a“, poput onoga na slici 3.14.

```
COM5 - Tera Term VT
File Edit Setup Control Window Help
[IPU1-0]
[IPU1-0] Vision SDK Usecases,
[IPU1-0] -----
[IPU1-0] 1: Single Camera Usecases
[IPU1-0] 2: Multi-Camera LUDS Usecases
[IPU1-0] 3: AUB RX Usecases, (TDA2x & TDA2Ex ONLY)
[IPU1-0] 4: Dual Display Usecases, (TDA2x EUM ONLY)
[IPU1-0] 5: ISS Usecases, (TDA3x ONLY)
[IPU1-0] 6: xCAM Usecases
[IPU1-0] 7: Network RX/TX Usecases
[IPU1-0] 8: Stereo Calibration
[IPU1-0] a: Miscellaneous test's
[IPU1-0]
[IPU1-0] c: ALPHA AMU Usecases
[IPU1-0]
[IPU1-0] f: Stereo Usecases
[IPU1-0]
[IPU1-0] s: System Settings
[IPU1-0]
[IPU1-0] x: Exit
[IPU1-0] Enter Choice:
[IPU1-0]
IHOST 1 18.849752 s: NETWORK_CTRL: Starting Server (port=5000) !!!
IHOST 1 18.849783 s: NETWORK_CTRL: Starting Server ... DONE (port=5000) !!
```

Sl. 3.15. Prikaz prvog izbornika za odabiranje željene ADAS funkcionalnosti, unutar „Tera Term“ programskog paketa.

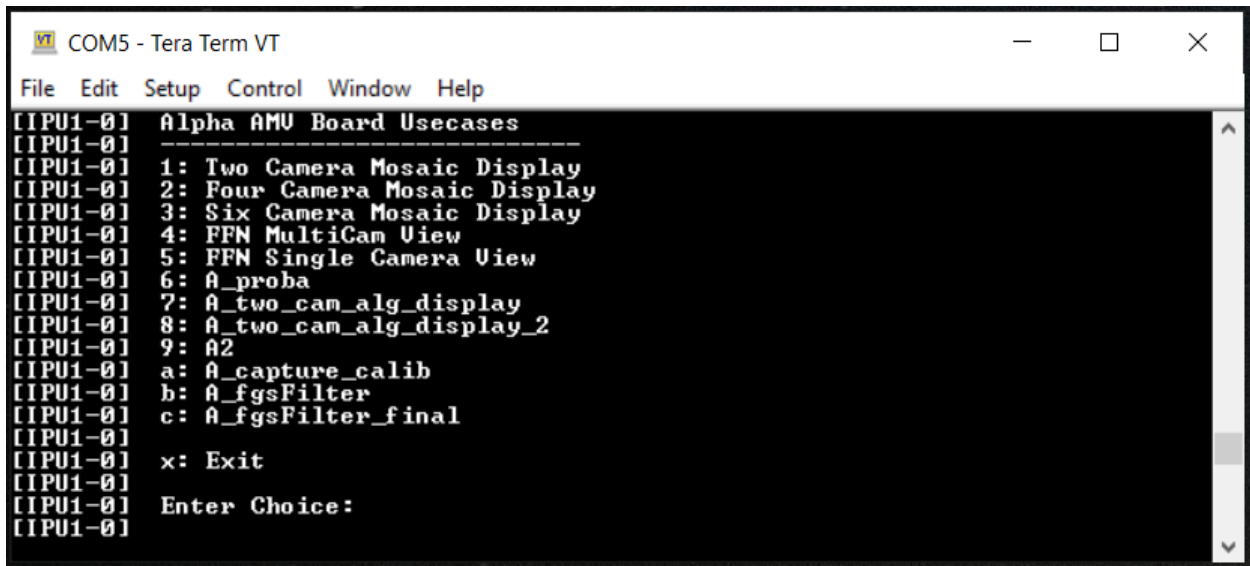
Zadnji je korak samo odabiranje željene ADAS funkcionalnosti. Za pokretanje rješenja za određivanje udaljenosti zasnovanog na korištenju stereo para slika, uz upotrebu FGS filtera s navođenjem, odabere se opcija *c*, nakon čega se pojavi izbornik, poput onoga na slici 3.15.

```
COM5 - Tera Term VT
File Edit Setup Control Window Help
[IPU1-0] Select use-case options,
[IPU1-0] -----
[IPU1-0] 1: Enable Pedestrian Detect (PD)
[IPU1-0] 2: Enable Traffic Sign Recognition (TSR)
[IPU1-0] 3: Enable Vehicle Detect (UD)
[IPU1-0] 4: Enable PD+TSR+UD
[IPU1-0]
[IPU1-0] Enter Choice:
[IPU1-0]
```

Sl. 3.14. Prikaz trećeg izbornika za odabiranje željene ADAS funkcionalnosti, unutar „Tera Term“ programskog paketa.

Potom, ponovno se odabere opcija *c*, koja dovodi do izbornika prikazanog slikom 3.16.

Na kraju, odabere se ponuđena opcija 3, kako bi algoritam za detekciju objekata detektirao vozila. Time se pokrenulo željeno rješenje.



```
COM5 - Tera Term VT
File Edit Setup Control Window Help
[IPU1-0] Alpha AMU Board Usecases
[IPU1-0] -----
[IPU1-0] 1: Two Camera Mosaic Display
[IPU1-0] 2: Four Camera Mosaic Display
[IPU1-0] 3: Six Camera Mosaic Display
[IPU1-0] 4: FFN MultiCam View
[IPU1-0] 5: FFN Single Camera Uiew
[IPU1-0] 6: A_proba
[IPU1-0] 7: A_two_cam_alg_display
[IPU1-0] 8: A_two_cam_alg_display_2
[IPU1-0] 9: A2
[IPU1-0] a: A_capture_calib
[IPU1-0] b: A_fgsFilter
[IPU1-0] c: A_fgsFilter_final
[IPU1-0]
[IPU1-0] x: Exit
[IPU1-0]
[IPU1-0] Enter Choice:
[IPU1-0]
```

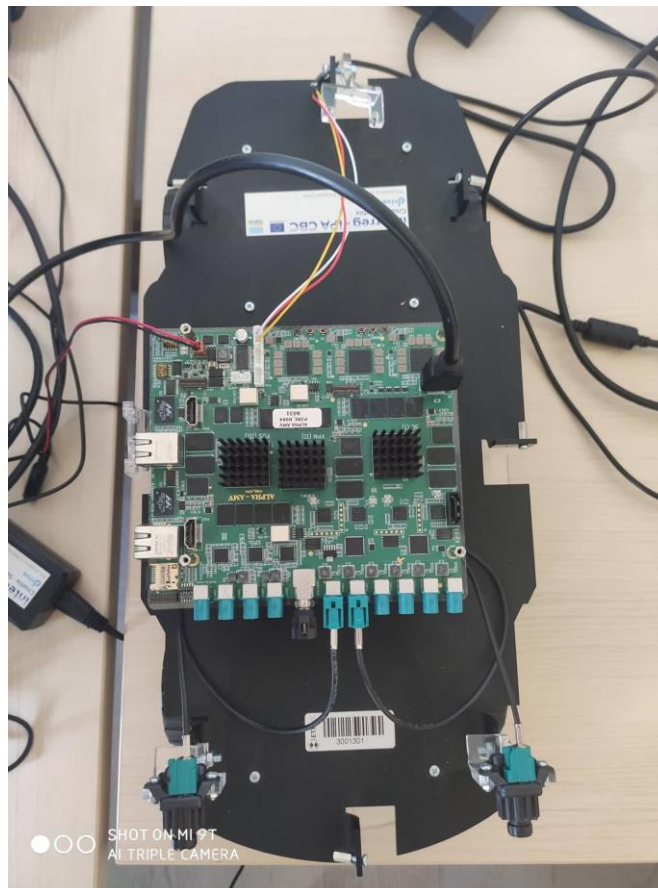
Sl. 3.16. Prikaz drugog izbornika za odabiranje željene ADAS funkcionalnosti, unutar „Tera Term“ programskog paketa.

4. TESTIRANJE RADA PREDLOŽENOG RJEŠENJA ZA ODREĐIVANJE UDALJENOSTI VOZILA ZASNOVANOG NA KORIŠTENJU STEREO PARA SLIKA

U ovom poglavlju opisan je način testiranja konačnog rješenja diplomskog rada te su predstavljeni rezultati. Najprije su opisani testni uvjeti i baza signala korištenih za testiranje. Potom, prezentirani su rezultati provedenog testiranja konačnog rješenja i testiranja dvaju dodatnih rješenja, koji su međusobno uspoređeni. Na kraju, komentirani su dobiveni rezultati te je dan kritički osvrt na predloženo konačno rješenje.

4.1. Baza signala upotrjebljenih za testiranje predloženog rješenja za određivanje udaljenosti vozila zasnovanog na korištenju stereo para slika

Konstrukcija sustava za testiranje rezultata predloženog rješenja diplomskog rada, na kojoj se nalazi ADAS ALPHA ploča sa stereo sustavom kamera, prikazana je na slici 4.1.



Sl. 4.1. Konstrukcija sustava za testiranje rezultata predloženog rješenja diplomskog rada.

Sustav za testiranje rezultata predloženog rješenja diplomskog rada, prikazan na slici 4.1., upotrijebljen je za prikupljanje testnih signala i automatski proračun rezultata. Testni signali predstavljaju video okvire dobivene sa stereo sustava kamera. Svaki je okvir rezolucije 1280×720 elemenata slike, zapisan u izvornom (engl. *raw*) formatu slike s YUV 422 *planar* formatom boja. Brzina snimanja videa iznosi 30 okvira u sekundi.

Sustavom stereo kamera snimana su četiri automobila pojedinačno, na šest različitih udaljenosti. Udaljenosti iznose 3, 4, 6, 8, 10 i 12 metara. Donja granica ovog intervala udaljenosti odabrana je jer minimalna teoretska udaljenost na kojoj se vozilo mora nalaziti, u odnosu na kameru, iznosi 1,54 metra. Uz poznate vrijednosti žarišne duljine lijeve kamere i horizontalne udaljenosti žarišta lijeve i desne kamere, prezentirane na slici 3.8., te maksimalne vrijednosti dispariteta, koja iznosi 128, primjenom jednadžbe (2.7.) moguće je dobiti ranije iskazanu vrijednost minimalne teoretske udaljenosti na kojoj se vozilo mora nalaziti, u odnosu na kameru. Također, kako će biti prikazano u podpoglavlju 4.2., testiranjem je utvrđeno da za udaljenosti vozila od kamere veće od 10 metara, pogreška procjene udaljenosti znatno raste. Stoga je udaljenost od 12 metara izabrana kao gornja granica testnog intervala udaljenosti.

Testiranje je odrađeno po sunčanom vremenu, no sustav za testiranje i snimani automobili nalazili su se u sjeni susjedne zgrade.

4.2. Rezultati testiranja predloženog rješenja za određivanje udaljenosti vozila zasnovanog na korištenju stereo para slika

Rezultati testiranja konačnog rješenja diplomskog rada, koje podrazumijeva određivanje udaljenosti detektiranog vozila zasnovano na stereo paru slika, uz primjenu FGS filtra s navođenjem, čiji se dijagram toka slučaja upotrebe nalazi u prilogu P.3.7., uspoređeni su s rezultatima testiranja početne verzije konačnog rješenja, odnosno određivanja udaljenosti detektiranog vozila zasnovanog na stereo paru slika, bez primjene FGS filtera, čiji se dijagram toka slučaja upotrebe nalazi u prilogu P.3.5. te s testnim rezultatima rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na slici dobivenoj samo s jedne kamere, prikazanog kodom na slici 3.10.

Oba ranije navedena rješenja zasnovana na stereo paru slika, upotrebljavaju tri metode za izračun udaljenosti detektiranog vozila. Prva metoda izračunava udaljenost na temelju srednje vrijednosti elemenata disparitetne mape, koji se nalaze u sredini površine obavijene graničnim okvirom, a zauzimaju četvrtinu te iste površine. Udaljenost detektiranog vozila, pomoću druge

metode, izračunava se na temelju medijana elemenata disparitetne mape, koji predstavljaju iste one elemente od interesa kao i kod prve metode. Zadnja metoda izračunava udaljenost na temelju medijana svakog petog elementa površine disparitetne mape obavijene graničnim okvirom. Mapa sa slikama svih grafičkih rezultata testiranih rješenja nalazi se u prilogu P.4.1.

4.2.1. Prvo testno vozilo

Procijenjene udaljenosti detektiranog prvog testnog vozila, prikazane su tablicom 4.1.

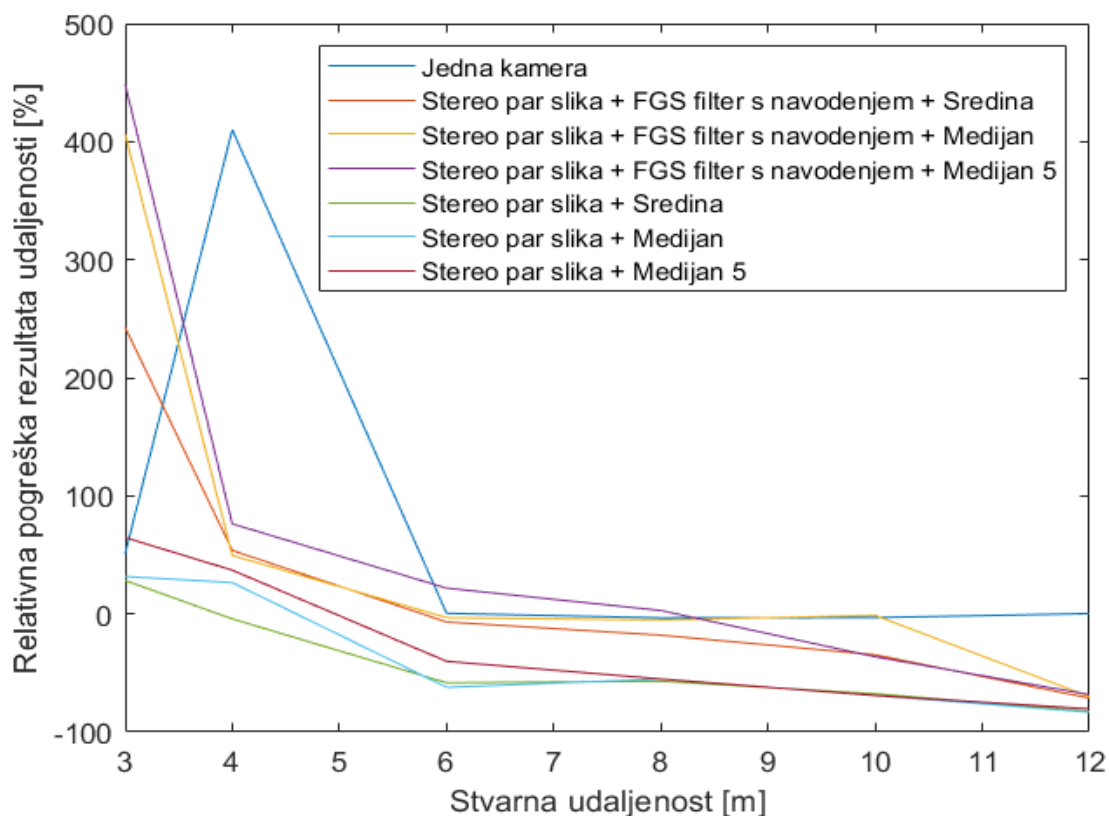
Tablica 4.1. Rezultati udaljenosti detektiranog prvog testnog vozila.

Stvarna udaljenost	Jedna kamera	Stereo sustav bez filtera			Stereo sustav s FGS filterom uz navođenje		
		Sredina	Medijan	Medijan 5	Sredina	Medijan	Medijan 5
3	4,53	3,85	3,95	4,94	10,28	15,19	16,45
4	20,40	3,83	5,06	5,48	6,15	5,98	7,05
6	6,03	2,50	2,27	3,59	5,59	5,81	7,31
8	7,73	3,44	3,59	3,59	6,56	7,59	8,23
10	9,69	3,24	3,09	3,09	6,54	9,87	6,37
12	12,03	2,10	2,01	2,35	3,44	3,59	3,80

Relativna pogreška (engl. *relative error*) dobivenih rezultata udaljenosti detektiranog vozila, računa se prema jednadžbi (4.1.) [32].

$$r = \frac{\hat{x} - x}{x} \times 100 \quad (4.1.)$$

Relativna pogreška označena je s r , a izražena u postotcima, dok \hat{x} predstavlja izračunatu vrijednost, a x stvarnu vrijednost. Slika 4.2. prikazuje graf kretanja relativne pogreške rezultata rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na stereo paru slika, uz primjenu FGS filtera s navođenjem, graf kretanja relativne pogreške rezultata rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na stereo paru slika, bez primjene FGS filtera te graf kretanja relativne pogreške rezultata rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na slici dobivenoj s jedne kamere, za prvo testno vozilo, na različitim udaljenostima.



Sl. 4.2. Usporedba relativne pogreške rezultata rješenja zasnovanog na slici dobivenoj s jedne kamere i relativne pogreške rezultata rješenja zasnovanih na stereo paru slika, uz primjenu FGS filtera s navođenjem te bez njegove primjene, za prvo testno vozilo.

Slika 4.3. prikazuje obojano disparitetnu mapu rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na stereo paru slika, bez primjene FGS filtera, te ispravljeni stereo par slika s detektiranim prvim testnim vozilom, koje je šest metara udaljeno od kamera.

Obojana disparitetna mapa rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na stereo paru slika, uz primjenu FGS filtera s navođenjem te ispravljeni stereo par slika s detektiranim prvim testnim vozilom, koje je šest metara udaljeno od kamera, nalaze se na slici 4.4.

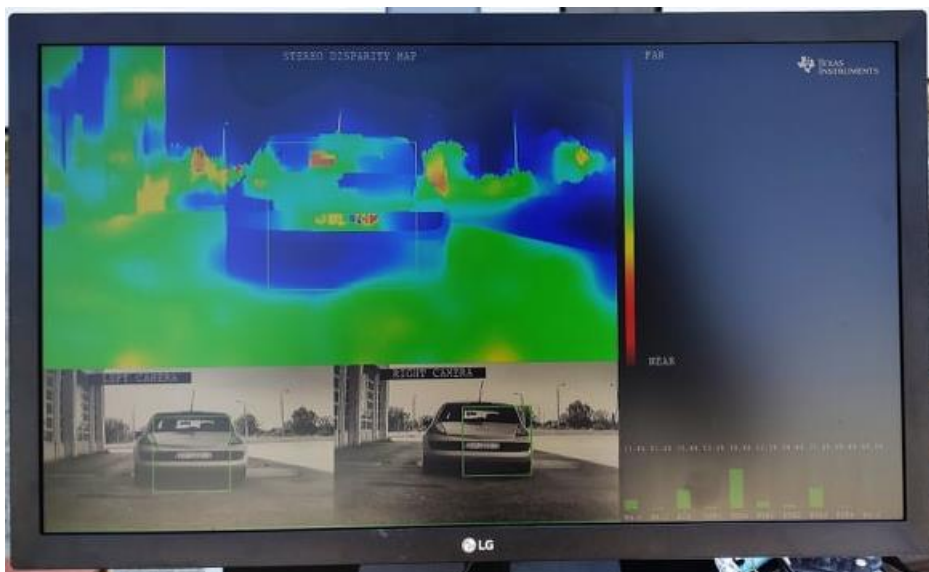
4.2.2. Drugo testno vozilo

Procijenjene udaljenosti detektiranog drugog testnog vozila, prikazane su tablicom 4.2.

Slika 4.5. prikazuje graf kretanja relativne pogreške rezultata rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na stereo paru slika, uz primjenu FGS filtera s navođenjem, graf kretanja relativne pogreške rezultata rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na stereo paru slika, bez primjene FGS filtera te graf kretanja



Sl. 4.4. Obojana disparitetna mapa rješenja zasnovanog na stereo paru slika, bez primjene FGS filtera te ispravljeni stereo par slika s detektiranim prvim testnim vozilom, na udaljenosti od šest metara.



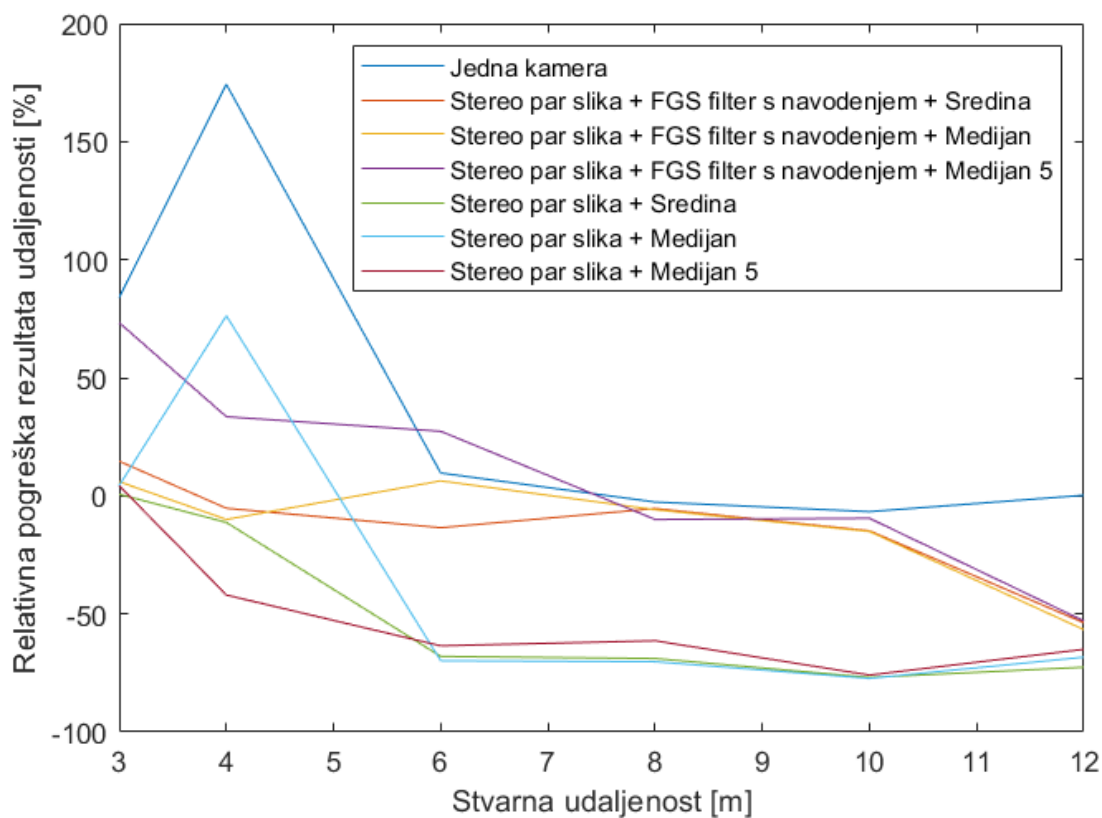
Sl. 4.3. Obojana disparitetna mapa rješenja zasnovanog na stereo paru slika, uz primjenu FGS filtera s navođenjem te ispravljeni stereo par slika s detektiranim prvim testnim vozilom, na udaljenosti od šest metara.

relativne pogreške rezultata rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na slici dobivenoj s jedne kamere, za drugo testno vozilo, na različitim udaljenostima.

Slika 4.6. prikazuje obojano disparitetnu mapu rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na stereo paru slika, bez primjene FGS filtera, te ispravljeni stereo par slika s detektiranim drugim testnim vozilom, koje je šest metara udaljeno od kamera.

Tablica 4.2. Rezultati udaljenosti detektiranog drugog testnog vozila.

Stvarna udaljenost	Jedna kamera	Stereo sustav bez filtera			Stereo sustav s FGS filterom uz navođenje		
		Sredina	Medijan	Medijan 5	Sredina	Medijan	Medijan 5
3	5,52	3,02	3,13	3,13	3,44	3,18	5,20
4	10,97	3,55	7,05	2,32	3,79	3,60	5,34
6	6,58	1,92	1,81	2,19	5,19	6,38	7,64
8	7,79	2,49	2,38	3,09	7,57	7,53	7,20
10	9,33	2,31	2,27	2,41	8,52	8,49	9,05
12	12,03	3,29	3,80	4,20	5,55	5,20	5,64



Sl. 4.5. Usporedba relativne pogreške rezultata rješenja zasnovanog na slici dobivenoj s jedne kamere i relativne pogreške rezultata rješenja zasnovanih na stereo paru slika, uz primjenu FGS filtera s navođenjem te bez njegove primjene, za drugo testno vozilo.

Obojana disparitetna mapa rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na stereo paru slika, uz primjenu FGS filtera s navođenjem te ispravljeni stereo par



Sl. 4.6. Obojana disparitetna mapa rješenja zasnovanog na stereo paru slika, bez primjene FGS filtera te ispravljeni stereo par slika s detektiranim drugim testnim vozilom, na udaljenosti od šest metara.

slika s detektiranim drugim testnim vozilom, koje je šest metara udaljeno od kamera, nalaze se na slici 4.7.



Sl. 4.7. Obojana disparitetna mapa rješenja zasnovanog na stereo paru slika, uz primjenu FGS filtera s navođenjem te ispravljeni stereo par slika s detektiranim drugim testnim vozilom, na udaljenosti od šest metara.

4.2.3. Treće testno vozilo

Procijenjene udaljenosti detektiranog trećeg testnog vozila, prikazane su tablicom 4.3.

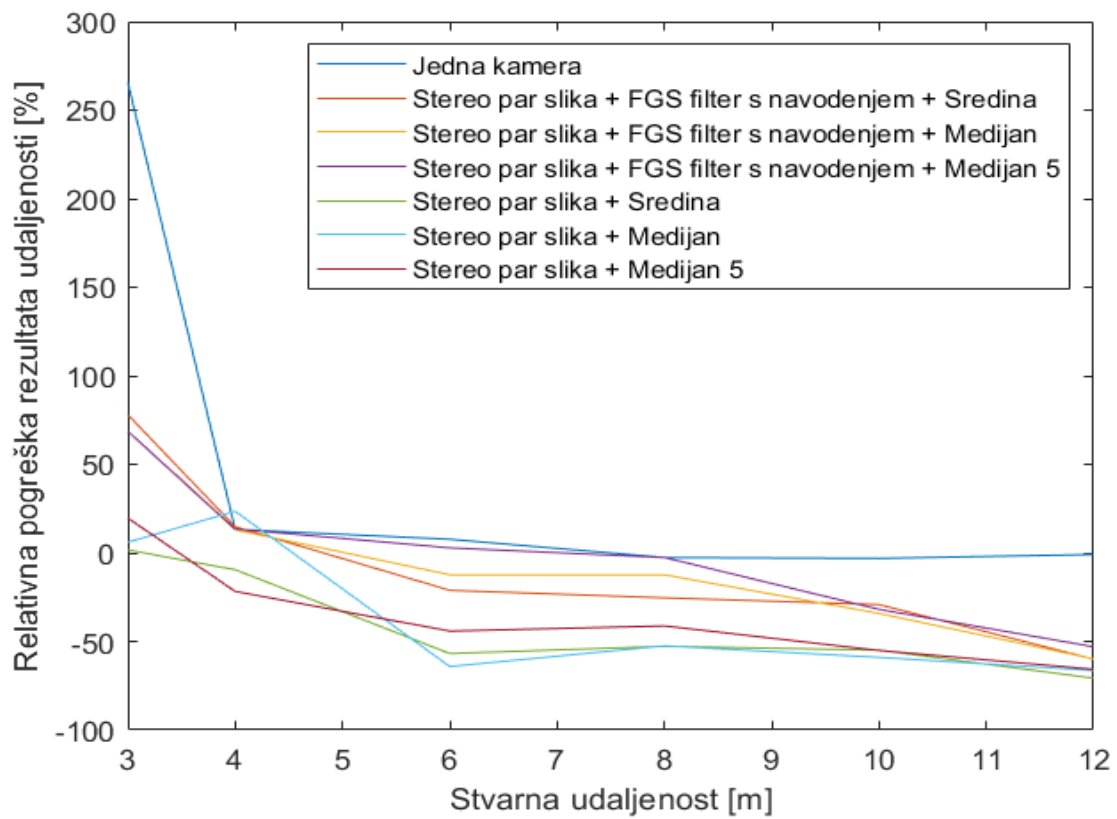
Tablica 4.3. Rezultati udaljenosti detektiranog trećeg testnog vozila.

<i>Stvarna udaljenost</i>	<i>Jedna kamera</i>	<i>Stereo sustav bez filtera</i>			<i>Stereo sustav s FGS filterom uz navođenje</i>		
		<i>Sredina</i>	<i>Medijan</i>	<i>Medijan 5</i>	<i>Sredina</i>	<i>Medijan</i>	<i>Medijan 5</i>
3	10,97	3,05	3,18	3,59	5,34	5,06	5,06
4	4,53	3,62	4,94	3,13	4,59	4,52	4,54
6	6,46	2,59	2,15	3,35	4,73	5,26	6,17
8	7,79	3,78	3,80	4,70	5,97	7,01	7,79
10	9,69	4,51	4,10	4,49	7,08	6,58	6,81
12	11,89	3,51	4,03	4,11	4,80	4,82	5,64

Slika 4.8. prikazuje graf kretanja relativne pogreške rezultata rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na stereo paru slika, uz primjenu FGS filtera s navođenjem, graf kretanja relativne pogreške rezultata rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na stereo paru slika, bez primjene FGS filtera te graf kretanja relativne pogreške rezultata rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na slici dobivenoj s jedne kamere, za treće testno vozilo, na različitim udaljenostima.

Slika 4.9. prikazuje obojano disparitetnu mapu rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na stereo paru slika, bez primjene FGS filtera, te ispravljeni stereo par slika s detektiranim trećim testnim vozilom, koje je šest metara udaljeno od kamere.

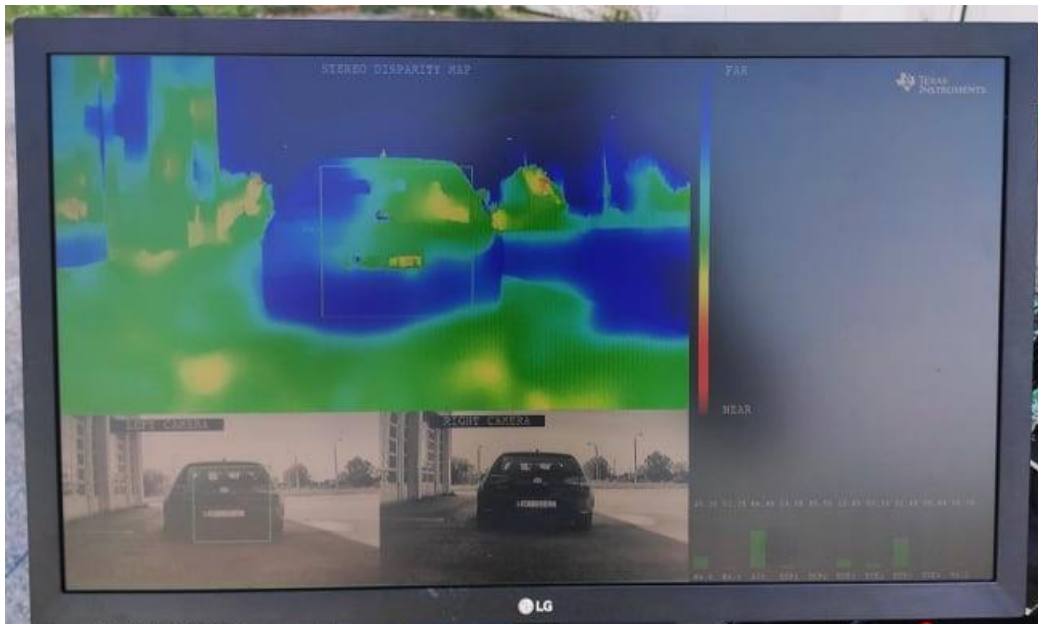
bojana disparitetna mapa rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na stereo paru slika, uz primjenu FGS filtera s navođenjem te ispravljeni stereo par slika s detektiranim trećim testnim vozilom, koje je šest metara udaljeno od kamere, nalaze se na slici 4.10.



Sl. 4.9. Usporedba relativne pogreške rezultata rješenja zasnovanog na slici dobivenoj s jedne kamere i relativne pogreške rezultata rješenja zasnovanih na stereo paru slika, uz primjenu FGS filtera s navodenjem te bez njegove primjene, za treće testno vozilo.



Sl. 4.8. Obojana disparitetna mapa rješenja zasnovanog na stereo paru slika, bez primjene FGS filtera te ispravljeni stereo par slika s detektiranim trećim testnim vozilom, na udaljenosti od šest metara.



Sl. 4.10. Obojana disparitetna mapa rješenja zasnovanog na stereo paru slika, uz primjenu FGS filtera s navođenjem te ispravljeni stereo par slika s detektiranim trećim testnim vozilom, na udaljenosti od šest metara.

4.2.4. Četvrto testno vozilo

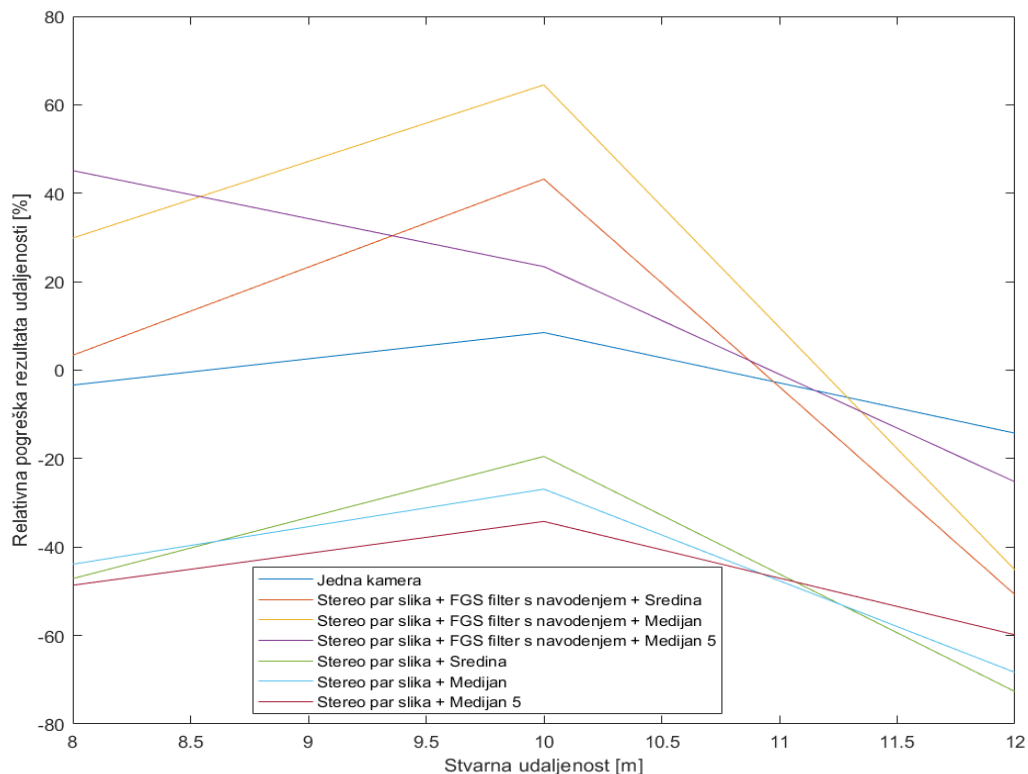
Četvrto testno vozilo poslužilo je za prikaz manjkavosti algoritma za detekciju vozila, koji je implementiran od strane proizvođača Vision SDK programske platforme, a posljedično i nedostataka triju rješenja za određivanje udaljenosti detektiranog vozila, predloženih na početku ovog podpoglavlja. Naime, testno je vozilo bijele boje, a ona predstavlja problem, kako algoritmu za detekciju vozila, tako i algoritmu za izračunavanje disparitetne mape. Budući da je ovo testno vozilo uspješno detektirano tek na udaljenostima od osam metara, pa nadalje, za te su udaljenosti i prezentirani dobiveni rezultati.

Procijenjene udaljenosti detektiranog četvrtog testnog vozila, prikazane su tablicom 4.4.

Tablica 4.4. Rezultati udaljenosti detektiranog trećeg testnog vozila.

Stvarna udaljenost	Jedna kamera	Stereo sustav bez filtera			Stereo sustav s FGS filterom uz navođenje		
		Sredina	Medijan	Medijan 5	Sredina	Medijan	Medijan 5
8	7,73	4,23	4,49	4,11	8,27	10,39	11,61
10	10,85	8,05	7,31	6,58	14,32	16,45	12,34
12	10,29	3,28	3,80	4,82	5,91	6,58	8,97

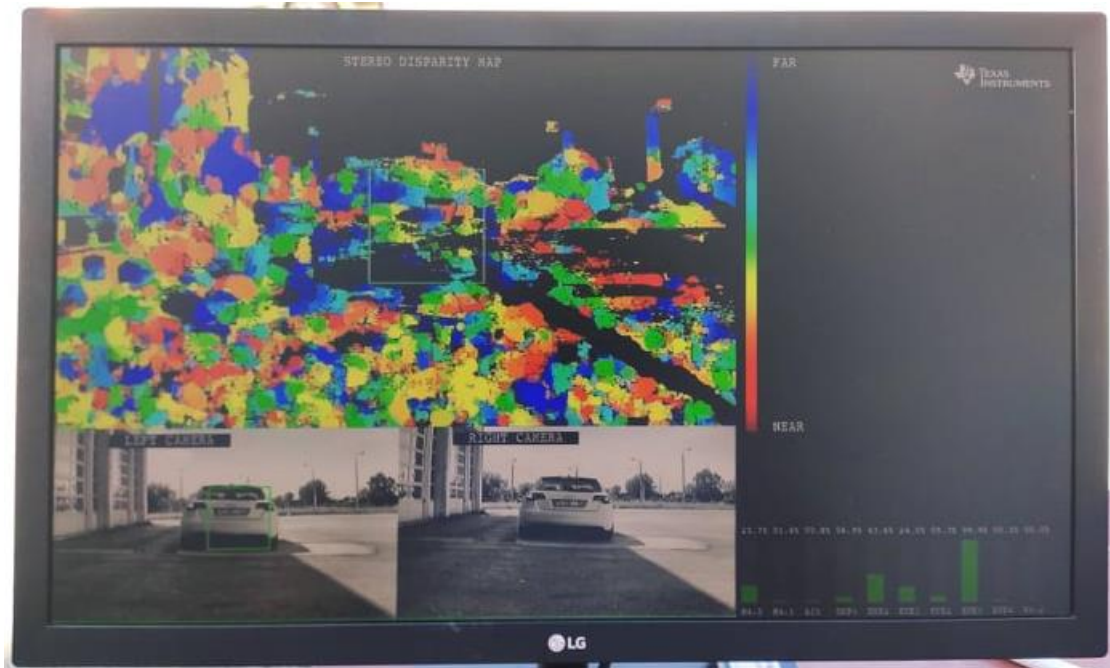
Slika 4.11. prikazuje graf kretanja relativne pogreške rezultata rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na stereo paru slika, uz primjenu FGS filtera s navođenjem, graf kretanja relativne pogreške rezultata rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na stereo paru slika, bez primjene FGS filtera te graf kretanja relativne pogreške rezultata rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na slici dobivenoj s jedne kamere, za četvrto testno vozilo, na različitim udaljenostima.



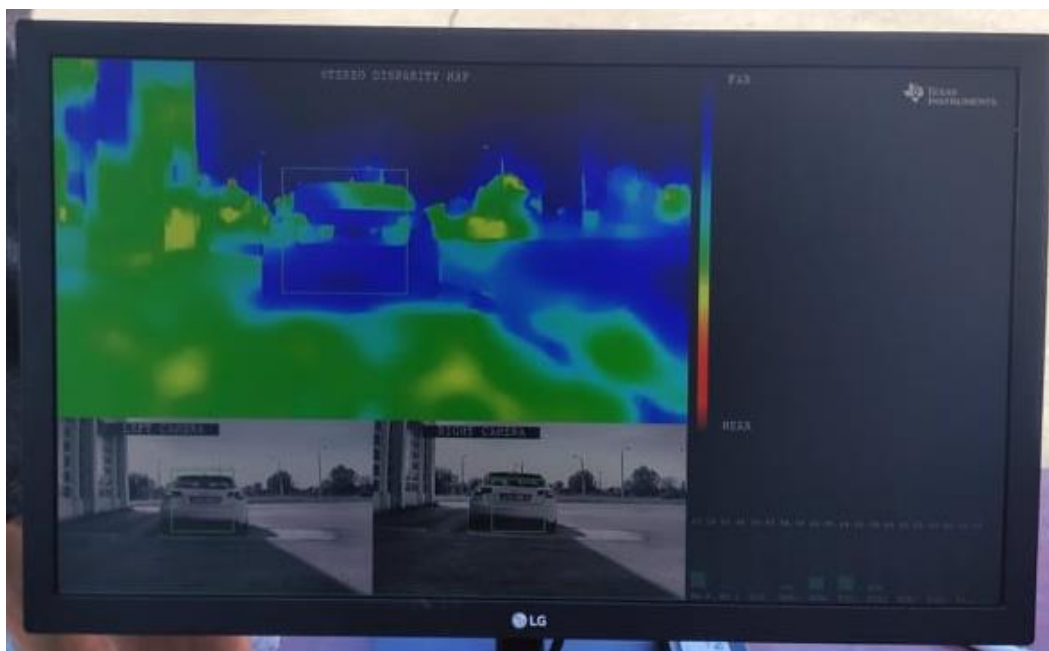
Sl. 4.11. Usporedba relativne pogreške rezultata rješenja zasnovanog na slici dobivenoj s jedne kamere i relativne pogreške rezultata rješenja zasnovanih na stereo paru slika, uz primjenu FGS filtera s navođenjem te bez njegove primjene, za četvrto testno vozilo.

Slika 4.12. prikazuje obojano disparitetnu mapu rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na stereo paru slika, bez primjene FGS filtera, te ispravljeni stereo par slika s detektiranim četvrtim testnim vozilom, koje je osam metara udaljeno od kamera.

Obojana disparitetna mapa rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na stereo paru slika, uz primjenu FGS filtera s navođenjem te ispravljeni stereo par slika s detektiranim četvrtim testnim vozilom, koje je osam metara udaljeno od kamera, nalaze se na slici 4.13.



Sl. 4.13. Obojana disparitetna mapa rješenja zasnovanog na stereo paru slika, bez primjene FGS filtera te ispravljeni stereo par slika s detektiranim četvrtim testnim vozilom, na udaljenosti od osam metara.



Sl. 4.12. Obojana disparitetna mapa rješenja zasnovanog na stereo paru slika, uz primjenu FGS filtera s navođenjem te ispravljeni stereo par slika s detektiranim četvrtim testnim vozilom, na udaljenosti od osam metara.

4.2.5. Srednja relativna pogreška i standardna devijacija procjene udaljenosti za prva tri testna vozila

Srednja relativna pogreška procjene udaljenosti za prva tri testna vozila, prikazana je u tablici 4.5.

Tablica 4.5. Srednja relativna pogreške procjene udaljenosti za prva tri testna vozila.

Stvarna udaljenost	Jedna kamera	Stereo sustav bez filtera			Stereo sustav s FGS filterom uz navođenje		
		Sredina	Medijan	Medijan ₅	Sredina	Medijan	Medijan ₅
3	1,3356	0,1022	0,1400	0,2956	1,1178	1,6033	1,9678
4	1,9917	-0,0833	0,4208	-0,0892	0,2108	0,1750	0,4108
6	0,0594	-0,6106	-0,6539	-0,4928	-0,1383	-0,0306	0,1733
8	-0,0287	-0,5954	-0,5929	-0,5258	-0,1625	-0,0779	-0,0325
10	-0,0430	-0,6647	-0,6847	-0,6670	-0,2620	-0,1687	-0,2590
12	-0,0014	-0,7528	-0,7267	-0,7039	-0,6169	-0,6219	-0,5811

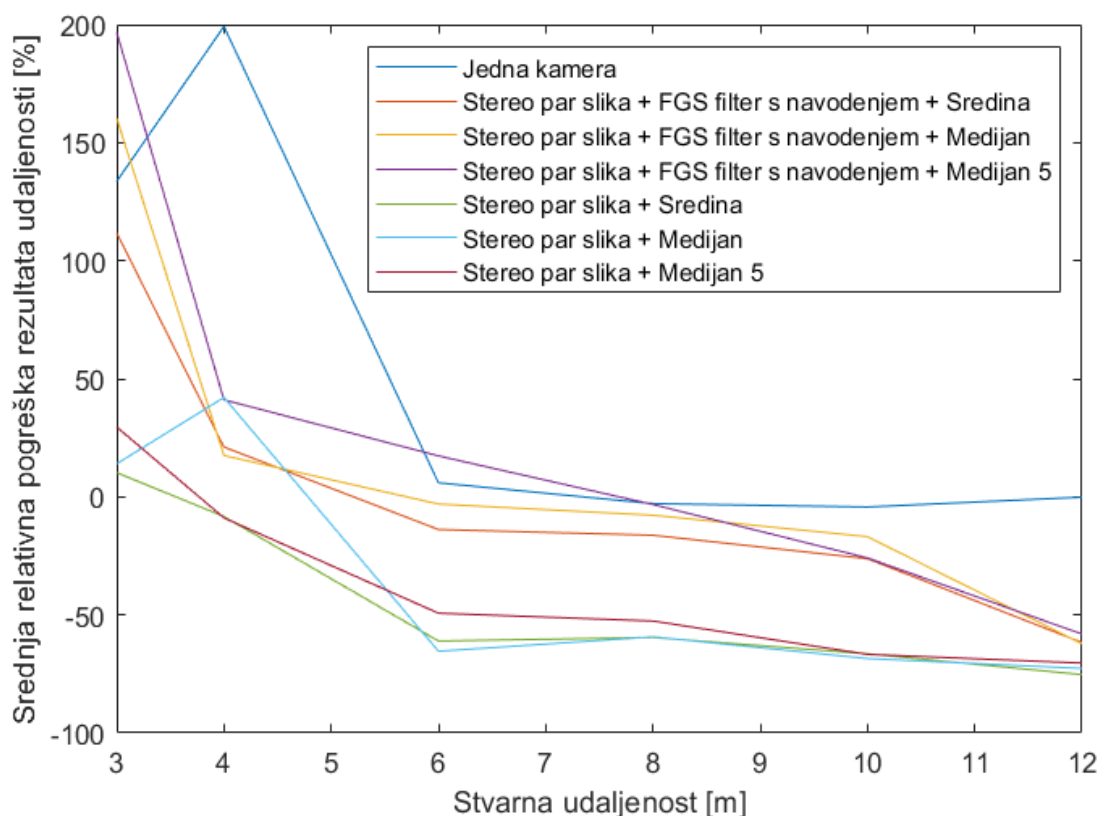
Slika 4.14. prikazuje graf kretanje srednje relativne pogreške rezultata rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na stereo paru slika, uz primjenu FGS filtera s navođenjem, graf kretanja srednje relativne pogreške rezultata rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na stereo paru slika, bez primjene FGS filtera te graf kretanja srednje relativne pogreške rezultata rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na slici dobivenoj s jedne kamere, za prva tri testna vozila, na različitim udaljenostima.

Standardna devijacija (engl. *standard deviation*) rezultata testiranih rješenja računa se prema jednadžbi (4.2.) [33].

$$s = \sqrt{\frac{\sum(\hat{x} - \bar{x})^2}{n - 1}} \quad (4.2.)$$

Standardna devijacija predstavljena je oznakom s , dok \bar{x} označava srednju procijenjenu vrijednost udaljenosti detektiranog vozila, za određeni skup podataka veličine n .

Tablica 4.6. prikazuje standardne devijacije rezultata testiranih rješenja, uzimajući u obzir prva tri testna vozila pozicionirana na svim testnim udaljenostima.



Sl. 4.14 Usporedba relativne pogreške rezultata rješenja zasnovanog na slici dobivenoj s jedne kamere i relativne pogreške rezultata rješenja zasnovanih na stereo paru slika, uz primjenu FGS filtera s navođenjem te bez njegove primjene, za prva tri testna vozila.

Tablica 4.6. Standardne devijacije rezultata testiranih rješenja.

	Jedna kamera	Stereo sustav bez filtera			Stereo sustav s FGS filterom uz navođenje		
		Sredina	Medijan	Medijan 5	Sredina	Medijan	Medijan 5
Standardna devijacija	3,7810	0,7004	1,3222	0,9555	1,7651	2,8071	2,7336
		1,0207			2,4773		

Iz dobivenih vrijednosti srednje relativne pogreške rezultata rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na korištenju slike dobivene s jedne kamere, za prva tri testna vozila, prezentiranih u tablici 4.5., jasno se vidi kako je pogreška znatno veća za manje vrijednosti udaljenosti vozila od kamere, odnosno za udaljenosti od 3 i 4 metra. Kako uspješnost procjene udaljenosti pomoću ovog rješenja direktno ovisi o visini graničnog okvira, velika vrijednost pogreške procjene udaljenosti za navedene udaljenosti vozila od kamere, posljedica je manjih visina graničnih okvira od potrebnih za potpuno obuhvaćanje detektiranih vozila. Za

udaljenosti detektiranih vozila od kamere iznosa od 6 do 12 metara, vrijednosti srednje relativne pogreške procjene udaljenosti ustaljuju se, te su znatno manjeg iznosa od dobivenih vrijednosti za udaljenosti od 3 i 4 metra. To je također posljedica visina graničnih okvira, koje su za navedene udaljenosti mnogo bliže vrijednostima potrebnim za potpuno i dostatno obuhvaćanje detektiranih vozila.

Srednja relativna pogreška rezultata rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na korištenju stereo para slika, bez primjene FGS filtera, manja je za manje vrijednosti udaljenosti vozila od kamera, odnosno za udaljenosti od 3 i 4 metra. Za udaljenosti vozila od kamera iznosa 6 metara, i veće, srednja relativna pogreška procjene udaljenosti gotovo konstantno se povećava, ali u negativnom smjeru (procijenjene udaljenosti manje su od stvarnih). Vrijednosti procijenjene udaljenosti vozila, za sve testne udaljenosti te sve tri metode izračuna korištene u ovom rješenju, imaju standardnu devijaciju iznosa 1,0207. Iz toga se može zaključiti da procijenjene udaljenosti detektiranih vozila ne odstupaju mnogo za različite testne udaljenosti. Tomu je mogući razlog postojanje više malih površina s ujednačenim disparitetima, koje se odnose na detektirano vozilo. Upravo se zbog toga pribjeglo upotrebi FGS filtra s navođenjem.

Rješenje za određivanje udaljenosti detektiranog vozila zasnovano na korištenju stereo para slika, uz primjenu FGS filtera s navođenjem, daje rezultate s najvećom srednjom relativnom pogreškom, za udaljenost detektiranog vozila od kamera iznosa 3 metra. Potom, pogreška znatno opada za vrijednosti udaljenosti od 4 do 10 metara. Za udaljenosti veće od 10 metara, iznos srednje relativne pogreške ponovno se povećava, ali u negativnom smjeru. Mogući razlog velike vrijednosti srednje relativne pogreške za udaljenost detektiranog vozila iznosa 3 metra, nemogućnost je algoritma za detekciju vozila da iscrta dovoljno veliki granični okvir, koji bi obuhvaćao cijelu površinu vozila. Povećanje iznosa srednje relativne pogreške, ali u negativnom smjeru, za udaljenosti vozila veće od 10 metara, vjerojatno je posljedica uklapanja male površine, koja predstavlja vozilo u mapi dispariteta, u veću površinu koja predstavlja okolinu vozila te niske rezolucije ispravljenog stereo para slika.

4.2.6. Vremena i frekvencije izvođenja predloženih rješenja

Vremena i frekvencije izvođenja triju rješenja predloženih na početku ovog podpoglavlja, a pokrenutih na ADAS ALPHA ploči, prezentirana su u tablici 4.7.

Najmanje je vremena potrebno za izvođenje rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na korištenju slika dobivenih s jedne kamere. Potom, veća su vremena izvođenja metoda upotrijebljenih u rješenju zasnovanom na korištenju stereo para slika,

Tablica 4.7. Vremena i frekvencije izvođenja triju predloženih rješenja.

	<i>Jedna kamera</i>	<i>Stereo sustav bez filtera</i>			<i>Stereo sustav s FGS filterom uz navođenje</i>		
		<i>Sredina</i>	<i>Medijan</i>	<i>Medijan 5</i>	<i>Sredina</i>	<i>Medijan</i>	<i>Medijan 5</i>
<i>Vrijeme izvođenja [ms]</i>	38	42	59	51	123	147	128
<i>Frekvencija izvođenja [fps]</i>	26,32	23,81	16,95	19,61	8,13	6,80	7,81

ali bez primjene FGS filtera. Konačno, najviše vremena za izvođenje iziskuju metode upotrjebljene u rješenju zasnovanom na korištenju stereo para slika, uz primjenu FGS filtera s navođenjem. Rješenje zasnovano na korištenju slika dobivenih s jedne kamere najmanje računski zahtjevno. Također, rješenje zasnovano na korištenju stereo para slika, uz primjenu FGS filtera, nadogradnja je rješenja zasnovanog na korištenju stereo para slika, bez primjene FGS filtera pa su dobivene vrijednosti vremena izvođenja logične.

4.3. Osvrt na dobivene rezultate

Pomoću rezultata testiranja prezentiranih u ovom poglavlju moguće je uvidjeti da, uz iznimku rezultata za testne udaljenosti detektiranog vozila od kamere iznosa tri i četiri metra, rješenje za određivanje udaljenosti zasnovano na korištenju slike dobivene s jedne kamere daje rezultate s najmanjom relativnom pogreškom. Također, jasno je vidljivo da rješenje zasnovano na korištenju stereo para slika, uz primjenu FGS filtera s navođenjem, rezultira vrijednostima s manjom relativnom pogreškom od vrijednosti dobivenih rješenjem zasnovanim na korištenju stereo para slika, ali bez primjene FGS filtera. Iz navedenog se može zaključiti da primjena FGS filtera s navođenjem ima smisla. No, rezultati takvog rješenja i dalje daju veću relativnu pogrešku od rješenja zasnovanog na korištenju slike dobivene s jedne kamere. Uz to, kako je prezentirano u dijelu 4.2.4., rezultati svih triju rješenja direktno ovise o mogućnostima algoritma za detekciju vozila. Dodatne manjkavosti predloženog konačnog rješenja uočene su prilikom procjene udaljenosti osvijetljenog, odnosno osunčanog vozila te vozila koje se nalazi u okruženju s velikom količinom detalja, poput grmlja i krošnji drveća. Iako je premisa rada bila da će rješenje za određivanje udaljenosti detektiranog vozila zasnovano na korištenju stereo para slika dati bolje rezultate od rješenja zasnovanog na slici dobivenoj s jedne kamere, to ovim radom nije dokazano.

Glavni je razlog tomu što je preduvjet za ispravniji rad stereo sustava, koji u slučaju razvoja rješenja zadatka ovog diplomskog rada nije ispunjen, kvalitetnija oprema od korištene, kako u vidu kamera koje daju video okvire veće rezolucije, tako i ploče s većom procesorskom moći te kvalitetnije konstrukcije na kojoj su kamere pričvršćene. Moguća poboljšanja navedenih nedostataka rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na stereo paru slika, uz primjenu FGS filtera s navođenjem, mogu se provesti u vidu poboljšanja algoritma za detekciju vozila, kako bi se povećala točnost (engl. *accuracy*) te kako bi granični okvir ocrtao vozilo što bliže njegovoj stvarnoj veličini. Moguće je poboljšati kalibracijski algoritam, kako bi se što veći broj odgovarajućih elemenata slike lijeve i desne kamere nalazio na istoj epipolarnoj liniji te posljedično algoritam za ispravljanje stereo para slika, koji bi davao ispravljeni stereo par slika veće rezolucije. Uz navedeno, moguće je provesti i optimizaciju vremena izvođenja, kako bi se rješenje prilagodilo za rad u stvarnom vremenu.

5. ZAKLJUČAK

Rezultati testiranja predloženog konačnog rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na korištenju stereo para slika, uz upotrebu FGS filtera s navođenjem, uspoređeni su s rezultatima rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na korištenju slike dobivene s jedne kamere te rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na korištenju stereo para slika, bez upotrebe FGS filtera. Testiranje je provedeno pomoću sustava za testiranje, čija je centralna komponenta ADAS ALPHA ploča, na koju je priključen ranije kalibrirani stereo sustav kamera. Upotrijebljena su četiri različita vozila, za koja se procjenjivala udaljenost na šest testnih pozicija, koje su se nalazile 3, 4, 6, 8, 10 i 12 metara udaljene od kamera. Također, izmjerena su vremena izvođenja pojedinih rješenja.

Najmanje je vremena potrebno za izvođenje rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na korištenju slika dobivenih s jedne kamere. Potom, veća su vremena izvođenja metoda upotrijebljenih u rješenju zasnovanom na korištenju stereo para slika, ali bez primjene FGS filtera. Konačno, najviše vremena za izvođenje iziskuju metode upotrijebljene u rješenju zasnovanom na korištenju stereo para slika, uz primjenu FGS filtera s navođenjem.

Pomoću rezultata testiranja moguće je uvidjeti da, uz iznimku rezultata za testne udaljenosti detektiranog vozila od kamere iznosa tri i četiri metra, rješenje za određivanje udaljenosti zasnovano na korištenju slike dobivene s jedne kamere daje rezultate s najmanjom relativnom pogreškom. Također, jasno je vidljivo da rješenje zasnovano na korištenju stereo para slika, uz primjenu FGS filtera s navođenjem, rezultira vrijednostima s manjom relativnom pogreškom od vrijednosti dobivenih rješenjem zasnovanim na korištenju stereo para slika, ali bez primjene FGS filtera. Iz navedenog se može zaključiti da primjena FGS filtera s navođenjem ima smisla. Međutim, iako je premisa rada bila da će rješenje za određivanje udaljenosti detektiranog vozila zasnovano na korištenju stereo para slika dati bolje rezultate od rješenja zasnovanog na slici dobivenoj s jedne kamere, to ovim radom nije dokazano. Glavni je razlog tomu što je preduvjet za ispravniji rad stereo sustava, koji u slučaju razvoja rješenja zadatka ovog diplomskog rada nije ispunjen, kvalitetnija oprema od korištene, kako u vidu kamera koje daju video okvire veće rezolucije, tako i ploče s većom procesorskom moći te kvalitetnije konstrukcije na kojoj su kamere pričvršćene.

Moguća poboljšanja navedenih nedostataka rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na stereo paru slika, uz primjenu FGS filtera s navođenjem, mogu

se provesti u vidu poboljšanja algoritma za detekciju vozila, kalibracijskog algoritma, algoritma za ispravljanje stereo para slika te optimizacije vremena izvođenja.

LITERATURA

- [1] M. Vranješ, „Digitalna obrada slike i videa za autonomna vozila, Laboratorijske vježbe, Vježba 1, Upoznavanje s ADAS razvojnom pločom“. Fakultet elektrotehnike, računarstva i informacijskih tehnologija, Osijek, Pristupljeno: kol. 21, 2020. [Na internetu].
- [2] „J3016: Taxonomy and Definitions for Terms Related to On-Road Motor Vehicle Automated Driving Systems - SAE International“. https://www.sae.org/standards/content/j3016_201401/ (pristupljeno ruj. 05, 2020).
- [3] „Three Sensor Types Drive Autonomous Vehicles“, *FierceElectronics*. <https://www.fierceelectronics.com/components/three-sensor-types-drive-autonomous-vehicles> (pristupljeno kol. 21, 2020).
- [4] „The 6 Levels of Vehicle Autonomy Explained | Synopsys Automotive“. <https://www.synopsys.com/automotive/autonomous-driving-levels.html> (pristupljeno kol. 21, 2020).
- [5] S. Rimac-Drlje i M. Vranješ, „Digitalna obrada slike i videa za autonomna vozila, Predavanja, 1. Uvod“. Fakultet elektrotehnike, računarstva i informacijskih tehnologija, Osijek, Pristupljeno: kol. 21, 2020. [Na internetu].
- [6] „What Is a CMOS Image Sensor? | The principle of Semiconductor“, *nanotec museum*. <http://www.tel.com/museum/exhibition/principle/> (pristupljeno kol. 24, 2020).
- [7] P. Alizadeh, „Object Distance Measurement Using a Single Camera for Robotic Applications“, str. 126.
- [8] J. Brownlee, „A Gentle Introduction to Object Recognition With Deep Learning“, *Machine Learning Mastery*, svi. 21, 2019. <https://machinelearningmastery.com/object-recognition-with-deep-learning/> (pristupljeno kol. 25, 2020).
- [9] „Object recognition definition and use cases“, *Deepomatic*, srp. 11, 2018. <https://deepomatic.com/what-is-object-recognition-and-how-you-can-use-it> (pristupljeno kol. 25, 2020).
- [10] R. Cupec, „Računalna geometrija i robotski vid, Predavanja, 6. Trodimenzionalna rekonstrukcija scene na temelju više slika“. Fakultet elektrotehnike, računarstva i informacijskih tehnologija, Osijek, stu. 22, 2010, Pristupljeno: kol. 27, 2020. [Na internetu].
- [11] Z.-N. Li, M. S. Drew, i J. Liu, *Fundamentals od Multimedia*, Second Edition. Switzerland: Springer International Publishing, 2014.
- [12] „Stereo Vision Tutorial - Part I · Chris McCormick“. <http://mccormickml.com/2014/01/10/stereo-vision-tutorial-part-i/> (pristupljeno kol. 27, 2020).
- [13] A. Ali, A. Hassan, A. R. Ali, H. Ullah Khan, W. Kazmi, i A. Zaheer, „Real-time vehicle distance estimation using single view geometry“, u *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, Snowmass Village, CO, USA, ožu. 2020, str. 1100–1109, doi: 10.1109/WACV45572.2020.9093634.
- [14] „The KITTI Vision Benchmark Suite“. <http://www.cvlibs.net/datasets/kitti/index.php> (pristupljeno ruj. 06, 2020).
- [15] „nuScenes dataset“. <https://www.nuscenes.org/> (pristupljeno ruj. 06, 2020).
- [16] „Level 5 - Lyft“. <https://self-driving.lyft.com/level5/> (pristupljeno ruj. 06, 2020).
- [17] L. Huang, T. Zhe, J. Wu, Q. Wu, C. Pei, i D. Chen, „Robust Inter-Vehicle Distance Estimation Method Based on Monocular Vision“, *IEEE Access*, sv. 7, str. 46059–46070, 2019, doi: 10.1109/ACCESS.2019.2907984.
- [18] J. Lim, J. Park, H. Park, H. Kim, i J. Paik, „Principal depth estimation using cost-volume filtering for fast stereo matching“, u *2018 International Conference on Electronics, Information, and Communication (ICEIC)*, Honolulu, HI, sij. 2018, str. 1–2, doi: 10.23919/ELINFOCOM.2018.8330635.

- [19] „vision.middlebury.edu/stereo/data“. <https://vision.middlebury.edu/stereo/data/> (pristupljeno ruj. 06, 2020).
- [20] Y. Xu, S. Mita, H. Tehrani, H. Chin, K. Ishimaru, i S. Nishino, „Real-time Stereo Disparity Quality Improvement for Challenging Traffic Environments“, u *2018 IEEE Intelligent Vehicles Symposium (IV)*, Changshu, lip. 2018, str. 1749–1754, doi: 10.1109/IVS.2018.8500538.
- [21] S. Joung, S. Kim, K. Park, i K. Sohn, „Unsupervised Stereo Matching Using Confidential Correspondence Consistency“, *IEEE Trans. Intell. Transp. Syst.*, sv. 21, izd. 5, str. 2190–2203, svi. 2020, doi: 10.1109/TITS.2019.2917538.
- [22] „HCI Tagging Database - Home“. <https://mahnob-db.eu/hci-tagging/> (pristupljeno ruj. 06, 2020).
- [23] „Digital Image Media Lab“. <https://diml.yonsei.ac.kr/dataset/> (pristupljeno ruj. 06, 2020).
- [24] „VISION SDK Software Architecture“. Texas Instruments, kol. 24, 2015, Pristupljeno: ruj. 01, 2020. [Na internetu].
- [25] D. Min, S. Choi, J. Lu, B. Ham, K. Sohn, i M. N. Do, „Fast Global Image Smoothing Based on Weighted Least Squares“, *IEEE Trans. Image Process.*, sv. 23, izd. 12, str. 5638–5653, pros. 2014, doi: 10.1109/TIP.2014.2366600.
- [26] „Calibration Best Practices“, *calib.io*. <https://calib.io/blogs/knowledge-base/calibration-best-practices> (pristupljeno ruj. 03, 2020).
- [27] „VisionSDK_MultiSensorFusionStereoCalibrationGuide Vision SDK User Guide TDA2xx Stereo Calibration“, *UserManual.wiki*. <https://usermanual.wiki/Document/VisionSDKUserGuideTDA2xxStereoCalibrationGuide.869762026/view> (pristupljeno ruj. 03, 2020).
- [28] A. D. Jepson i D. J. Fleet, „Image Pyramids“. University of Toronto, 2005, Pristupljeno: ruj. 03, 2020. [Na internetu]. Dostupno na: <http://www.cs.toronto.edu/~jepson/csc320/notes/pyramids.pdf>.
- [29] D. Hafner, O. Demetz, i J. Weickert, „Why Is the Census Transform Good for Robust Optic Flow Computation?“, u *Scale Space and Variational Methods in Computer Vision*, Berlin, Heidelberg, 2013, str. 210–221, doi: 10.1007/978-3-642-38267-3_18.
- [30] „Car dimensions of all makes with size comparison tools“. <https://www.automobiledimension.com/> (pristupljeno ruj. 12, 2020).
- [31] „soundsilence/ImageSmoothing“, *GitHub*. <https://github.com/soundsilence/ImageSmoothing> (pristupljeno ruj. 07, 2020).
- [32] E. W. Weisstein, „Relative Error“. <https://mathworld.wolfram.com/RelativeError.html> (pristupljeno ruj. 16, 2020).
- [33] „Standard Deviation: Simple Definition, Step by Step Video“, *Statistics How To*. <https://www.statisticshowto.com/probability-and-statistics/standard-deviation/> (pristupljeno ruj. 20, 2020).

SAŽETAK

Cilj ovog diplomskog rada bio je razviti rješenje za određivanje udaljenosti detektiranog vozila zasnovano na stereo paru slika te ga implementirati na realnu ADAS platformu. Kompletno programsko rješenje zadatka ovog diplomskog rada izvedeno je u Vision SDK programskoj platformi te je implementirano na ADAS ALPHA razvojnoj ploči. Predloženo konačno rješenje zadatka diplomskog rada temelji se na primjeni FGS filtera s navođenjem nad izračunatom mapom dispariteta, kako bi se zagladile površine koje predstavljaju određene objekte snimljene 3D scene, a ujedno sačuvali rubovi tih objekata. Rezultati testiranja konačnog rješenja uspoređeni su s rezultatima rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na korištenju slike dobivene s jedne kamere te rješenja za određivanje udaljenosti detektiranog vozila zasnovanog na korištenju stereo para slika, bez upotrebe FGS filtera. Za testiranje upotrijebljena su četiri različita vozila, za koja se procjenjivala udaljenost na šest testnih pozicija. Pomoću rezultata testiranja moguće je uvidjeti da rješenje zasnovano na korištenju slike dobivene s jedne kamere daje rezultate s najmanjom relativnom pogreškom, te se najbrže izvodi. Također, jasno je vidljivo da rješenje zasnovano na korištenju stereo para slika, uz primjenu FGS filtera s navođenjem, rezultira vrijednostima s manjom relativnom pogreškom od vrijednosti dobivenih rješenjem zasnovanim na korištenju stereo para slika, ali bez primjene FGS filtera. Iz navedenog se može zaključiti da primjena FGS filtera s navođenjem ima smisla. Moguća poboljšanja nedostataka predloženog konačnog rješenja, poput bržeg izvođenja i manje pogreške procjene udaljenosti, mogu se provesti u vidu poboljšanja algoritma za detekciju vozila, kalibracijskog algoritma, algoritma za ispravljanje stereo para slika te optimizacije vremena izvođenja.

Ključne riječi: ADAS, FGS filter, stereo sustav kamera, Vision SDK

DETERMINING THE DISTANCE OF A DETECTED VEHICLE IN FRONT OF A VEHICLE WITH A BUILT-IN STEREO CAMERA SYSTEM

ABSTRACT

The aim of this master thesis was to develop a solution for determining the distance of the detected vehicle based on a stereo pair of images and implement it on a realistic ADAS platform. The complete software solution for this thesis was performed in the Vision SDK software platform and was implemented on the ADAS ALPHA development board. The proposed final solution to the thesis's task is based on the application of FGS filter with guidance over the calculated disparity map, in order to smooth the surfaces representing certain objects of the captured 3D scene, while preserving the edges of the objects. The test results of the final solution are compared with the results of the solution for determining the distance of the detected vehicle based on the usage of images obtained from one camera and with the results of the solution for determining the distance of the detected vehicle based on the usage of stereo image pairs, without FGS filter. Four different vehicles were used for testing, for which the distance was estimated at six test positions. Using the test results, it is possible to see that a solution based on the usage of an image obtained from a single camera gives results with the least relative error and is performed the fastest. Also, it is clearly seen that a solution based on the use of stereo image pairs, with the application of FGS guided filter, results in values with less relative error than the values obtained by a solution based on the usage of stereo image pairs, but without the application of FGS filter. From the above, it can be concluded that the usage of FGS filter with guidance makes sense. Possible improvements to the shortcomings of the proposed final solution, such as faster execution and less distance estimation error, can be implemented in the form of improvements to the vehicle detection algorithm, the calibration algorithm, the algorithm for correcting stereo image pairs and the optimization of the execution time.

Key words: ADAS, FGS filter, stereo camera system, Vision SDK

ŽIVOTOPIS

Marko Miljković rođen je 17. lipnja 1995. godine u Vinkovcima. U Vinkovcima završava osnovnu školu „Antun Gustav Matoš“, te 2009. godine upisuje Prirodoslovno – matematičku gimnaziju „Matija Antun Reljković“ u Vinkovcima. Srednju je školu završio 2013. godine. Godine 2015. ostvaruje upis na Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek, preddiplomski sveučilišni studij, smjer „Računarstvo“. Diplomski sveučilišni studij pri istom fakultetu, upisuje 2018. godine, smjer „Robotika i umjetna inteligencija“.

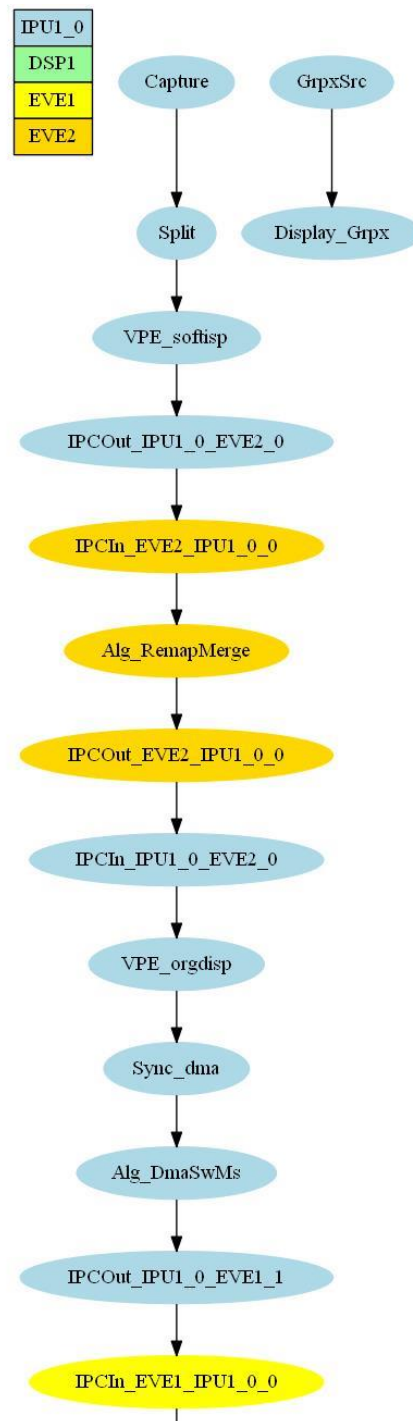
Potpis autora

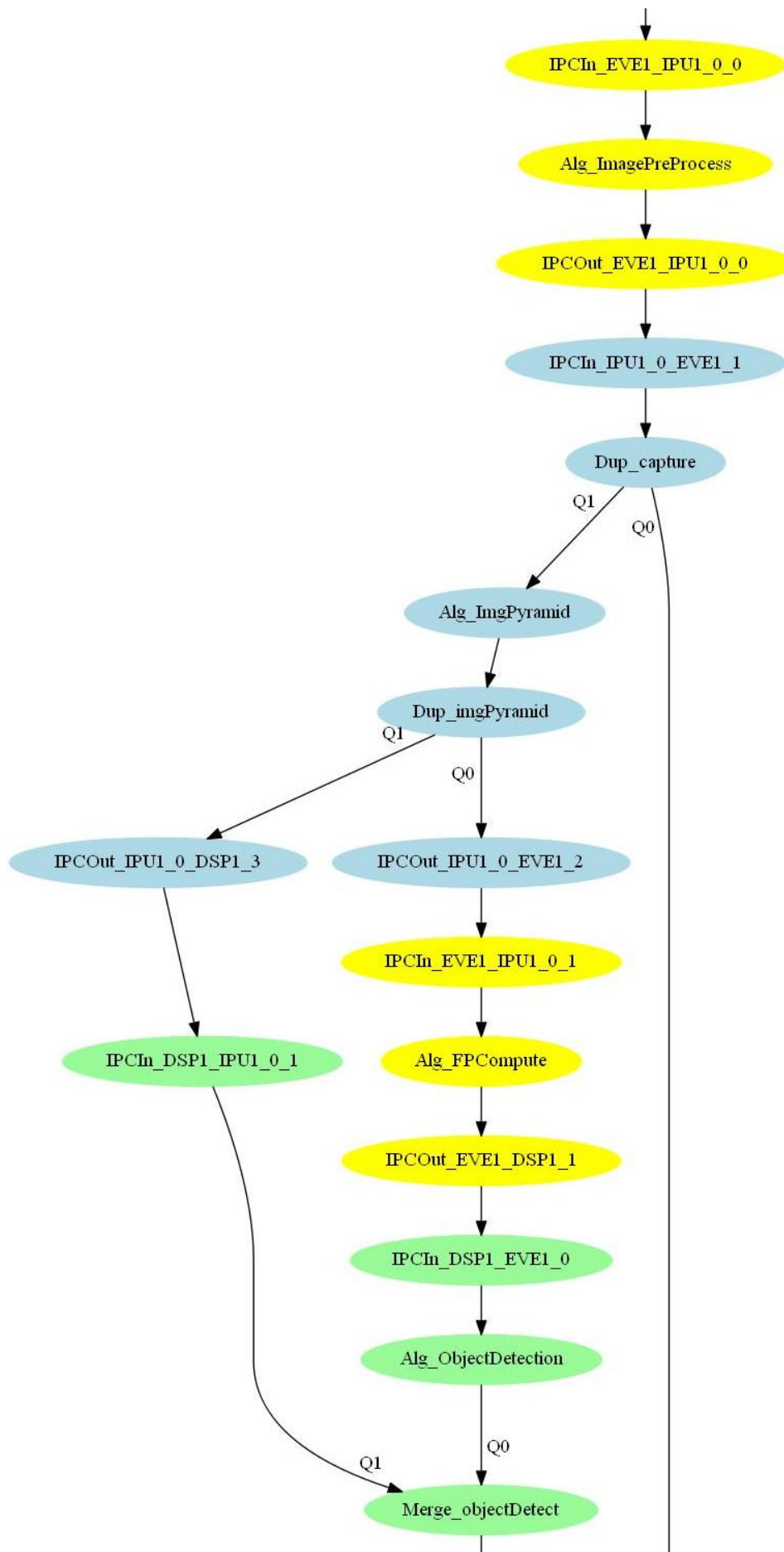
PRILOZI

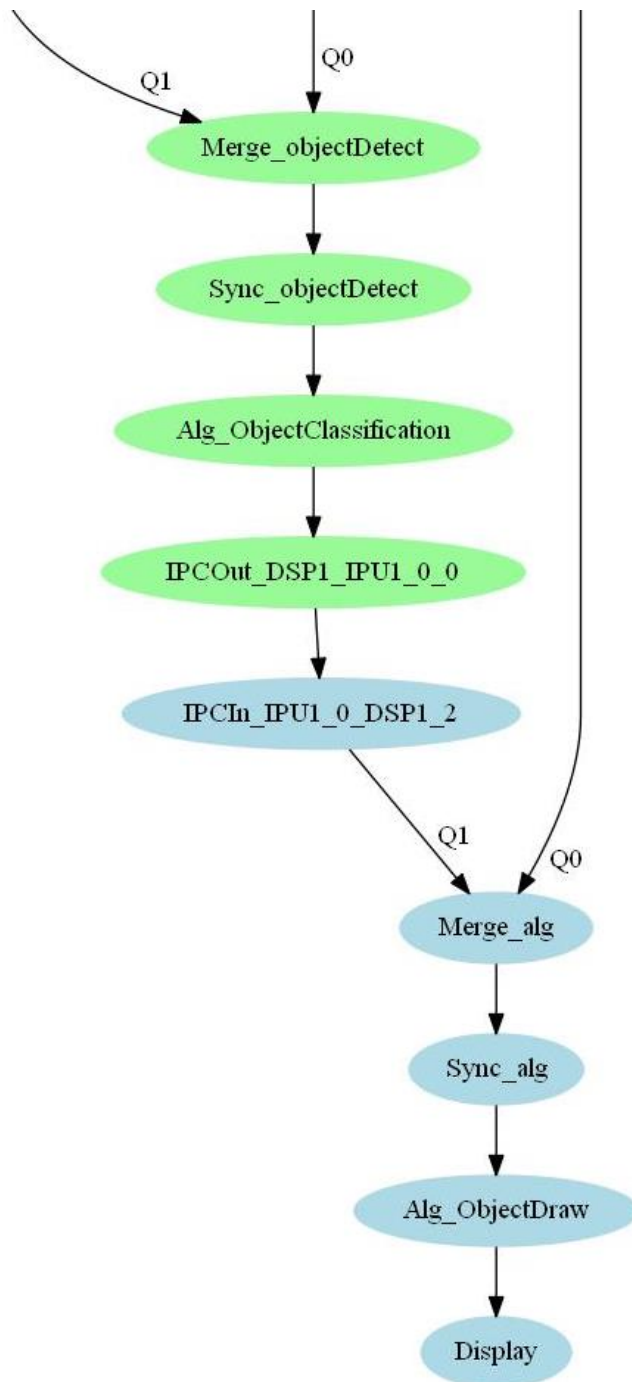
P.3.1. – Mapa s ispisima kodova samostalno razvijenih algoritamskih *link*-ova (nalazi se u elektroničkoj verziji).

P.3.2. – Mapa sa slikama upotrijebljenim kao ulaz u kalibracijski algoritam (nalazi se u elektroničkoj verziji).

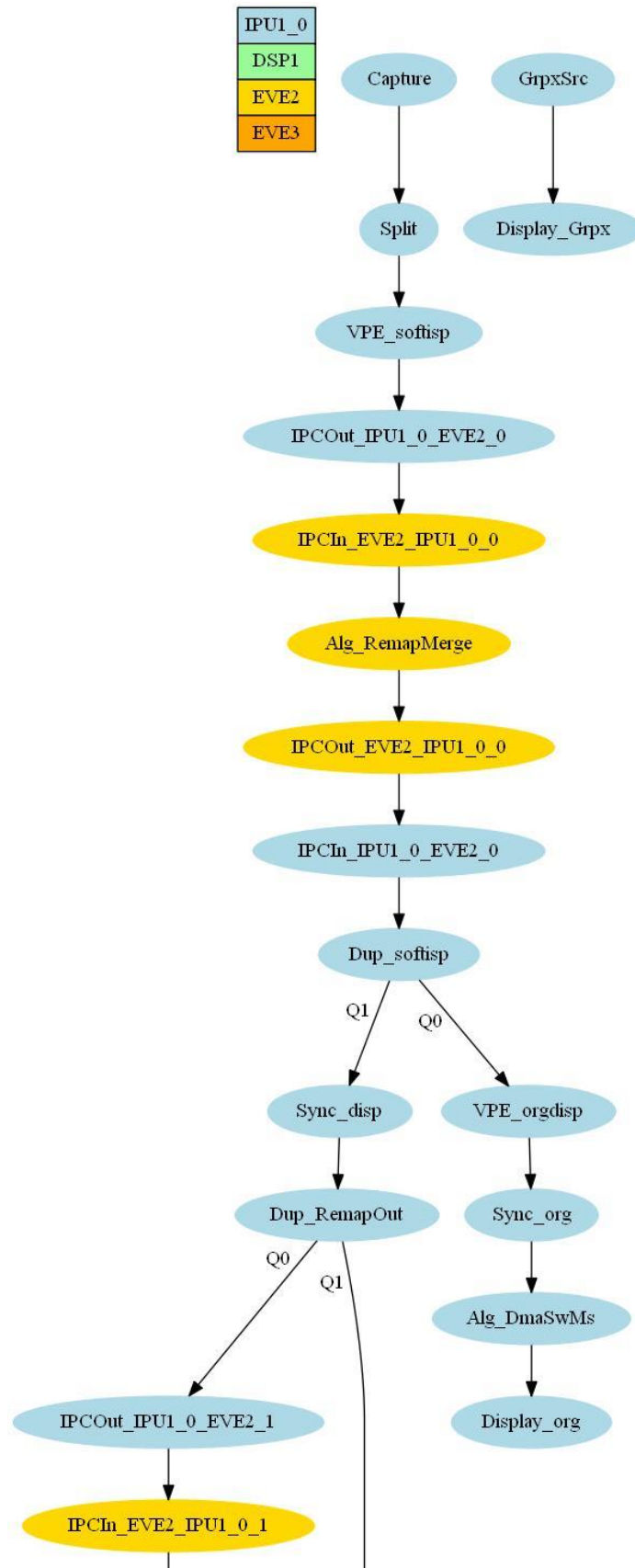
P.3.3. Dijagram toka slučaja upotrebe za detekciju vozila na ispravljenom stereo paru slika.

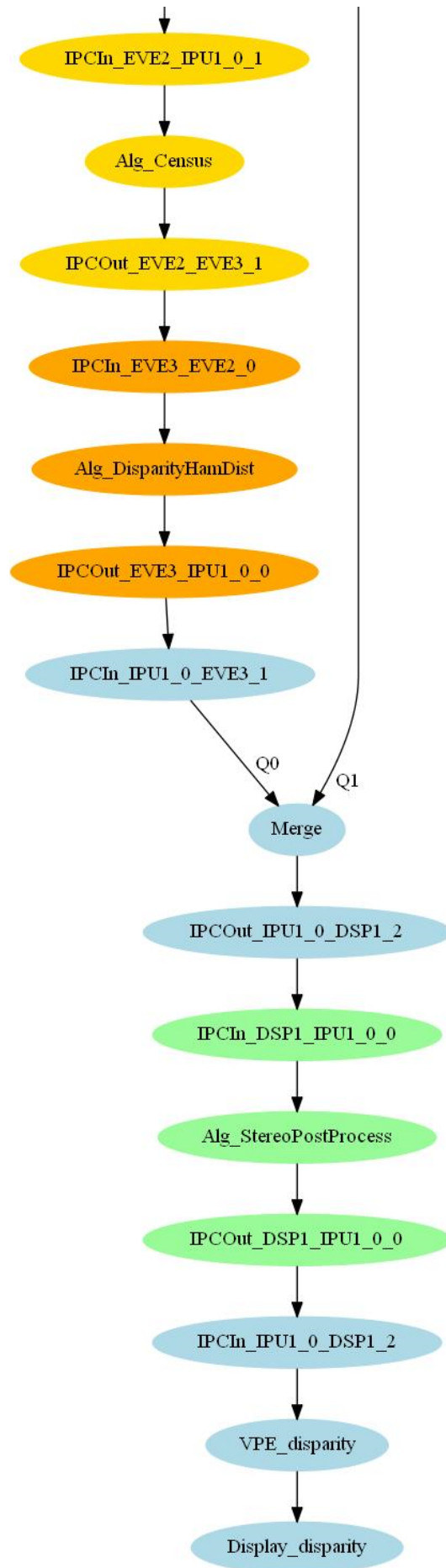




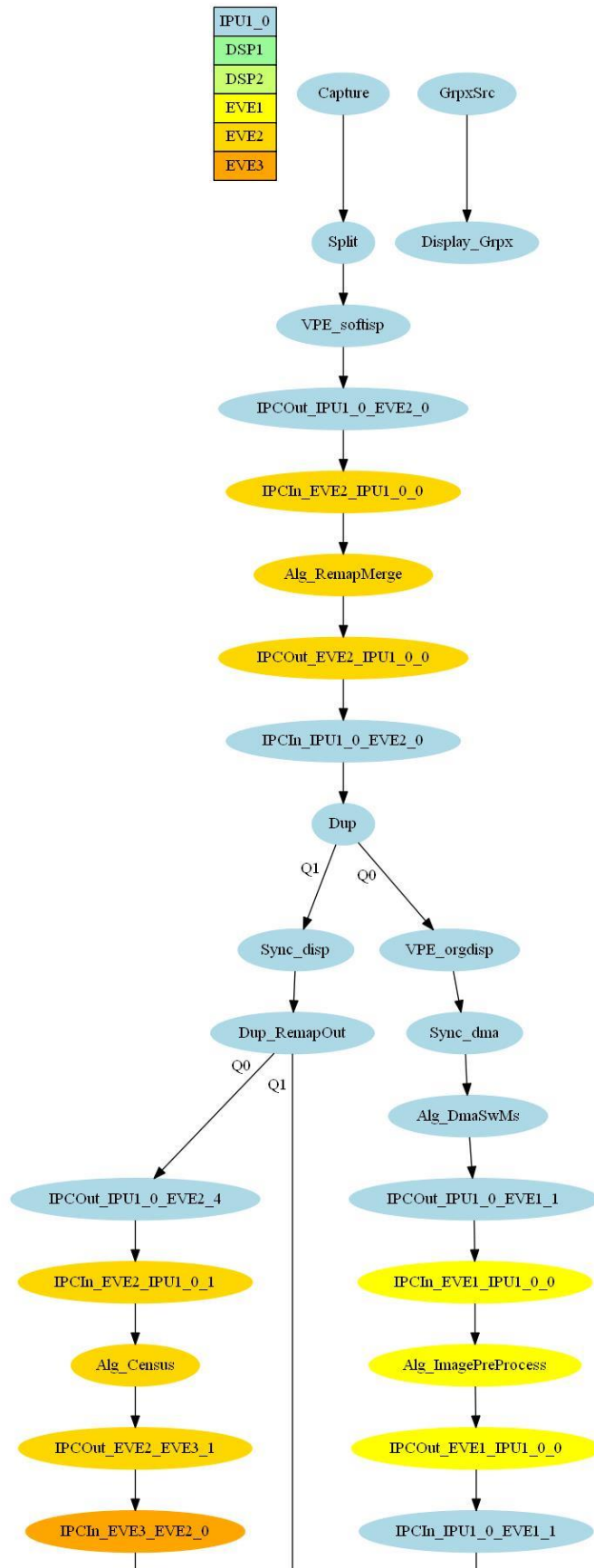


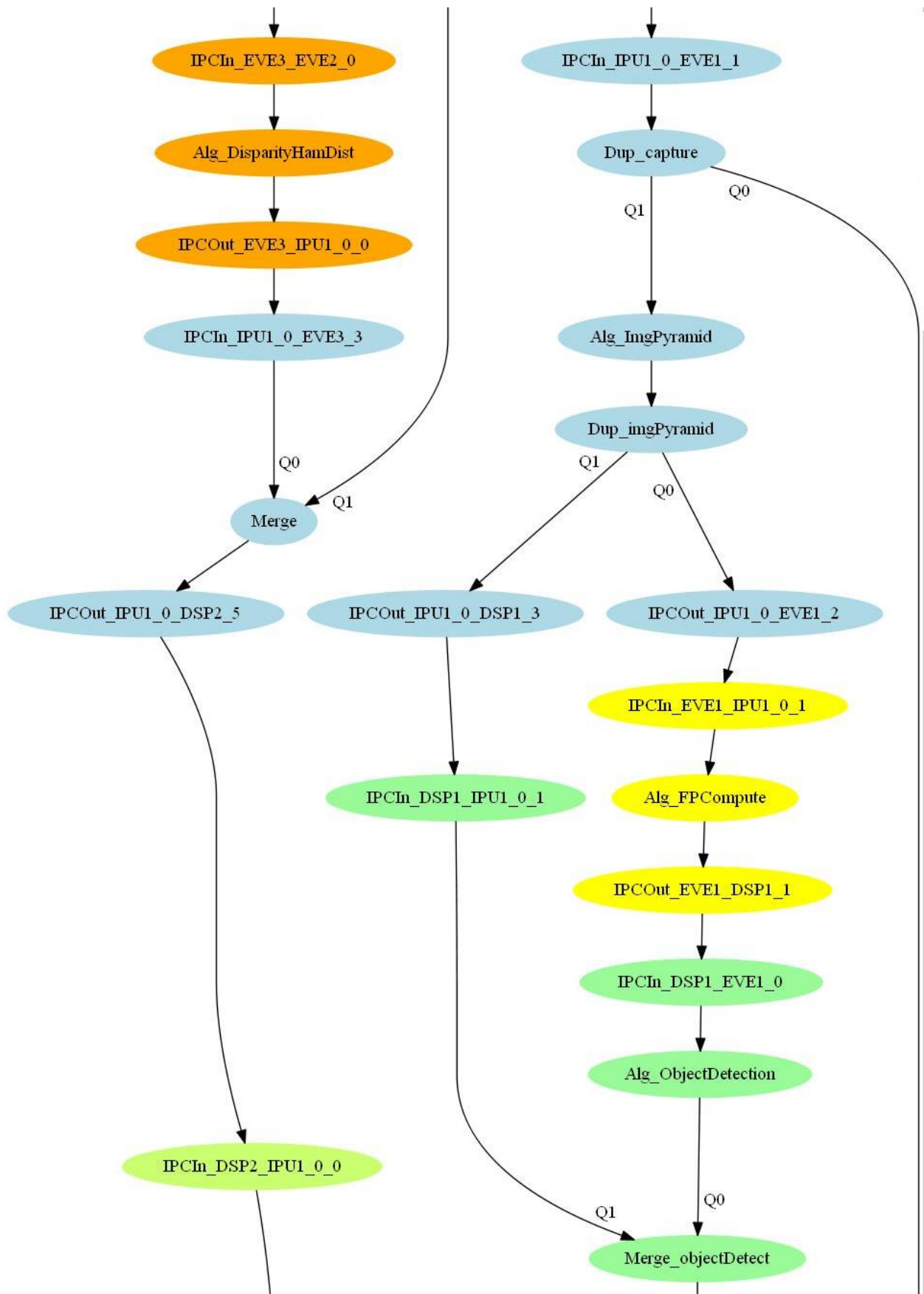
P.3.4. Dijagram toka slučaja upotrebe za izračun mape dispariteta na temelju ispravljenog stereo para slika.

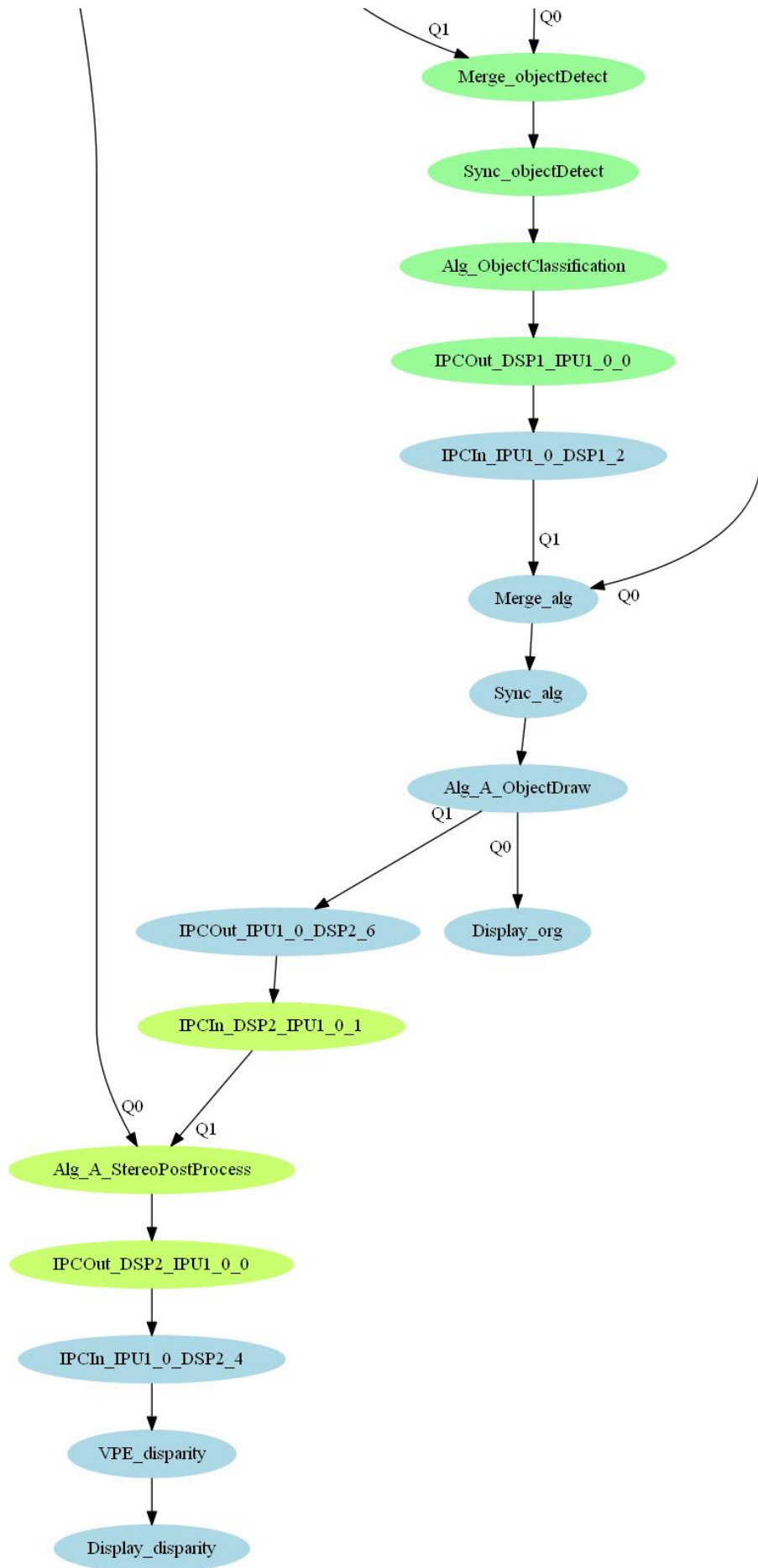




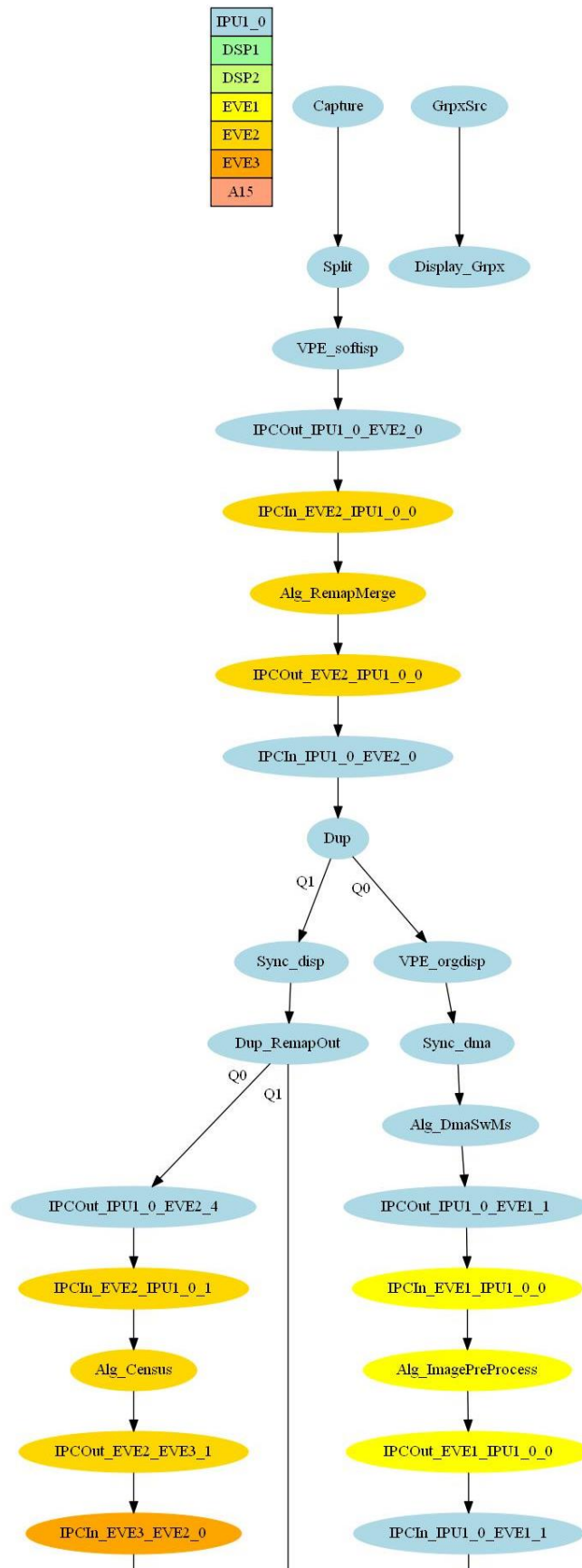
P.3.5. Dijagram toka slučaja upotrebe početne verzije konačnog rješenja zadatka diplomskog rada.

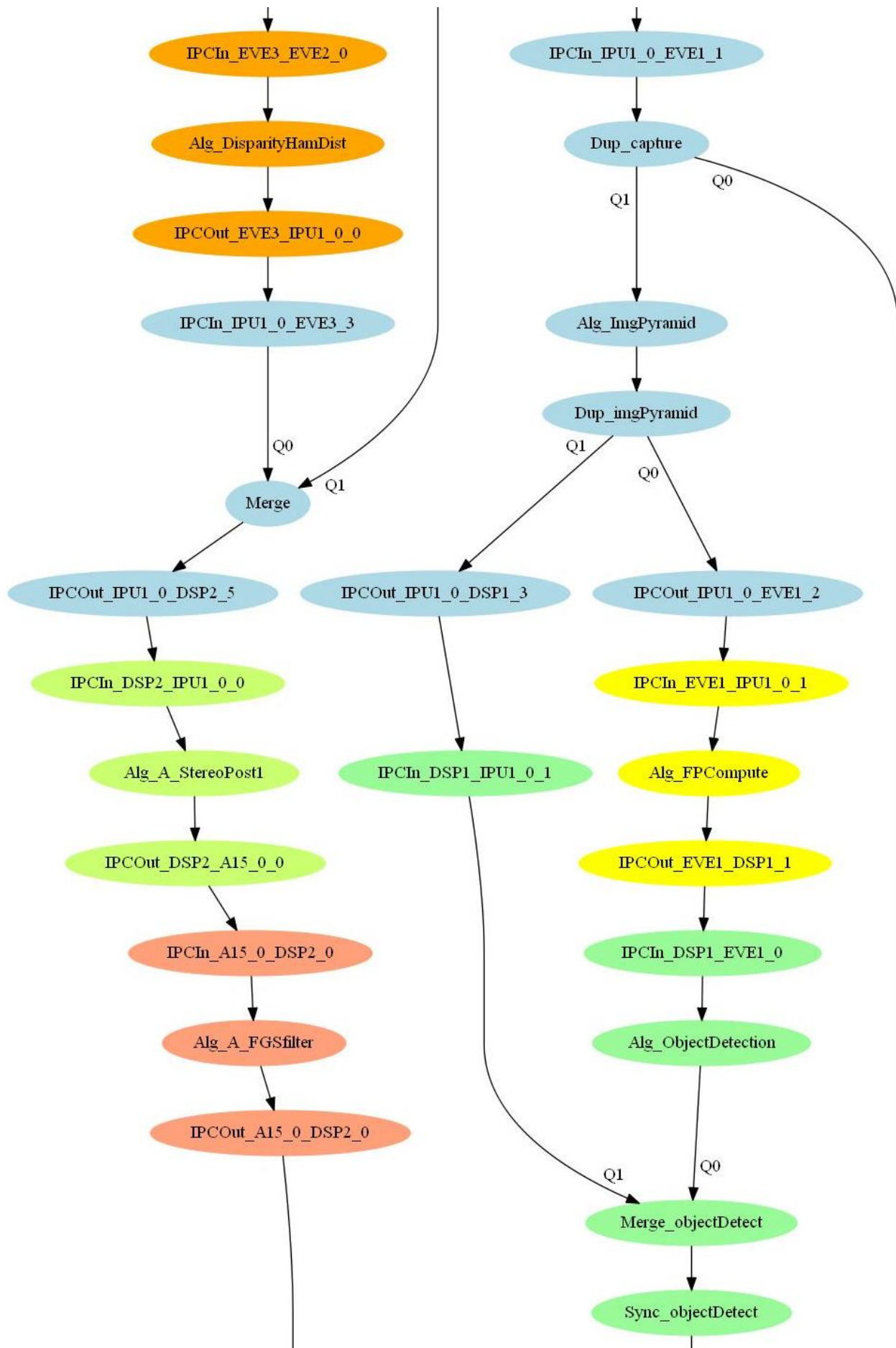


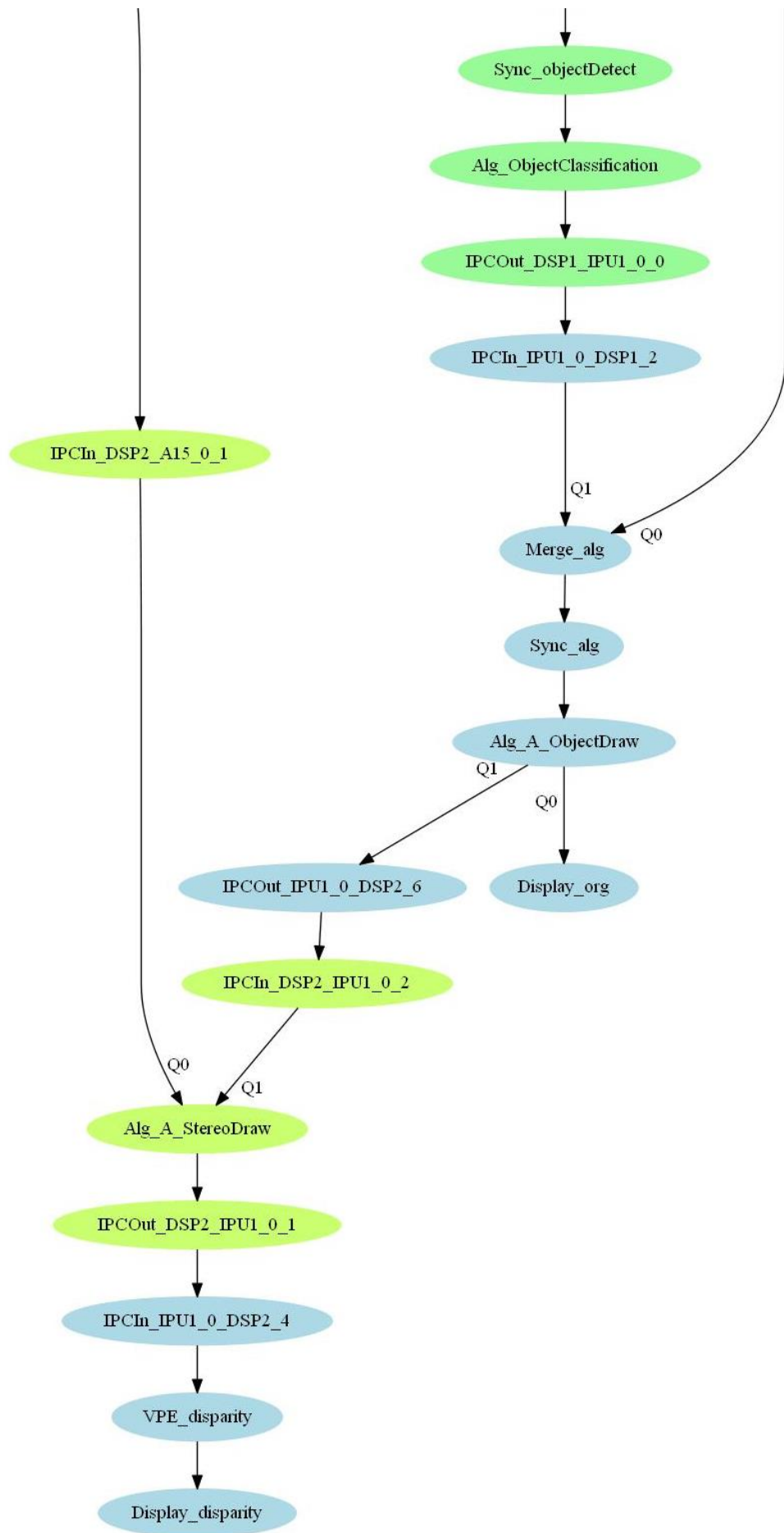




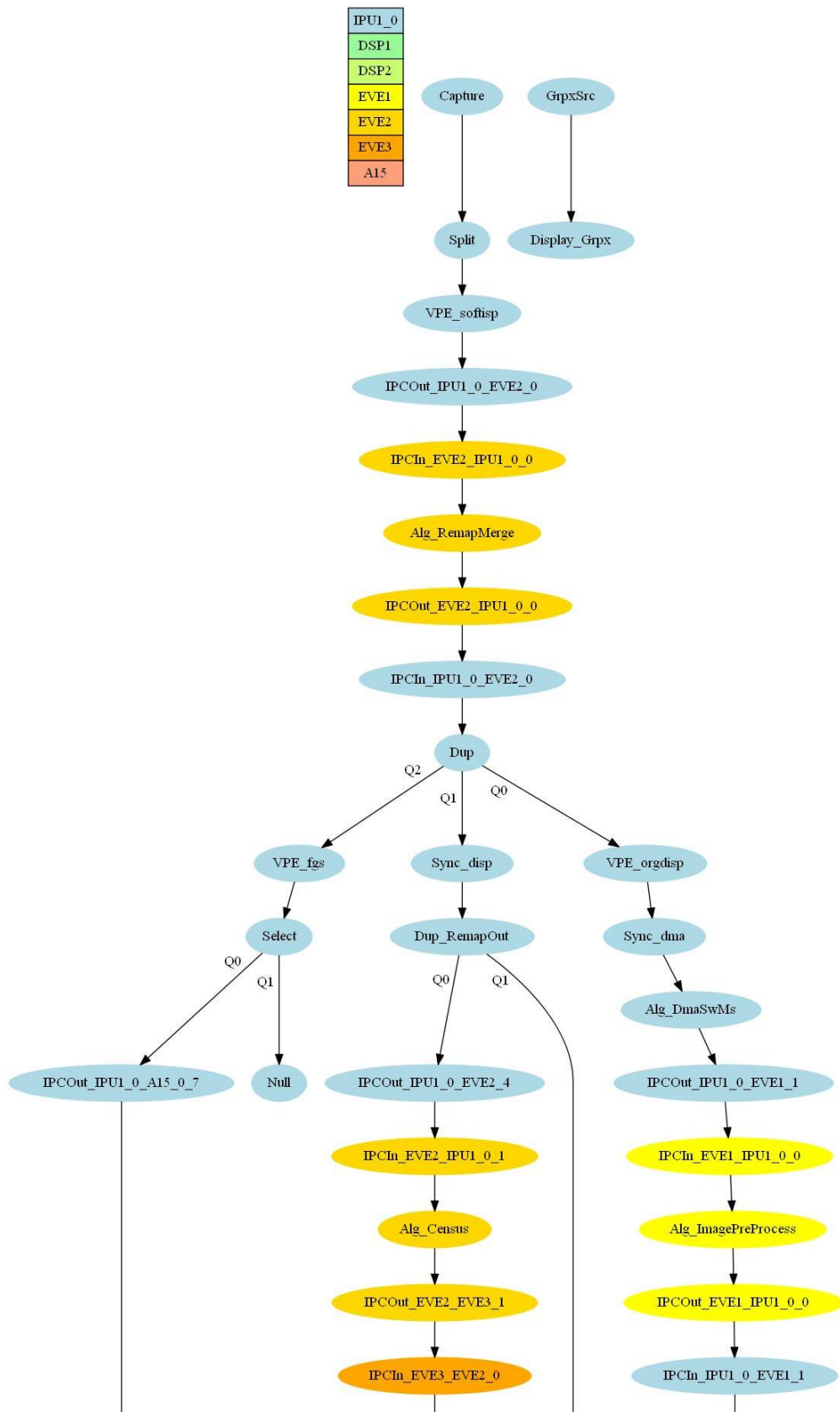
P.3.6. Dijagram toka slučaja upotrebe za određivanje udaljenosti detektiranog vozila zasnovanog na stereo paru slika, uz implementirani FGS filter bez navođenja.

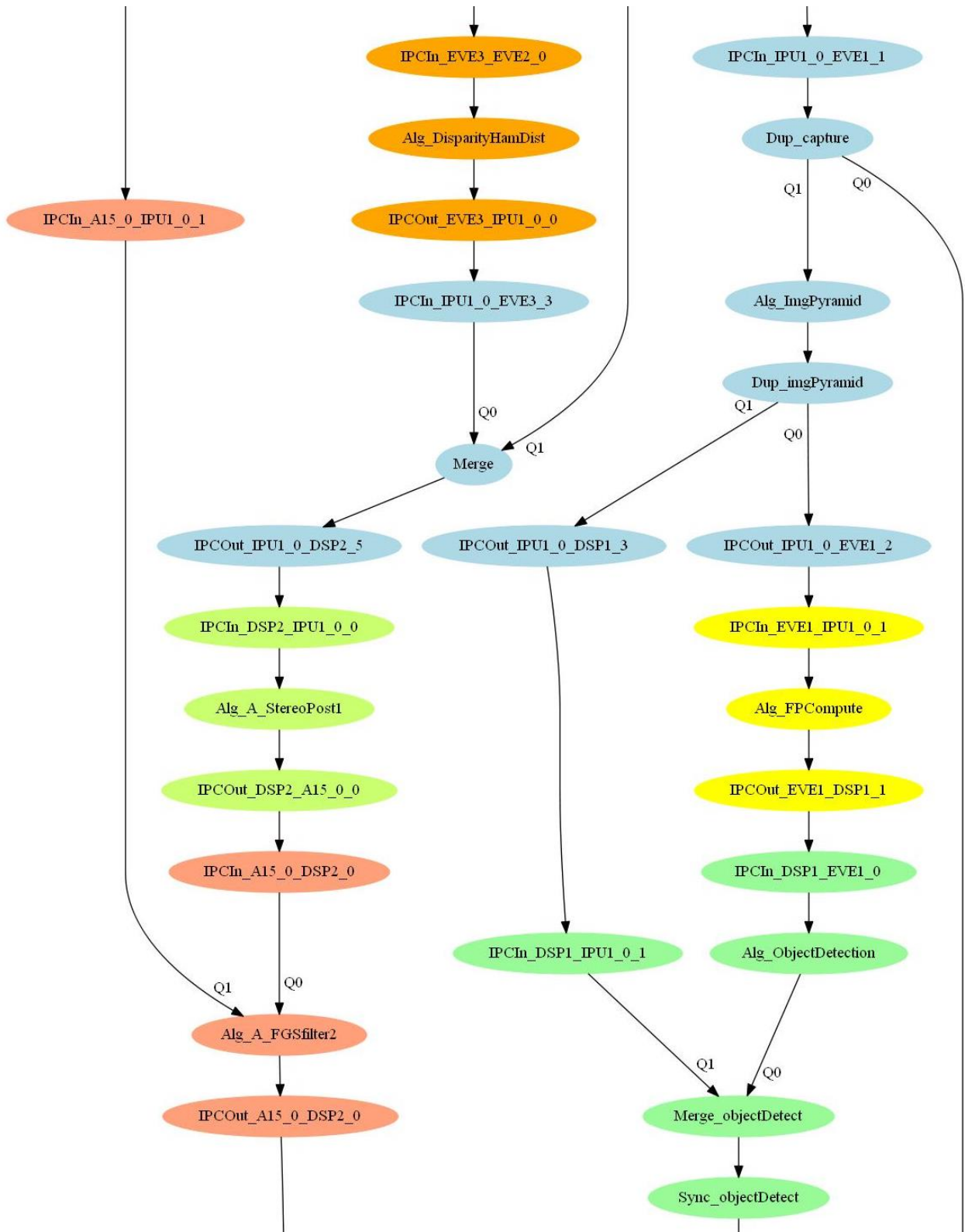


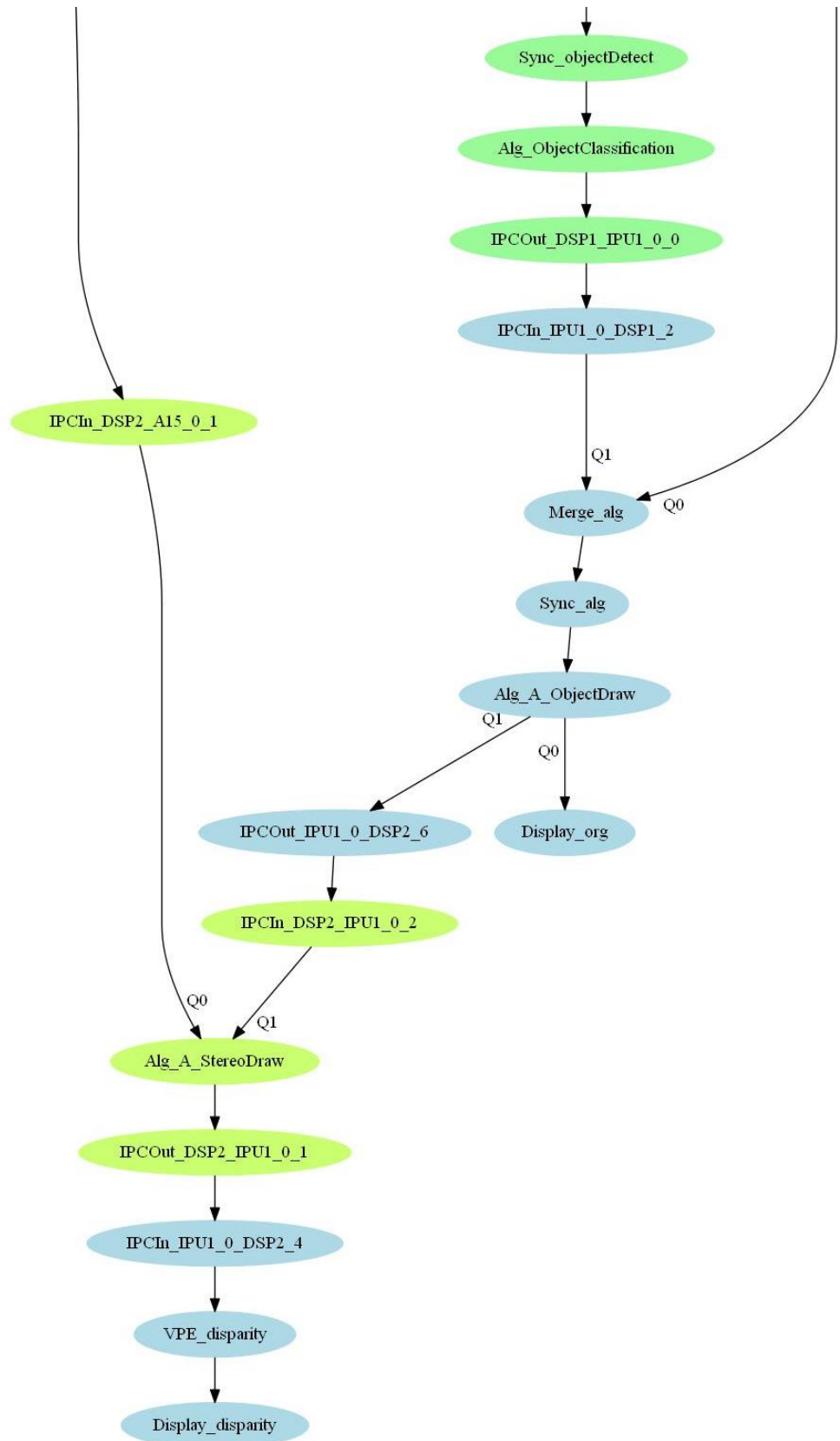




P.3.7. Dijagram toka slučaja upotrebe za određivanje udaljenosti detektiranog vozila zasnovanog na korištenju stereo para slika, uz implementirani FGS filter koji video okvir dobiven s lijeve kamere upotrebljava za navođenje.







P.4.1. – Mapa sa slikama grafičkih rezultata testiranih rješenja (nalazi se u elektroničkoj verziji).