Detekcija rizika od sudara i pametna kontrola brzine

Šimara, Eugen

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://urn.nsk.hr/urn:nbn:hr:200:376488

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2025-03-26**

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

DETEKCIJA RIZIKA OD SUDARA I PAMETNA KONTROLA BRZINE

Diplomski rad

Eugen Šimara

Osijek, 2020.

SADRŽAJ

1. UVOD 1
2. POSTOJEĆA RJEŠENJA ZA PROCJENU BRZINE I UDALJENOSTI VOZILA IZ VIDEO OKVIRA 3
3. ALGORITAM ZA DETEKCIJU RIZIKA OD SUDARA I PAMETNU KONTROLU BRZINE 7
3.1. Dijagram toka izrađenog algoritma 10
3.2. Viola-Jones algoritam 111
3.2.1. Treniranje Viola-Jones algoritma 155
3.3. Mjera uspješnosti detekcije (engl. *IoU – Intersection over Union*) 188
3.4. Procjena udaljenosti i brzine vozila koje se nalazi u video okviru 222
3.4.1. Procjena udaljenosti vozila 22
3.4.2. Procjena brzine vozila 24
4.TESTIRANJE I VERIFIKACIJA ALGORITMA 27
4.1. Klasifikacija detektiranih vozila 27
4.1.1. Testiranje provedeno na video okvirima 28
4.2. Testiranje točnosti izračunate brzine 271
5.ZAKLJUČAK 85
LITERATURA 86
SAŽETAK 88
ABSTRACT 89
ŽIVOTOPIS 90
PRILOZI 91

1. UVOD

Promet je jedan od najvećih uzročnika smrti u modernom vremenu (zauzima čak treće mjesto po brojnosti žrtava ako u obzir uzmemo sve moguće uzročnike smrti [1]). Potaknuti tom činjenicom kao i razvojem tehnologije, ljudi su odlučili smanjiti broj žrtava koji stradavaju iz godine u godinu, tako što su u automobile ugradili sustave za pomoć vozaču pri vožnji (engl *advanced driver assistance systems - ADAS*). ADAS se sastoji od više podsustava koji pomažu vozaču pri vožnji (bilo to izravno ili ne izravno) da ne dođe do prometne nezgode. U ADAS-u se koriste različiti senzori i algoritmi, koji s velikom točnošću mogu detektirati objekte u okolini vozila te poduzeti ispravnu radnju u vrlo kratkom vremenu kako bi se izbjegle neželjene situacije. Neki od sustava u potpunosti mogu reagirati umjesto vozača u određenom scenariju, dok se neki od njih aktiviraju samo uz vozačevu dozvolu. Ljudska pogreška i tromost prouzrokovale su mnoge nesreće, stoga ADAS sustavi preuzimaju sve veći dio funkcionalnosti. Tako npr. reakcija vozača Formule F1 iznosi oko 0,2 s od zamjećivanja do poduzimanje određene radnje, dok pomoćni sustav poduzme istu tu radnju u vremenu iskazanom u nekoliko desetaka milisekundi [2].

Mogućnosti autonomne vožnje rastu iz dana u dan pa se tako autonomnost dijeli na 5 razina, gdje svaka razina (što je ona veća) predstavlja sve veću kontrolu ADAS-a nad automobilom. U skorijoj budućnosti, svjedočiti ćemo o automobilima koji se u potpunosti samostalno kreću kroz grad od točke A do točke B koje je prethodno zadao vozač, bez ikakve opasnosti od sudara. Pomoću navedenih sustava ne postiže se samo smanjivanje opasnosti od sudara već se postiže i komfornija vožnja, pa se tako vozač može posvetiti osobnim stvarima poput čitanja/pisanja mailova, gledanju filmova itd.

U ovom radu napravljen je i opisan algoritam koji pomoću slika odnosno video okvira dobivenih s kamera, koje se nalaze na prednjoj i stražnjoj strani automobila, procjenjuje udaljenost i brzinu vozila koje se nalazi iza/ispred te poduzima određene radnje kako ne bi došlo do sudara. Algoritam izdvaja područje s video okvira koja je za njega relevantno te na izdvojenom području promatra objekte od interesa (u ovom slučaju automobile). Nakon prepoznavanja objekata te poznavajući njegovu širinu u elementima slike (engl. *pixel*), algoritam računa udaljenost objekta. Nadalje, pomoću promjene udaljenosti detektiranog vozila iz

prethodnog video okvira u odnosu na udaljenost vozila u trenutnom video okviru (vrijeme između ta dva okvira nam je poznato), te poznavajući brzinu vozila na kojem je algoritam implementiran, računa se brzina detektiranog vozila:

- Slučaj A: **prednja** kamera – vozilo se približava (izračunata brzina vozila ispred manja je od brzine vozila na kojem je implementiran algoritam).
- Slučaj B: **prednja** kamera – vozilo se udaljava (izračunata brzina vozila ispred veća je od brzine vozila na kojem je implementiran algoritam).
- Slučaj C: **stražnja** kamera – vozilo se približava (izračunata brzina vozila iza, veća od brzine vozila na kojem je implementiran algoritam).
- Slučaj D: **stražnja** kamera – vozilo se udaljava (izračunata brzina vozila iza, manja je od brzine vozila na kojem je implementiran algoritam).
- Slučaj E: **stražnja** i **prednja** kamera – vozilo se ne udaljava niti približava (izračunata brzina vozila iza i ispred je jednaka brzini vozila na kojem je implementirat algoritam).

Na temelju proračunate brzine (uzmimo za primjer slučaj A) algoritam bi vozilu trebao smanjiti brzinu kako bi se izbjegla potencijalna opasnost od sudara. Isto tako, postoji mogućnost da se vozila kreću jednakom brzinom (npr. kolona u gužvi) no udaljenost između vozila koje se nalazi ispred je premala, algoritam će također smanjiti brzinu vozila kako bi se postigao zadovoljavajući razmak između vozila te izbjegla potencijalna opasnost od sudara.

U narednom poglavlju, opisana su postojeća rješenja koja također u sebi sadrže detekciju vozila sa kamere te se vrši proračun njihove brzine. Treće poglavlje detaljno opisuje rješenje koje je obrađeno ovim radom dok četvrto poglavlje predstavlja testiranje i verificiranje algoritma na stvarnim videima snimljenim u različitim uvjetima (kiša, sunce, magla) sa prednje i stražnje strane kamere automobila.

2. POSTOJEĆA RJEŠENJA ZA PROCJENU BRZINE I UDALJENOSTI VOZILA IZ VIDEO OKVIRA

U zadnjem desetljeću razvijeno je mnogo algoritama za procjenu brzine i udaljenosti vozila. Brzinu vozila moguće je dobiti direktno (očitavanjem brzine pomoću senzora (lidara[1]/radara[2])) ili indirektno pomoću kamera. U nastavku ovog poglavlja, bit će opisana neka od postojećih rješenja.

U radu [3] predložena je metoda za računanje brzine pomoću kamere koja je postavljena na fiksnom objektu (rasvjetni stup, most, zgrada, itd.). Iz video okvira dobivenih pomoću kamere, proračun brzine se vrši u nekoliko koraka. Prvi korak jest određivanje voznih traka koje će se promatrati (područje od interesa) i to na način da se odrede linije koje omeđuju voznu traku. Detekcija linija vozne trake vrši se pomoću *Canny edge* [4] algoritma koji na slici pronalazi rubove, nakon čega se na slici na kojoj su pronađeni rubovi pomoću *Hough*-ove [5] transformacije pronalaze samo ravne linije pod određenim kutom. Drugi korak je prebacivanje slike s linijama u ptičju perspektivu te se metodom „oduzimanja pozadine“, prepoznaje automobil. Nakon što se vozila detektiraju, ona se prate pomoću neke od metode za praćenje (*Median flow,* KFC, CSRT, MIL, itd. [6]) određeni broj video okvira. Poznavajući traku u kojoj se vozilo kreće, određuje se početna linija promatranja i krajnja linija promatranja koje su okomite na smjer linija koje omeđuju voznu traku (slika 2.1.). Udaljenost horizontalnih linija je poznata i izračunata iz broja elemenata slike pomoću parametara kamere. Posljednji korak jest aproksimacija brzine vozila u voznoj traci, za koju je potrebno mjeriti vrijeme potrebno da zadnji kraj automobila prijeđe te dvije linije. Na osnovu vremena dobivenog mjerenjem i poznate udaljenosti između horizontalnih linija, izračunava se brzina automobila. Prednost ovog algoritma jest jednostavnost zbog korištenja pozadinskog oduzimanja što je ujedno i mana jer se algoritam ne može implementirati na sustav gdje kamera nije fiksno postavljena.

[1] Engl. ***LI**ght **D**etection **A**nd **R**anging* – detekcija objekata pomoću svjetlosnih valova
[2] Engl. ***RA**dio **D**etection **A**nd **R**anging* – detekcija objekata pomoću radio valova

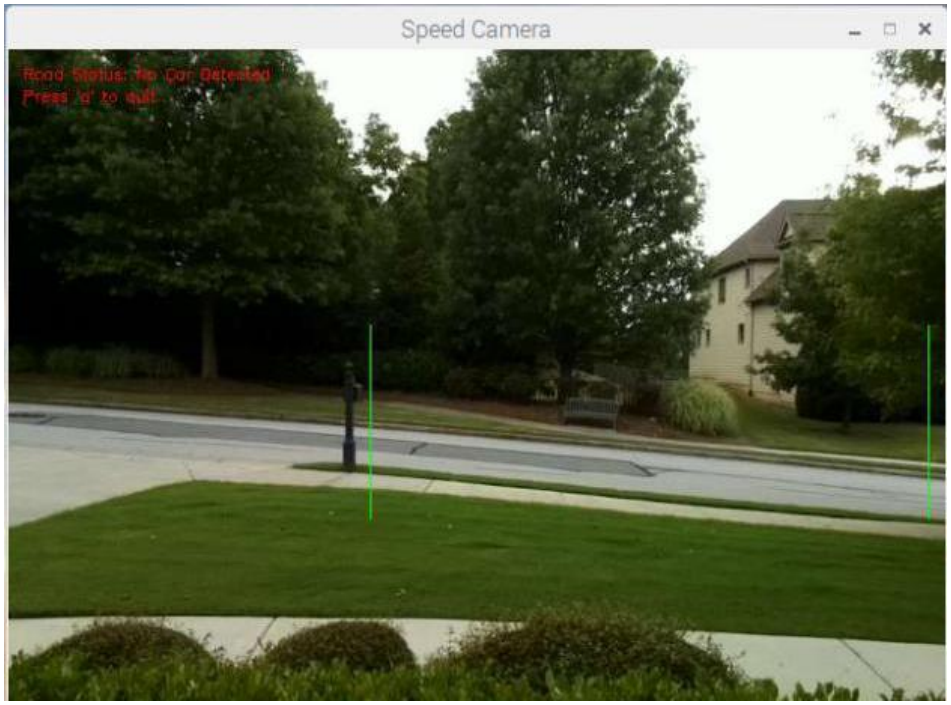
Slika 2.1. Prikaz krajnje u početne linije promatranja

U radu [7] izrađen je algoritam koji vrši detekciju i procjenu udaljenosti vozila koji se također sastoji od nekoliko koraka. Prvo se detektiraju vozila od interesa, a to se postiže pomoću *Viola-Jones* [8] algoritma koji koristi *Haarove* značajke [9], kako bi prepoznao objekt. Problem kod ovakve detekcije predstavlja nepreciznost *Viola-Jones* algoritma, odnosno pravokutnik koji označuje objekt od interesa ne omeđuje objekt isključivo od njegova lijeva do njegova desna ruba, već zahvaća dio pozadine što dovodi do pogreške u procjenjivanju širine objekta. Isto tako, detektirani objekt isključivo se omeđuje kvadratom što nije pogodno za objekte koje su pravokutni, trokutasti itd. (Slika 2.2.) te je potrebno provoditi dodatno filtriranje (npr. po boji ako objekt ima specifičnu boju) kako bi se dobila njegova stvarna širina. Proces detekcije je računalno zahtjevan proces te se između dvije detekcije radi praćenje koje zahtjeva manju računalnu snagu i vrijeme obrade. Za praćenje se koristi *Kalmanov* filtar [10]. Kako *Viola-Jones* daje jednaku visinu i širinu detektiranog objekta, visina objekta se zanemaruje te se za računanje udaljenosti koristi samo širina vozila. Poznavajući parametre kamere, širinu vozila (širina vozila je uzeta kao nekakva srednja vrijednost svih vrsta vozila što unosi pogrešku prilikom računanja udaljenosti a samim time i brzine) te broj elemenata slike koje zauzima traženi objekt na predanoj slici, računa se njegova udaljenost. Mana ovakvog algoritma jest pogreška koja se unosi nepoznavanjem točne širine objekta, čiju udaljenost želimo odrediti, dok mu je prednost to što je

omogućena funkcionalnost u stvarnom vremenu jer se koristi kombinacija detekcije i praćenja objekta.

Slika 2.2. Detekcija čovjeka koji je u sredini pomoću kvadrata i pomoću pravokutnika

U radu [11] predložena je metoda vrlo slična metodi iz rada [7]. Ovaj algoritam također ima kameru koja je fiksa (pričvršćena na nekakav objekt koji se nalazi kraj ceste gdje želimo mjeriti brzinu vozila). U ovom slučaju nema detekcije vozila i algoritam je napravljen isključivo za mjesta gdje prolaze samo vozila, npr. autocesta (nema pješaka niti drugih pokretnih objekata) i to na taj način da se prolazak vozila snima s boka. Određena su dvije imaginarne linije koje su postavljene u visini prolaska vozila (slika 2.3.). Prilikom prolaska automobila preko tih linija, program podiže zastavicu te se pamti vrijeme podizanja zastavice. Udaljenost između linija je proizvoljna (zadaje se unutar programa) te se na osnovu broja elemenata slike između tih dviju linija (linije su postavljene vertikalno na video okviru) i poznavanjem specifikacija kamere, određuje stvarna udaljenost koju omeđuju te dvije linije. Brojanjem video okvira koji prođu između okidanja prve i druge zastavice i poznavanjem broja okvira (engl. *FPS - frames per second*) koje video ima u jedinici vremena, izračunava se brzina vozila koje je prošlo kroz regiju koju se promatra. Mana ovakvog algoritma jest ta sto kroz regiju koja se promatra može proći bilo što (insekt koji je dovoljno blizu kameri, ptica, pješak itd.) te može doći do okidanja druge zastavice i prije nego je vozilo zapravo došlo do linije koja programski okida drugu zastavicu. Prednost ovakvog algoritma jest minimalistička računalna moć i jednostavnost implementacije.



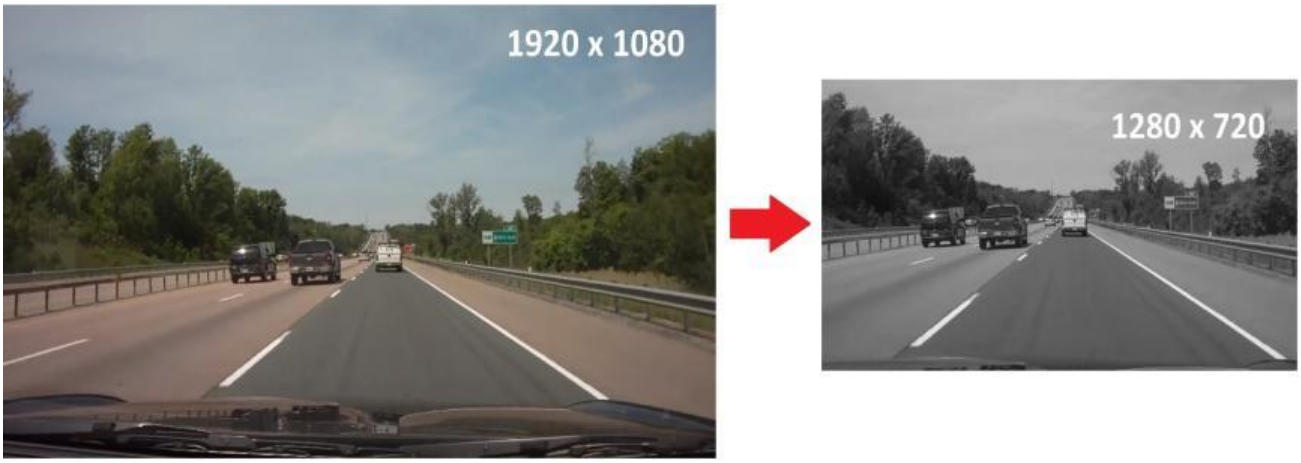
Slika 2.3. Prikaz linija pomoću kojih program okida zastavice [12]

3. ALGORITAM ZA DETEKCIJU RIZIKA OD SUDARA I PAMETNU KONTROLU BRZINE

U ovom odlomku dan je opis predloženog algoritma. Predloženi algoritam dijeli se u 6 glavnih koraka:

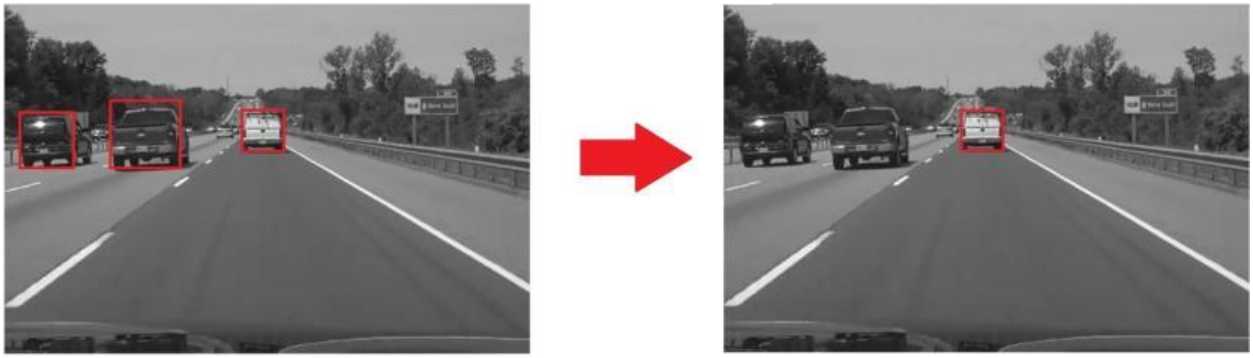
- Predobradu slike (engl. *Preprocesing*)
- Detekcija vozila
- Odbacivanje vozila koji nisu od interesa
- Praćenje detektiranih vozila
- Određivanje udaljenosti detektiranog vozila
- Određivanje brzine detektiranog vozila

Pod predobradu slike, podrazumijeva se smanjivanje veličine slike s jedne rezolucije (npr. 1920x1080) na rezoluciju 1280x720 kako bi se ubrzao algoritam, prebacivanje slike iz RGB formata u crno-bijeli format (engl. *RedGreenBlue – grayscale*) te odbacivanje područja koje nije od interesa sto je prikazano slikom 3.1.



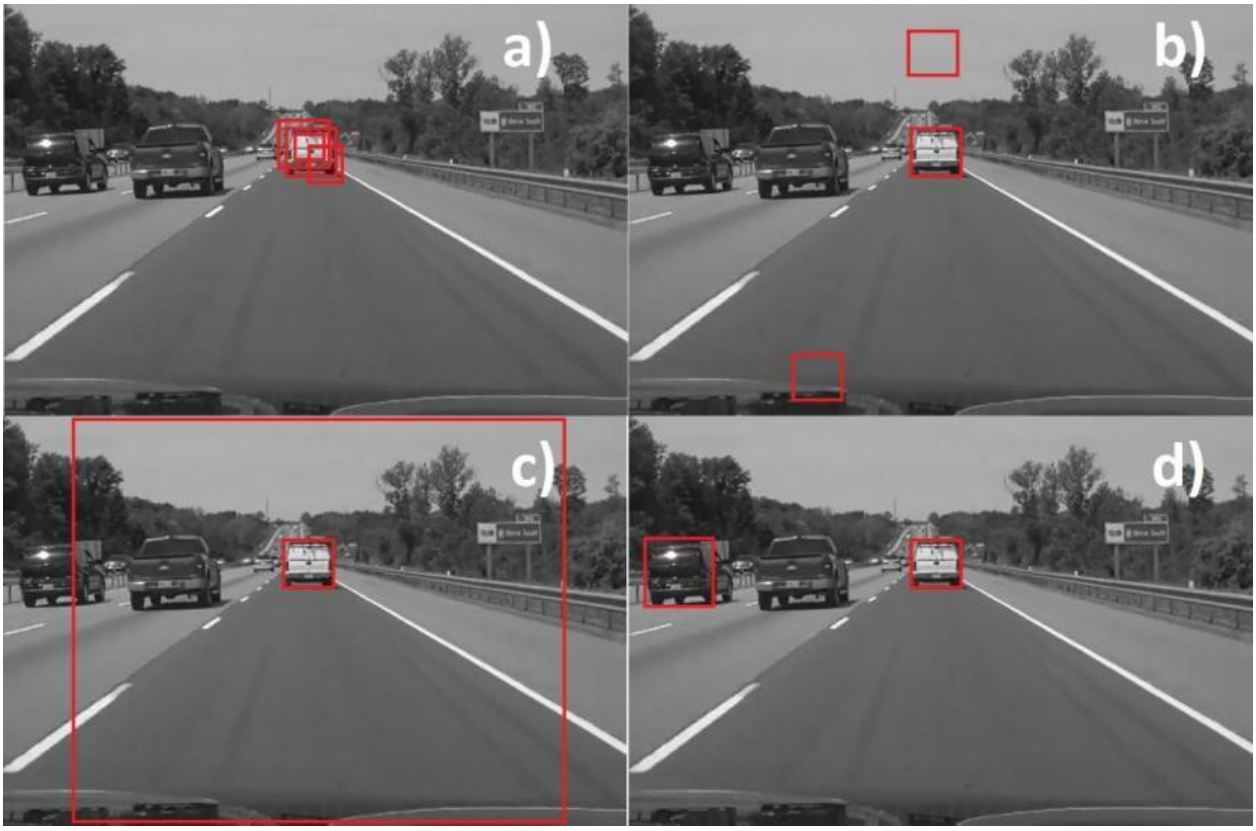
Slika 3.1. Prikaz pred obrade slike

Detekcija vozila vrši se pomoću *Viola-Jones* algoritma koji će u sljedećem poglavlju biti detaljno opisan. Nakon što je izvršena detekcija, na slici su detektirana sva moguća vozila te je neke od njih potrebno zanemariti jer je algoritmu potrebno vozilo koje se nalazi isključivo ispred njegove putanje kretanja što je prikazano na slici 3.2.



Slika 3.2. Filtriranje nepotrebnih vozila

Filtriranje nepotrebnih automobila odvija se na način da se promatraju koordinate graničnog pravokutnika (engl. *Bouding box*) detektiranih automobila, koji su dobiveni pomoću *Viola-Jones* algoritma. Tako npr., ne može postojati granični pravokutnik unutar graničnog pravokutnika, granični pravokutnik ne može biti označen negdje na nebu, granični pravokutnik ne može biti 600x600 elemenata slike, jer nije moguće da automobil zauzima više od pola originalne slike, granični pravokutnik koji sadrži x koordinatu koja je daleko od centra slike (npr. slika je širine 1280 elemenata slike, dakle sredina je na 640-om elementu slike što bi značilo da granični okvir koji ima *x* koordinatu vrijednosti npr. 40 biva odbačen iz razloga što je on udaljen 600 elemenata slike od centra slike) također mora biti odbačen. Neki od gore navedenih slučajeva su prikazani na slici 3.3.

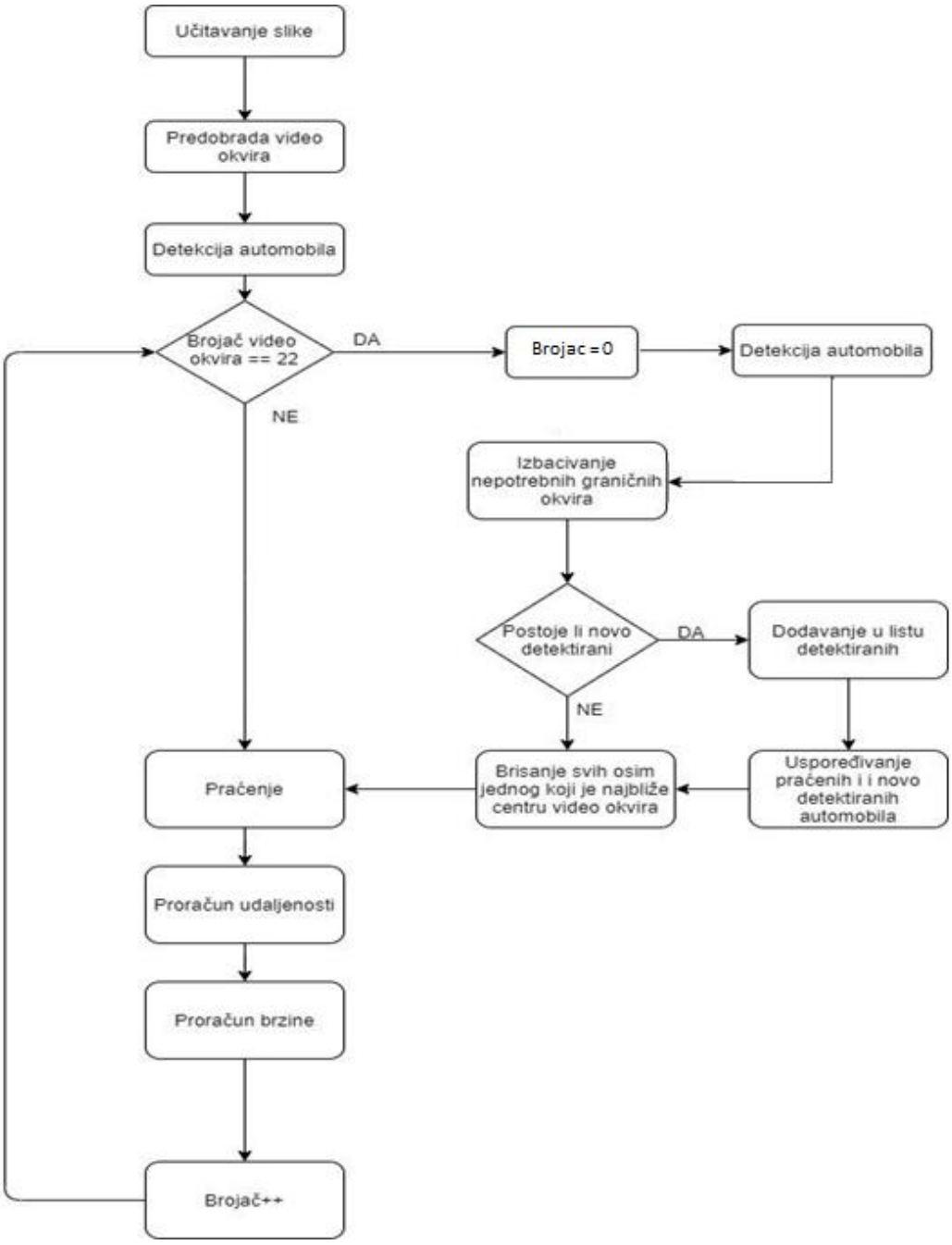


Slika 3.3 ***a)*** *preklapanje više graničnih okvira,* ***b)*** *iscrtavanje graničnog okvira na nebu,* ***c)*** *iscrtavanje prevelikog graničnog okvira,* ***d)*** *iscrtavanje graničnog okvira koji se ne nalazi u traci ispred*

Nakon što su vozila koja se ne nalaze na putanji kretanja uklonjena, radi se praćenje detektiranih vozila, kako bi se ubrzao rad algoritma, jer je *Viola-Jones* vrlo zahtjevan što se tiče računalne snage te onemogućuje rad u stvarnom vremenu (engl. *real-time*). Praćenje se vrši pomoću *Medianflow* algoritma i odvija se narednih 22 video okvira nakon detekcije (22 je određeno eksperimentalno) nakon čega se opet odvija detekcija i tako se neprestano isprepliću. *Medianflow* radi tako da mu se predaju elementi slike koji se žele pratiti. Elemente slike koji su mu ranije predani algoritam pokušava pronaći na svakom od narednih video okvira, gdje se za svaki video okvir gleda samo prethodni položaj promatranih elemenata slike. Kada su elementi slike pronađeni u trenutnom video okviru, računa se medijan vrijednost putanja svih tih elemenata slike te se aproksimira pomak objekta iz prethodnog video okvira u trenutni. Računanje brzine se temelji na procijenjenoj udaljenosti što bi značilo da ako se javi pogreška u

procjeni udaljenosti, javiti će se greška i u proračunu brzine vozila. Detaljniji opis proračuna udaljenosti i brzine slijedi u narednim poglavljima.

3.1. Dijagram toka izrađenog algoritma



Slika 3.16. Dijagram toka algoritma

Na slici 3.16. slikovito je prikazan dijagram toka izrađenog algoritma. Prvi korak je dohvaćanje video sekvence s kamere koja se analizira kao video okvir po video okvir (engl. *frame*). Nakon dohvaćanja video okvira, radi se predobrada video okvira gdje se slika iz RGB formata prebacuje u *Gray scale* format, nakon toga se video okvir skalira na 1280x720 veličinu te se odbacuju određene regije slike koje nisu od interesa. Na obrađenom video okviru provodi se detekcija pomoću *OpenCV* funkcije *car_cascade.detectMultiScale(gray, 1.09, 50)* kojoj se prosljeđuju 3 parametra. Prvi označava sliku na kojoj će se vršiti detekcija, drugi označava broj prema kojemu će se skalirati detektor (što je broj manji to će detekcija trajati duže) dok treći broj označava koliko susjednih detektora mora označiti da je upravo na tom mjestu vozilo kako bi se ono uistinu proglasilo vozilom.

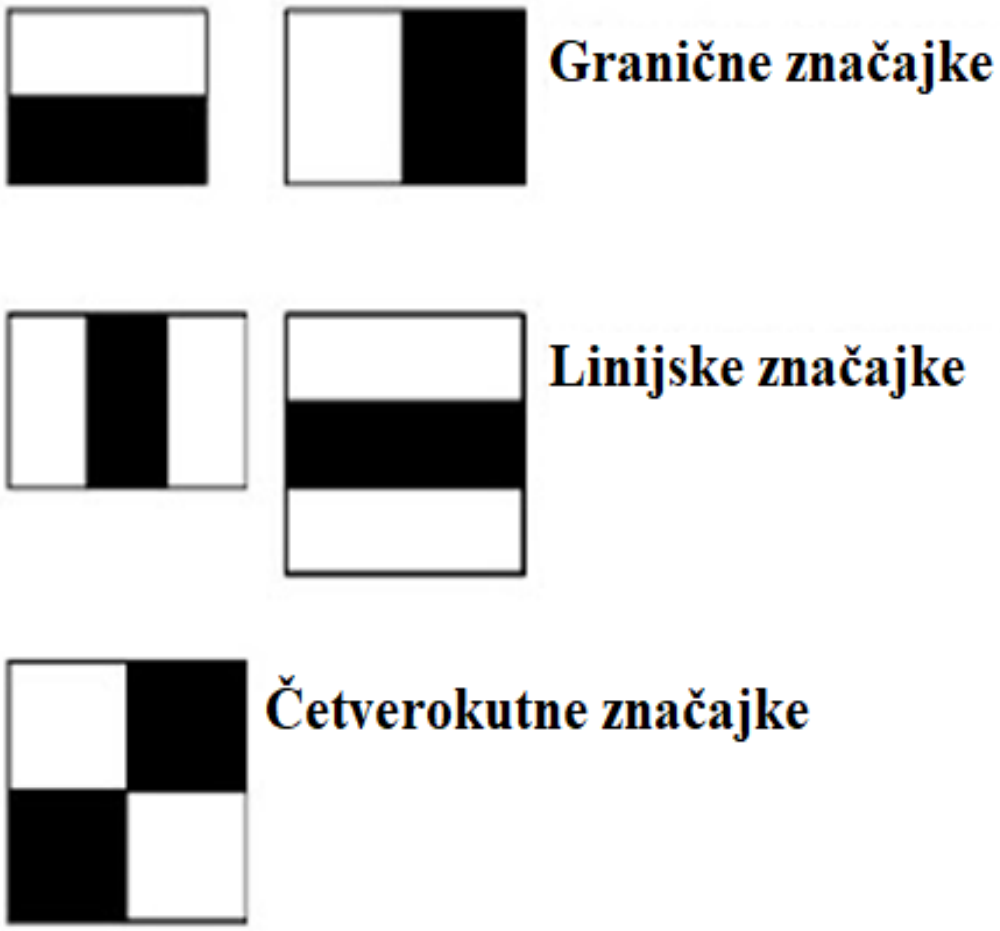
Detekcija se vrši svakih 22 (broj 22 određen je eksperimentalno – zadovoljava brzinu rada programa, vrši li se detekcija češće npr. svaki 21. video okvir, tada algoritam radi sporije ili ako se detektira svaki 23., tada se radi o presporom osvježavanju detektiranih automobila) video okvira što se kontrolira pomoću brojača, ako je brojač manji od 22 tada se samo nastavlja sa praćenjem ranije detektiranih vozila. Ako je brojač postigao vrijednost 22 tada se ponovo vrši detekcija, nakon čega se odbacuju irelevantne detekcije te se novo detektirani uspoređuju sa ranije praćenim vozilima. Postoji li razlika u pikselima veća od 20 piksela između praćenog i novo detektiranog vozila, tada se odbacuje praćenje i zamjenjuje se sa novom detekcijom. Na posljetku se radi odbacivanje svih detektiranih vozila osim jednog, onog najbližeg sredini jer algoritam mora prilagođavati brzinu vozila u odnosu na brzinu vozila ispred i iza sebe u istoj traci. Nakon pronalaska isključivo jednog relevantnog vozila, slijedi proračun brzine i udaljenosti te poduzimanje određene radnje (npr. kočenje ako se vozilo nalazi preblizu). Sve prethodno navedeno ponavlja se iz video okvira u video okvir. Skripta u kojoj je pohranjen kod algoritma može se pronaći na DVD-u priloženom uz ovaj rad u mapi Prilog P.3.1.

3.2. Viola-Jones algoritam

Početkom 2000. godine, Paul Viola i Michael Jones predlažu metodu detekcije objekata pomoću *Haarovih* značajki te tako nastaje algoritam za detekciju objekata koji dobiva ime po navedenom dvojcu *Viola-Jones* [8]. Njihov pristup se temelji na učenju kaskadne funkcije koja

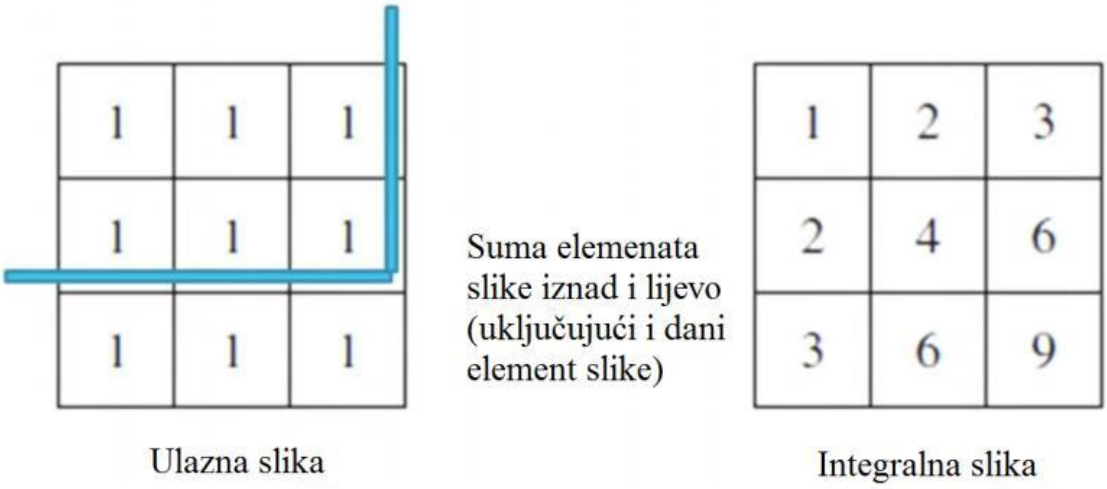
se sastoji od više slabih klasifikatora, te se njihovom linearnom kombinacijom dobiva jaki klasifikator. *Viola-Jones* algoritam koristi kaskadu više jakih klasifikatora što mu omogućuje vrlo visoku preciznost.

Prvenstveno je služio za detekciju lica, ali je našao svoju primjenu i za detekciju drugih objekata te je tako korišten i u ovom radu za detekciju određene vrste vozila (u ovom slučaju isključivo automobila). Temelj rada ovog algoritma zasnovan je na ranije spomenutim *Haarovim* značajkama koje se mogu pojaviti u svega nekoliko oblika što je prikazano na slici 3.4.



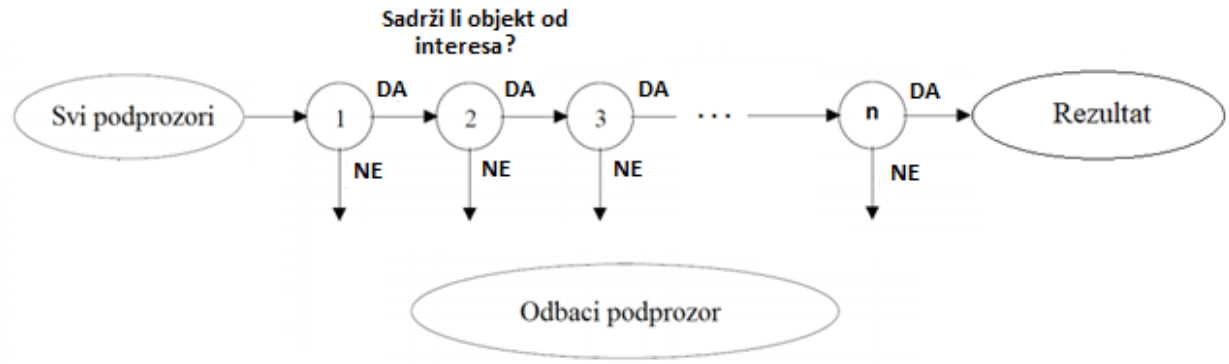
Slika 3.4. Haarove značajke [8]

Kako bi kaskadna funkcija davala što točnije rezultate odnosno kako bi što točnije prepoznavala traženi objekt, potrebno ju je istrenirati. Treniranje je proces gdje se kaskada uči na temelju slika (pozitivnih i negativnih) koje ranije moraju biti sakupljene. Pozitivne slike su slike koje u sebi sadrže objekt od interesa (u ovom slučaju automobil) te ih je potrebno što više. Negativne slike predstavljaju skup slika koje nipošto ne smiju sadržavati objekt od interesa jer bi to dovelo pogreške u daljnjem radu. U ovom radu, trening je obavljen na 5000 pozitivnih i 4000 negativnih slika automobila. Skup koji sadrži pozitivne slike, izrađen je samostalno te su slike napravljene s pomoću kamere mobitela Samsung Galaxy S10+, dok je negativni skup slika preuzet s interneta. *Viola-Jones* algoritam koristi *Haarove* značajke, *AdaBoost* algoritam te integralnu sliku koji znatno ubrzavaju proces učenja. Algoritam dijeli sliku na manje dijelove koji se nazivaju podprozori i iznose svega 24x24 elementa slike i na svakom tom podprozoru mora provući sve moguće veličine (1x2, 2x2, 4x3, itd.) svih značajki (značajka tip 1,2,3,4 i 5) te se dobiva broj od čak 160 000+ značajki [13]. Na svakom od podprozora radi se sumiranje vrijednosti piksela iznad područja prekrivenog bijelim dijelom karakteristike te sumiranje vrijednosti iznad područja prekrivenog crnim dijelom karakteristike nakon čega se te dvije vrijednosti oduzimaju. Ovo predstavlja dugotrajan proces zbog čega je uvedena integralna slika [14]. Integralna slika predstavlja sumu vrijednosti elemenata te slike iznad i lijevo od promatranog elementa slike što je prikazano na slici 3.5.



Slika 3.5. Primjer izračuna integralne slike [13]

AdaBoost algoritam radi na principu odbacivanja svih irelevantnih klasfikatora što značajno ubrzava proces treniranja pa se tako broja od prethodno navedenih 160 000+ klasifikatora, spušta na svega par tisuća relevantnih klasifikatora. Nakon *AdaBoost* algoritma, dobiva se kaskada klasifikator, čija je shema prikazana slikom 3.6., kroz koji se propušta slika, a izlaz kaskade klasifikatora donosi rezultat (*true* ili *false* odnosno sadrži li slika traženi objekt ili ga ipak ne sadrži). Svaki dio slike (podprozor) mora proći kroz kaskadu klasifikatora te svaki podprozor prolazi kroz sljedeću razinu u kaskada klasifikatoru, sve dok ne biva odbačen na jednoj od razina. Ako podprozor prođe sve razine u kaskadi klasifikatoru, tada se smatra da je objekt od interesa pronađen na tom djelu slike.

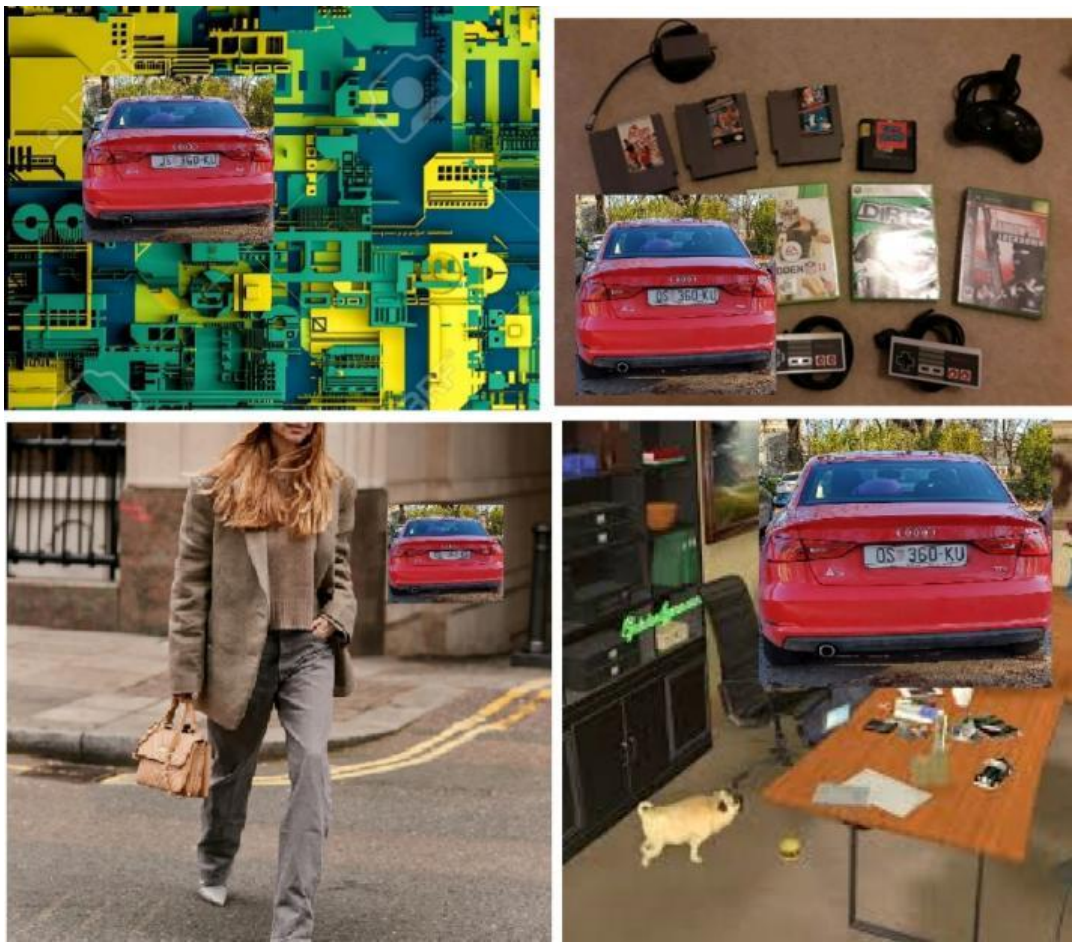


Slika 3.6. Prikaz rada kaskade klasifikatora

Viola-Jones algoritam ima svoje nedostatke i prednosti. Prednost mu je ta što je vrlo jednostavan, brz i učinkovit algoritam (izuzevši prikupljanje skupa pozitivnih i negativnih slika koje može biti dugotrajno). Mana jest ta što je dimenzija objekta koji se traži unaprijed definirana te će se detektirani objekt uvijek tražiti u prije definiranoj veličini (npr. 24x24, 300x300, što neće omogućavati detekciju istog objekta koji ima dimenziju 150x300). Mana jest i vrijeme potrebno da bi se kaskada istrenirala te neotpornost algoritma na rotaciju ili iskrivljenost traženog objekta.

3.2.1. Treniranje Viola-Jones algoritma

Kako bi se uspješno istrenirao detektor koji će prepoznavati automobile, potrebno je sakupiti određen broj pozitivnih slika (nekoliko tisuća). Sakupljanjem više slika za treniranje neće rezultirati nužno bolje istreniranim detektorom. Budući da detektor mora prepoznavati i prednju i stražnju stranu automobila, potrebno je sakupiti slike prednje i stražnje strane automobila različitijih proizvođača, kako bi uspješnost detektiranja bila što bolja. Pozitivne slike se mogu i „izraditi“ i to na taj način da se traženi objekt polijepi u različitim veličinama i pozicijama na određenom broju slika. Primjer ovakve izrade pozitivnog skupa nalazi se na slici 3.7. Treniranje na ovakvom skupu slika, donijeti će lošije rezultate od rezultata koji su dobiveni na stvarnim slikama [15].

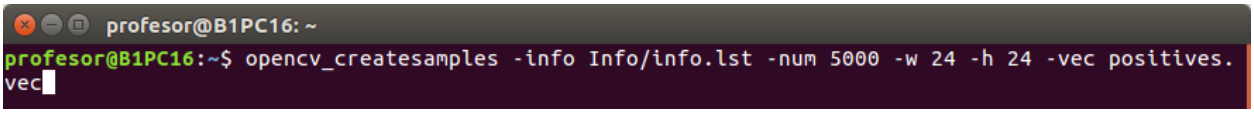


Slika 3.7. Četiri umjetno kreirane pozitivne slike

U ovom radu nije korištena prethodno navedena opcija kreiranja pozitivnih slika već su sve slike uslikane u stvarnim uvjetima. Slike se mogu se pronaći na DVD-u priloženom uz ovaj rad u mapi Prilog P.3.2. Uslikani su automobili na parkingu i to na taj način da je svaki

Slika 3.8. Prikaz skupa pozitivnih slika

automobil uslikan iz 3 različita kuta – lijevo s boka, iza/ispred, desno s boka, slika 3.8. Uslikano je 5000 takvih slika što bi značilo da je uslikano oko 1667 automobila različitih boja, klasa, proizvođača. Nakon prikupljene baze slike, potrebno je osigurati da su sve slike dovoljno velike odnosno da su veće od dimenzije koja će biti navedena kao najmanja prilikom treninga *Viola-Jones* algoritma (u ovom slučaju radi se o dimenziji 24x24 elementa slike). Za potrebe treninga potrebno je kreirati sljedeće datoteke: *info.lst*, *bg.txt* i *positives.vec*. Navedene datoteke mogu se pronaći na DVD-u priloženom uz ovaj rad u mapi Prilog P.3.3. Datoteka *info.lst* sadrži podatke o svim pozitivnim slikama, odnosno sadrži dimenziju objekta koji trebamo detektirati, njegovu poziciju na slici te broj koji predstavlja broj objekata na slici (npr. jedna slika može sadržavati 2 automobila, dakle, oba automobila će imati svoje koordinate odnosno nalazište na slici). Sve pozitivne slike sadrže isključivo jedan automobil te je *info.lst* kreiran pomoću skripte *info.py* koja u tekstualnu datoteku upisuje sljedeće: **ime_slike, broj_automobila_naSlici, x_koordinatu, y_koordinatu, h_visinu** te **w_sirinu** automobila na slici (npr. 0001.jpeg, 1, 0, 0, 300, 300). Datoteka *bg.txt* sadrži sve nazive negativnih slika koje će se koristiti prilikom treniranja detektora. Nakon što su napravljene prethodno navedene datoteke te nakon što su



Slika 3.9.. Naredba za kreiranje datoteke positives.vec

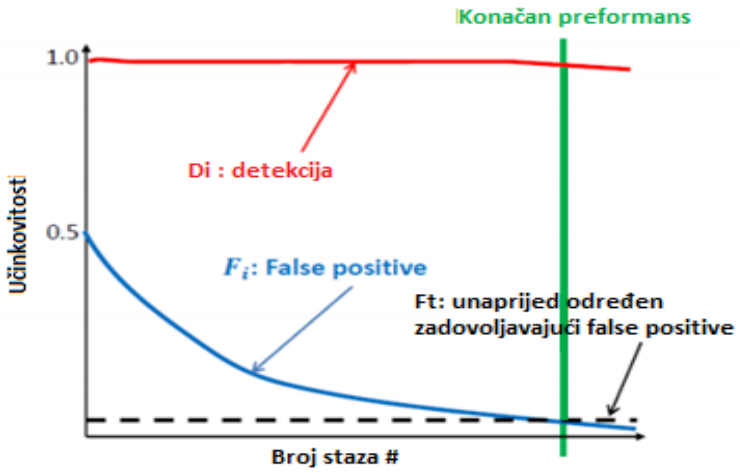
prikupljene slike, potrebno je kreirati *positives.vec* datoteku. Ona se kreira tako što se u terminal upiše naredba prikazana slikom 3.9.

Funkciji *opencv_createsamples* predaje se putanja do *info.lst* datoteke, broj pozitivnih slika, veličina koja opisuje najmanju moguću veličinu traženog objekta te putanja do datoteke *positives.vec*. Kada je učinjeno sve prethodno navedeno, moguće je pokrenuti treniranje detektora naredbom prikazanom na slici 3.10. Funkciji *opencv_traincascade* predaje se putanja do datoteke *data*, gdje će se spremati detektor nakon pojedine razine treniranja, putanja *positives.vec* koja je prethodno kreirana, putanja do datoteke *bg* koja sadrži sve nazive negativnih slika, broj pozitivnih slika, broj negativnih slika, broj razina do koje se želi „utezati“ detektor odnosno kaskada te najmanja moguća visina i širina traženog objekta.

```
profesor@B1PC16: ~
profesor@B1PC16:~$ opencv_traincascade -data Data -vec positives.vec -bg bg.txt -numPos 5000 -num
Neg 4000 -numStages 24 -w 24 -h 24 -mode ALL
```

Slika 3.10. Naredba za pokretanje treniranja kaskade

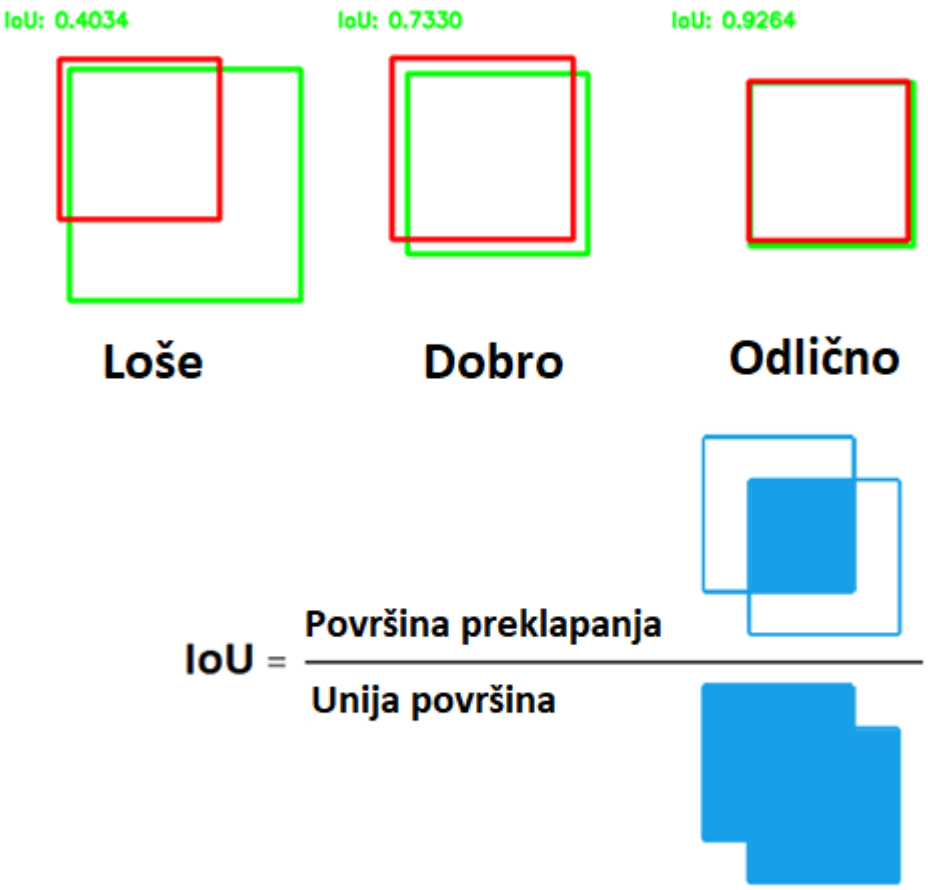
Prilikom treniranja potrebno je paziti da se detektor ne pretrenira. Pretreniranost detektora znači da postoji mogućnost da na slici na kojoj postoji objekt od interesa, on nije prepoznat. Slika 3.11. najbolje prikazuje točku gdje treba zaustaviti treniranje kako bi se došlo do najveće preciznosti detektora, a to je točka gdje je detekcija još uvijek dovoljno velika (nije počela previše padati), a netočna detekcija je dovoljno mala da zadovoljava unaprijed određene zahtjeve (npr. 5% netočne detekcije).



Slika 3.11. Prikaz odnosa razina testiranja i točnosti detektora [13]

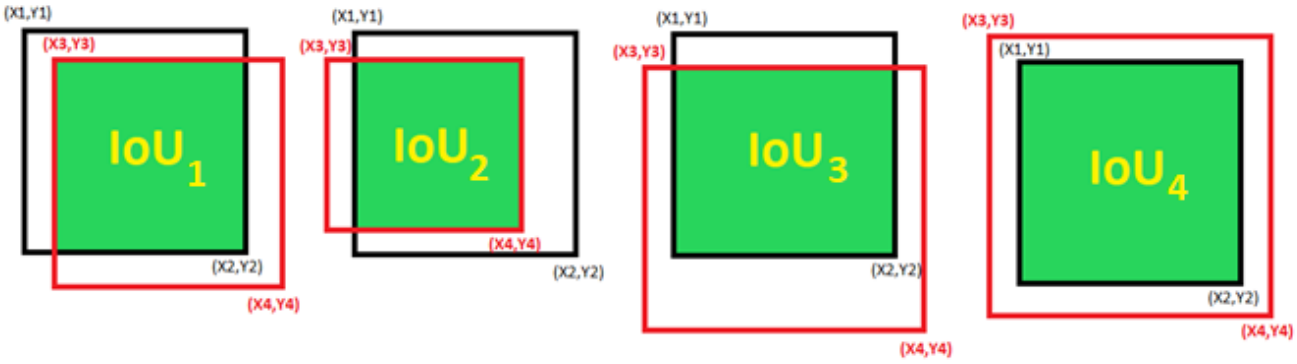
3.3. Mjera uspješnosti detekcije (engl. *IoU – Intersection over Union*)

Mjera uspješnosti detekcije iskazana je u postotcima i računa se kao omjer površine preklapanja stvarnog graničnog okvira (engl. *Ground-truth bounding box*) i detektiranog graničnog okvira (engl. *Predicted bounding box*), te unije površine stvarnog graničnog okvira i detektiranog graničnog okvira. Na slici 3.12. prikazana je mjera uspješnosti detekcije na 3 primjera.



Slika 3.12. Prikaz računanja IoU i 3 primjera IoU [16]

Viola-Jones algoritam svaki detektirani automobil predstavlja s *x*,*y*,*w* i *h* vrijednostima, gdje *x* i *y* predstavljaju koordinate lijeve gornje točke graničnog okvira dok *w* i *h* predstavljaju širinu i visinu automobila. Postoji više mogućih scenarija preklapanja graničnih okvira, koji su prikazani na slici 3.13.



Slika 3.13. Prikaz 4 različita preklapanja graničnih okvira

Prvo je potrebno izvesti formule za svaki od navedenih scenarija:

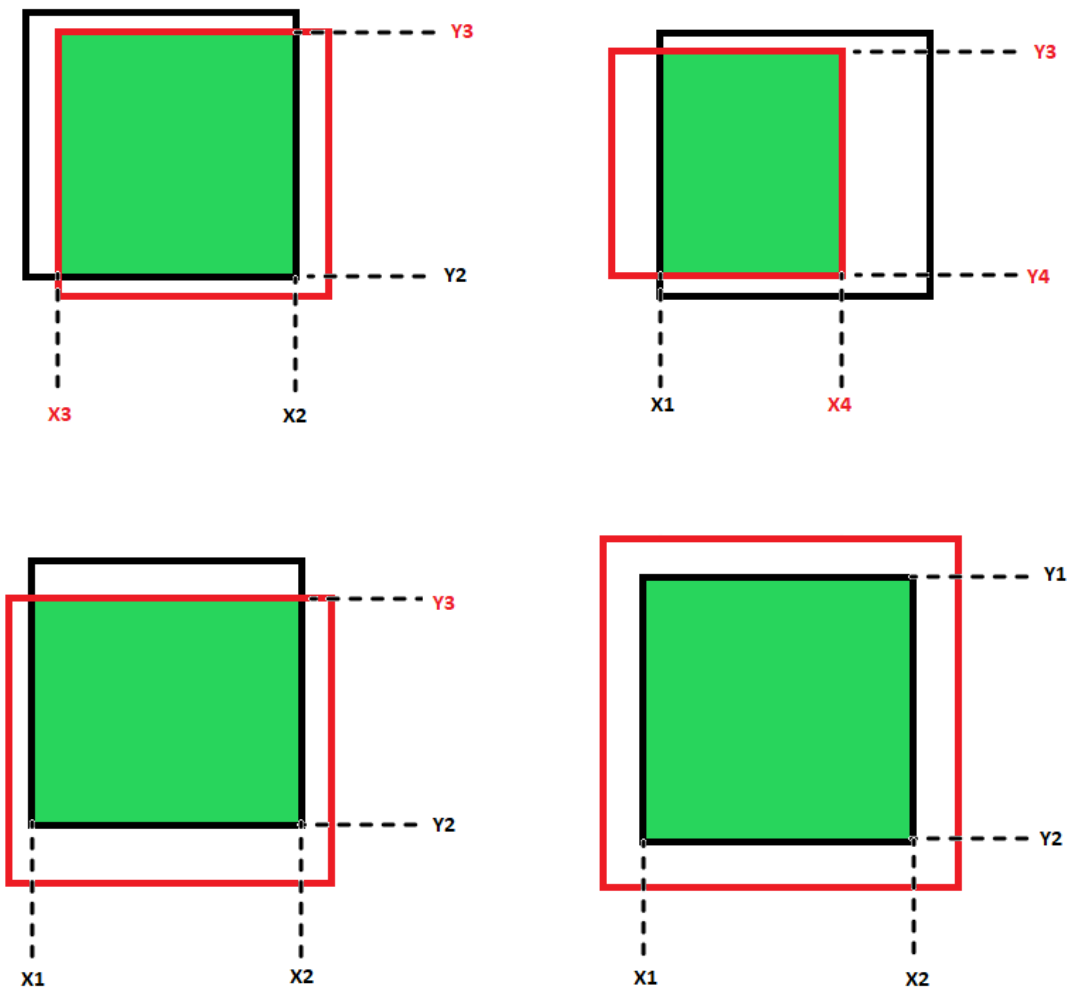
$$IoU_1 = \frac{(x2-x3)*(y2-y3)}{(x2-x1)*(y2-y1)+(x4-x3)*(y4-y3)-(x2-x3)*(y2-y3)}, \quad (3\text{-}1)$$

$$IoU_2 = \frac{(x4-x1)*(y4-y3)}{(x2-x1)*(y2-y1)+(x4-x3)*(y4-y3)-(x4-x1)*(y4-y3)}, \quad (3\text{-}2)$$

$$IoU_3 = \frac{(x2-x1)*(y2-y3)}{(x2-x1)*(y2-y1)+(x4-x3)*(y4-y3)-(x2-x1)*(y2-y3)}, \quad (3\text{-}3)$$

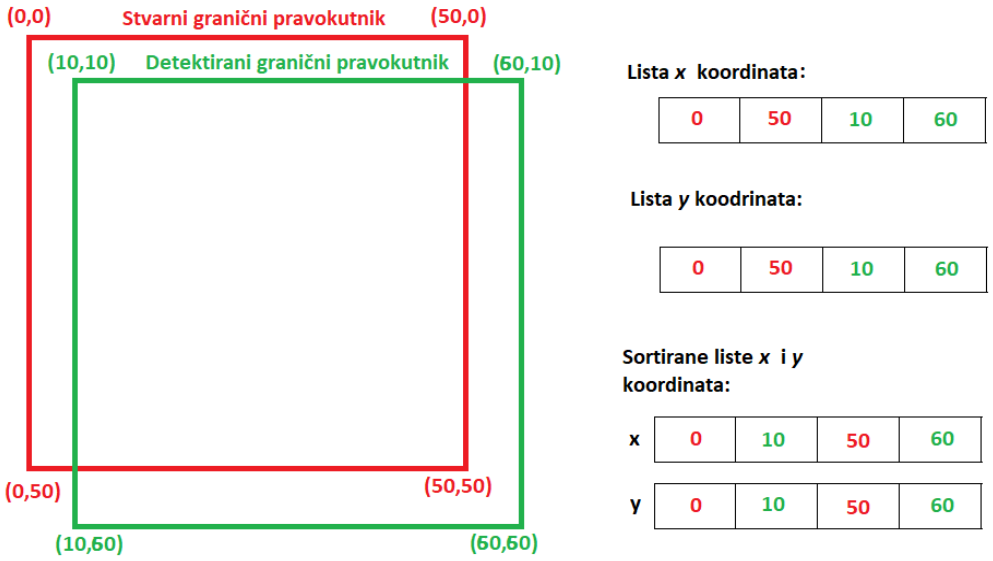
$$IoU_4 = \frac{(x2-x1)*(y2-y1)}{(x4-x3)*(y4-y3)+(x2-x1)*(y2-y1)-(x2-x1)*(y2-y1)}. \quad (3\text{-}4)$$

Kako se može vidjeti, svaki od scenarija sadrži po 4 *x* koordinate i po 4 *y* koordinate. U svakom od scenarija u brojniku su uvrštene dvije središnje *x* koordinate koje se oduzimaju (uvijek se od veće oduzima manja) što je pomnoženo s razlikom dvije središnje *y* koordinate (također od veće oduzeta manja). Nadalje, u nazivniku su jednostavno izračunate površine jednog graničnog okvira te su sumirane sa površinom drugog graničnog pravokutnika i oduzeta im je vrijednost koja se nalazi u nazivniku kako bi se dobila unija. Slika 3.14. prikazuje gore opisano, dakle za bilo koji od scenarija može se zaključiti da je brojnik uvijek uvršteno samo središnja dva člana *x* i središnja dva člana *y*. Skripte koje služe za izračun IoU i odstupanja širine stvarnog i detektiranog graničnog okvira mogu se pronaći na DVD-u priloženom uz ovaj rad u mapi Prilog 3.5.



Slika 3.14. Slikoviti prikaz središnja dva člana x i y

Navedeno rješenje treba implementirati u vlastiti algoritam i to je izvedeno na taj način da se koordinate stvarnog i predviđenog graničnog okvira postave u jednu listu (zasebna lista za *x* koordinate i zasebna lista za *y* koordinate) te se sortiranjem liste (od najmanjeg ka najvećem) omogućuje pristup dva središnja člana (slika 3.15.). Svaka od koordinata spremljena je u zasebnu listu, dakle postoje dvije liste za svaki objekt; u jednu listu se spremaju isključivo *x* koordinate dok se u drugu listu spremaju *y* koordinate te se pomoću te dvije liste može izračunati mjera uspješnosti detekcije. Skripta za sortiranje liste može se pronaći na DVD-u priloženom uz ovaj rad u mapi Prilog P.3.4.



Slika 3.15. Prikaz sortiranja liste *x* i *y* koordinata

Prije računanja mjere uspješnosti detekcije, potrebno je označiti stvarne granične okvire (ovdje dolazi do izražaja ljudska pogreška – što je ona manja to će i rezultati biti točniji) što je napravljeno pomoću *open_labeling.py* skripte koja služi za prolaženje kroz sve slike na kojima je potrebno izračunati mjeru uspješnosti detekcije. Skripta za označavanje može se pronaći na DVD-u priloženom uz ovaj rad u mapi Prilog P.3.5. Nakon označavanja slika, kao izlaz se dobije *.xml* datoteka za svaku sliku, u kojoj su spremljene koordinate objekta. Takvu datoteku potrebno je parsirati i spremiti u tekstualnu datoteku kako bi je algoritam mogao pročitati. Parsiranje je izvedeno pomoću *parser.py* koji se može pronaći na DVD-u priloženom uz ovaj rad u mapi Prilog P.3.6.

3.4. Procjena udaljenosti i brzine vozila koje se nalazi u video okviru

Današnji automobili opremljeni su sustavima koji pružaju pomoć vozaču prilikom vožnje kako bi se smanjila mogućnost prometnih nezgoda. Nepažnja i kriva procjena brzine vozača, često su uzroci nesreće. Iz tog razloga, ovakav algoritam pomaže smanjenju vjerojatnosti nastanka prometne nesreće, jer zamjenjuje percepciju vozača sa senzorima i aktuatorima, koji imaju manju vjerojatnost pogreške i veću brzinu odziva. Vrlo bitnu ulogu u preciznosti predloženog algoritma imaju kamera i istrenirana kaskada. Što je brzina okvira video signala veća to je i preciznost algoritma veća. Preciznost također određuje i točnost kaskade u detektiranju vozila (vozilo može biti bliže ili dalje nego što je ono detektirano). Kako sama informacija o udaljenosti automobila ne donosi previše saznanja o detektiranom vozilu i potencijalnoj koliziji s istim, dodana je i informacija brzine vozila koje je detektirano.

3.4.1. Procjena udaljenosti vozila

Procjena udaljenosti uvelike se razlikuje od kamere do kamere stoga za implementaciju algoritma u bilo koji drugi sustav, parametri kamere moraju biti poznati. Prilikom razvoja algoritma, korištena je kamera mobilnog telefona *Samsung Galaxy S10+* te su sve trening slike kao i testni video uzorci snimljeni navedenom kamerom. Ukoliko se algoritam želi koristiti neka druga kamera, potrebno je ponovno istrenirati kaskadu jer će statistika detektiranja u tom slučaju biti znatno lošija. Kamera *Samsung Galaxy S10+* ima žarišnu duljinu kamere od 4.32 mm, a širina senzora je 7.06 mm [17, 18]. Poznavajući parametre kamere, izračun udaljenosti detektiranog vozila od kamere može se dobiti is izraza (3-5):

$$D\ [mm] = \frac{f\ [mm] * h[mm] * h_{slike}[elemenat\ slike]}{h'\ [elemenat\ slike] * h_{\text{senzor}}\ [mm]} \tag{3-5}$$

Gdje je:

- D – udaljenost objekta od kamere iskazan u milimetrima
- f - žarišna duljina leće iskazana u milimetrima
- h – visina vozila iskazana u milimetrima (stvarna visina vozila)

- h_{slike} – visina video okvira iskazana u elementima slike
- h' – visina objekta u video okviru iskazana u elementima slike
- h_{senzor} – visina senzora kamere iskazana u milimetrima

Kao što iz formule može vidjeti, u ovom slučaju se udaljenost računa pomoću više varijabli, a jedna od njih je i visina vozila. Kako se visina vozila znatno može razlikovati od vozila do vozila, potrebno je koristiti formulu (3-6) u kojoj se za izračun koristi širina automobila. Širina također može oscilirati no oscilacije su znatno manje (30-ak centimetara) te je za širinu uvrštena nekakva srednja vrijednost od 1,8 m [19].

$$D\ [mm] = \frac{f\ [mm] * \mathrm{w}[mm] * w_{slike}[elemenat\ slike]}{w'\ [elemenat\ slike] * w_{\mathrm{senzor}}\ [mm]} \tag{3-6}$$

Gdje je:

- w – širina vozila iskazana u milimetrima (stvarna visina vozila)
- w_{slike} – širina video okvira iskazana u elementima slike
- w' – širina objekta u video okviru iskazana u elementima slike
- w_{senzor} – širina senzora kamere iskazana u milimetrima

Na slici 3.17. prikazana je udaljenost detektiranog automobila od kamere na 4 različite udaljenosti (5, 10, 15 i 20 m). Plavi kvadrat označava detektirani automobil dok zeleni kvadrat označava prostor koji prati *Medianflow*. Svaka od 4 navedene udaljenosti je prethodno izmjerena kako bi postojala *ground-truth* informacija s kojom se kasnije uspoređuju vrijednosti dobivene algoritmom. Kao što se sa slike može vidjeti, postoji određena pogreška udaljenosti koja se javlja zbog nepreciznosti graničnog okvira koji je dobiven od *Viola-Jones* algoritma. Isto tako, pogreška postoji jer je za širinu vozila u formulu uvrštena prosječna vrijednost automobila od 1,80 m dok je stvarna širina automobila sa slike 1,66 m.

Slika 3.17. Prikaz 4 različite udaljenosti detektiranog automobila (5, 10, 15 i 20 m)

3.4.2. Procjena brzine vozila

Glavni dio ovog algoritma predstavlja procjena brzine detektiranog vozila. Brzina uvelike ovisi o preciznosti detekcije udaljenosti detektiranog vozila iz razloga što se brzina proračunava direktno iz dane udaljenosti. Naime, poznat je broj video okvira koje kamera generira u jedinici vremena, te pomoću te informacije vrlo jednostavno se da izračunati brzina detektiranog vozila. Naravno, kako bi brzina bila točna, brzina kojom se kreće kamere kojom se snima također mora biti poznata (u ovom slučaju ona iznosi 0 km/h jer je kamera stacionarna). Poznavajući navedene parametre, brzina se može izračunati po izrazu (3-7):

$$v\ [^{m}/_{s}] = v_o\ + \frac{D_{i-1} - D_i}{t} \tag{3-7}$$

Gdje je:

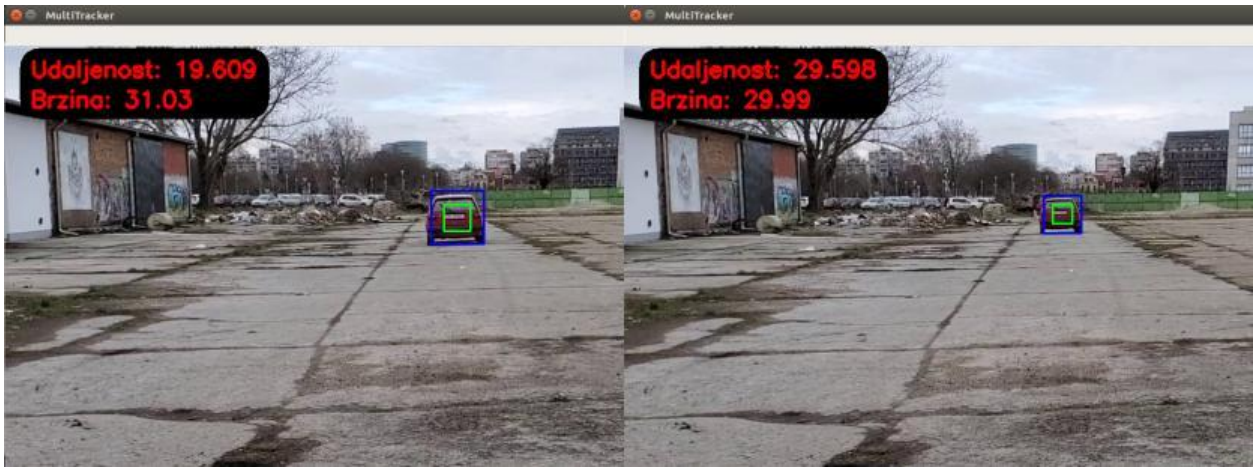
- v – izračunata brzina detektiranog vozila u metrima po sekundi
- v_o – brzina vozila na kojem je postavljena kamera iskazana u metrima po sekundi

- D_{i-1} – udaljenost detektiranog vozila u prethodnom video okviru iskazana u metrima
- D_i – udaljenost detektiranog automobila u trenutnom video okviru iskazana u metrima
- t – vrijeme između dva promatrana video okvira

Vrijeme t koje se uvrštava u jednadžbu, ovisi o parametrima kamere, točnije, ovisi o broju video okvira koje kamera snimi u jedinici vremena. Korištena kamera ima 30 FPS-a što znači da u jednoj sekundi kamera snimi 30 različitih video okvira. Algoritam računa brzinu na temelju dva susjedna video okvira (to se također može promijeniti, npr. da se brzina računa svakih 5 video okvira) što bi značilo da je potrebno izračunati vrijeme između ta dva video okvira. Ako kamera u sekundi izbaci 30 sličica, to bi značilo da svakih 33,33 ms generira po jednu sliku. Dakle, vrijeme između dva susjedna video okvira iznosi 33,33 ms te uz poznatu udaljenost detektiranog automobila u oba video okvira, određuje se prevaljeni put tako da se oduzme udaljenost prethodnog i trenutnog video okvira. Imajući te dvije informacije, algoritam izračunava brzinu. Kako bi izračunata brzina što manje varirala, algoritam koristi kružni *buffer* koji „*pegla*“ odstupanje brzine te računa mod od zadnjih 15 vrijednosti brzine (broj 15 također je određen eksperimentalno, stavi li se veći broj od 15 dolazi do velike pogreške, dok manji broj od 15 ne *„pegla“* vrijednosti dovoljno). Osvježavanje brzine se vrši svake pola sekunde dok se osvježavanje udaljenosti vrši iz video okvira u video okvir.

Na slikama 3.18. i 3.19. prikazana su dva slučaja: prvi slučaj prikazuje brzinu gibanja detektiranog vozila u iznosu od 20 km/h dok drugi slučaj prikazuje brzinu gibanja detektiranog vozila u iznosu od 30 km/h. Sa slike se vidi da postoji pogreška u proračunu brzine zbog odstupanja graničnog okvira detektiranog vozila.

Slika 3.18. Brzina kretanja detektiranog vozila iznosi 20 km/h



Slika 3.19. Brzina kretanja detektiranog vozila iznosi 30 km/h

4.TESTIRANJE I VERIFIKACIJA ALGORITMA

Pojam verifikacija podrazumijeva skup radnji pomoću kojih se nastoji utvrditi odgovara li izrađeni softver/hardver specifikacijama koje su ranije zatražene. Konačni cilj jest uvjerenje da softver/hardver radi dovoljno dobro uz određenu pouzdanost što ne znači da on nema niti jednu grešku, ali da je dovoljno dobar i pouzdan za predviđeni oblik korištenja. Verifikacija odgovara na pitanje „*Are we building the product right?*“ te uz nju postoji još i validacija koja odgovara na pitanje „*Are we building the right product*“. Navedeni pojmovi nastaju iz razloga što se specifikacijama ne uspijeva baš u potpunosti pokriti sve zahtjeve korisnika iako je to cilj, što znači da mora postojati određen kompromis između klijenta i pružatelja usluge.

Testiranje se provodi kako bi se eksperimentalno pokazalo koliko softver/hardver odstupa od traženih specifikacija, pokusno se izvodi dio ili cijeli algoritam na prikupljenim podacima te se isti analiziraju. U ovom slučaju, testiranje se vršilo na sekvenci slika koje su snimane istom kamerom koja je korištena za prikupljanje slika i trening. Nakon testiranja slijedi verifikacija kojom se potvrđuje koliko dobro algoritam odziva na određene specifikacije. Prilikom testiranja i verifikacije, prikupljene su slike sa prednje i stražnje kamere automobila u različitim uvjetima osvjetljenja, vremenskih prilika, brzina vožnje te različitim vozilima. U narednom potpoglavlju seminarskog rada mogu se vidjeti rezultati.

4.1. Klasifikacija detektiranih vozila

Detektirana vozila su klasificirana na 4 različite grupe:

- *False positive*
- *False negative*
- *True positive*
- *True negative*

gdje *false positive* označava skup slika na kojima postoji detekcija vozila dok ono u stvarnosti ne postoji, *false negative* predstavlja skup slika na kojima nema detekcije od strane algoritma no u stvarnosti postoji vozilo, *true positive* predstavlja skup slika na kojima je detektirano vozilo te to isto vozilo postoji i u stvarnosti dok *true negative* predstavlja sve slike na kojima nema ništa za detektirati niti u stvarnosti niti je algoritam išta detektirao (tablica 4.1.).

Tablica 4.1. Klasifikacija detektiranih slika

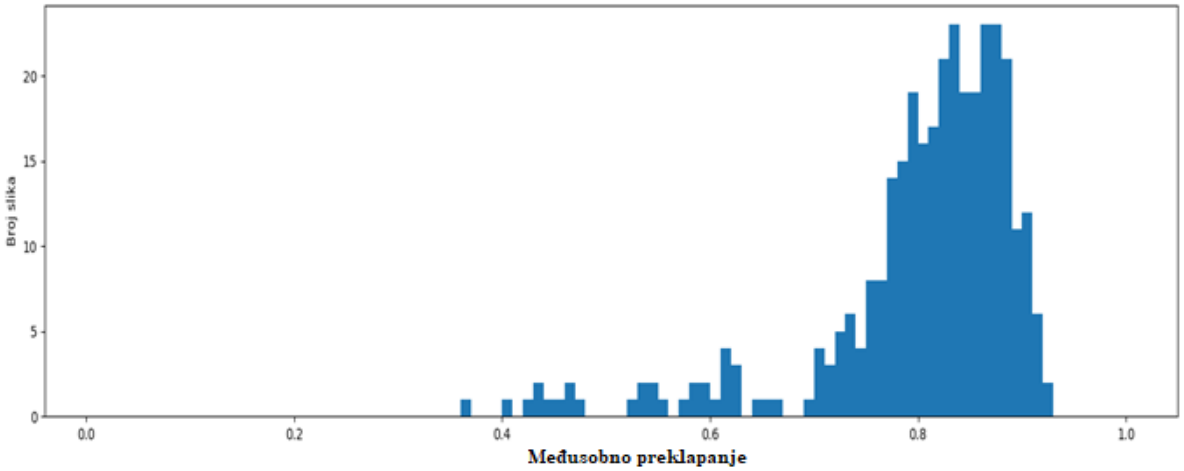
	DETEKCIJA	**STVARNOST**
False positive	+	-
False negative	-	+
True positive	+	+
True negative	-	-

4.1.1. Testiranje provedeno na video okvirima

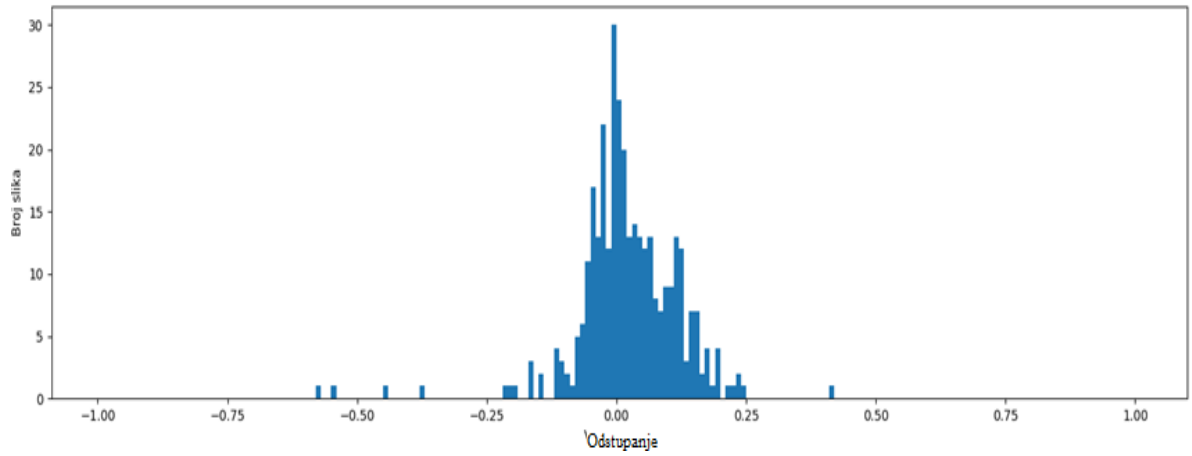
U ovom potpoglavlju napravljeno je testiranje na slikama u različitim uvjetima te su prikazane vjerojatnosti pogreške na svakom od video sekvenci. Video sekvence mogu se pronaći na DVD-u priloženom uz ovaj rad u mapi Prilog P.4.1. Također je izrađen i histogram *IoU*-a stvarnog i detektiranog graničnog okvira, te je prikazano odstupanje širine stvarnog i detektiranog graničnog okvira. Skripta pomoću koje je računato prosječno odstupanje širine stvarnog i graničnog pravokutnika nalazi se na DVD-u u mapi Prilog P.4.2.

Prednja strana sunce			
False positive	19	5%	
False negative	82	23%	
True positive	277	77%	
True negative	27	87%	
Slike sadrže auto	354		
Slike ne sadrže auto	31		
Precision	93.58%		
Recall	77.16%		
Srednji IoU	83%		
Srednje odstupanje	11%		

Tablica 4.2. Statistika za video „Prednja *strana sunce*“



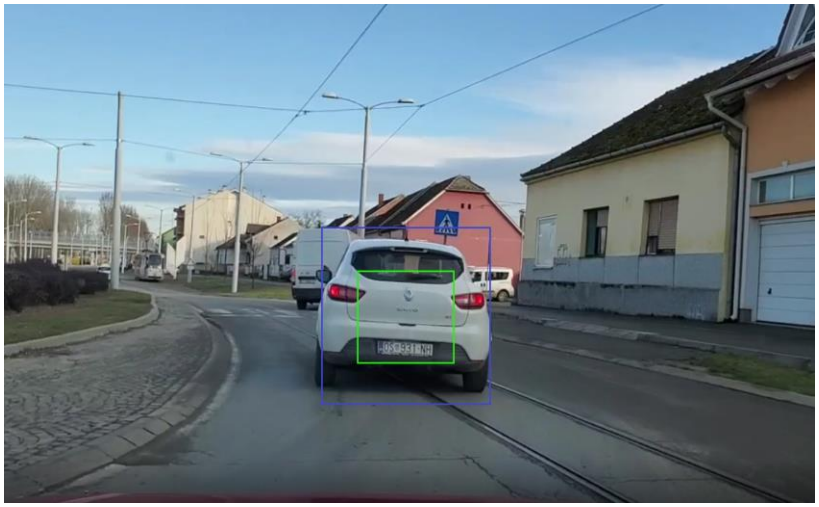
Slika 4.1. Histogram IoU stvarnog i detektiranog graničnog okvira



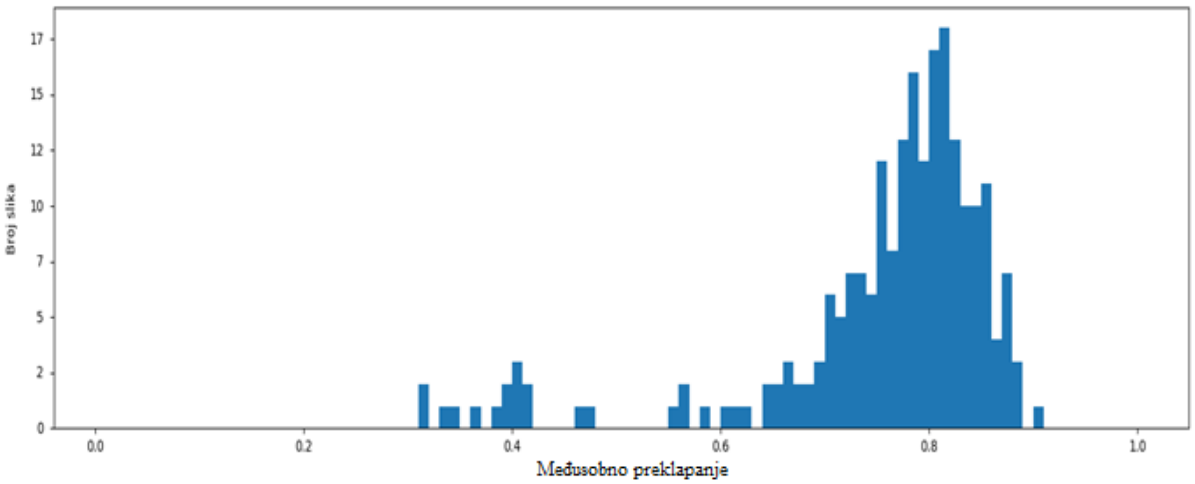
Slika 4.2. Odstupanje širine stvarnog i detektiranog graničnog okvira

Testiranjem video sekvence „*Prednja strana sunce*“ dobiven je srednji IoU od 83% te je srednje odstupanje širine stvarnog i dobivenog graničnog okvira svega 11%. Odstupanje iznosi 11% iz razloga što dobiveni granični okvir dosta „šeta“ kada je automobil koji se detektira, jako blizu automobilu na kojemu je implementiran algoritam. Iz histograma koji prikazuje IoU stvarnog i dobivenog graničnog okvira može se vidjeti kako je velika većina dobivenih graničnih okvira preko 80% međusobnog preklapanja što je dobro. Na grafu odstupanja (slika 4.2) može se vidjeti kako je svega 5 mjerenja (od ukupnih 354) odstupilo širinu stvarnog i dobivenog graničnog okvira preko ±25 % što je također dobro.

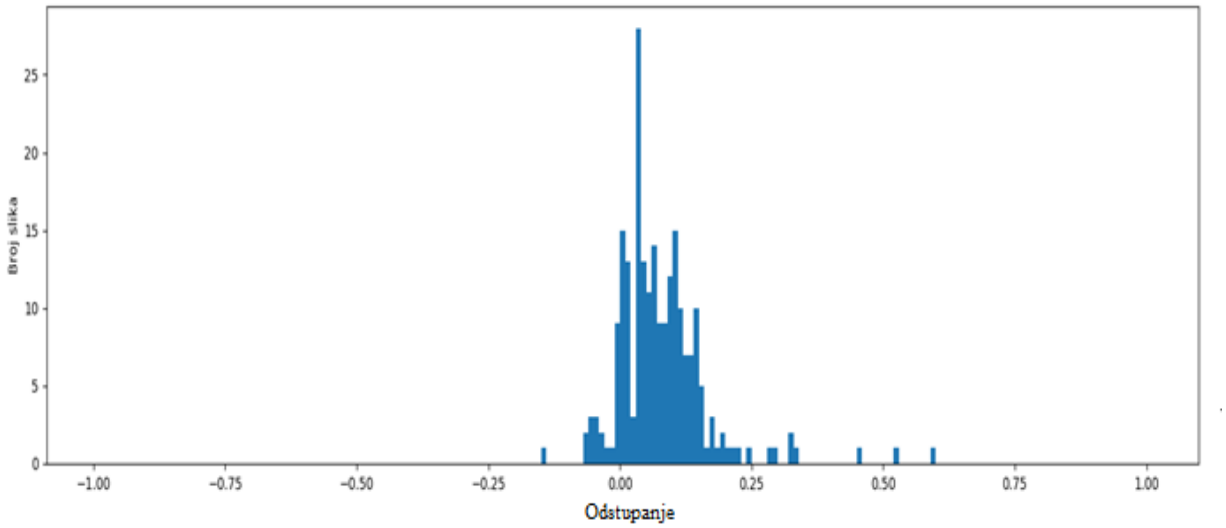
Zadnja strana sunce		
False positive	34	15%
False negative	14	6%
True positive	208	94%
True negative	27	87%
Slike sadrže auto	222	
Slike ne sadrže auto	31	
Precision	85.95%	
Recall	93.69%	
Srednji IoU	75%	
Srednje odstupanje	7%	



Tablica 4.3. Statistika za video „*Zadnja strana sunce*“



Slika 4.3. Histogram IoU stvarnog i detektiranog graničnog okvira



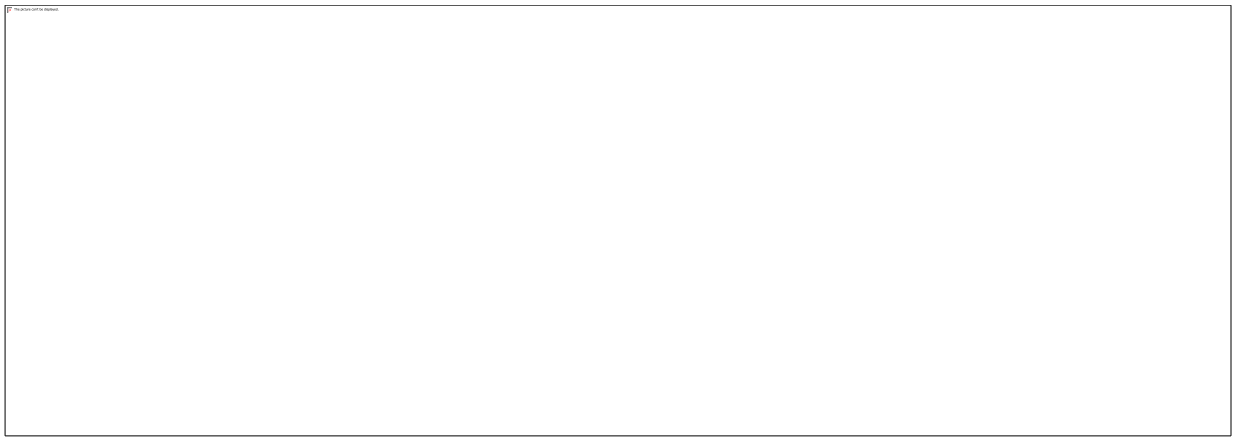
Slika 4.4. Odstupanje širine stvarnog i detektiranog graničnog okvira

Testiranjem video sekvence „*Zadnja strana sunce*“ dobiven je srednji IoU od 75% te je srednje odstupanje širine stvarnog i dobivenog graničnog okvira svega 7%. U ovom slučaju odstupanje iznosi nešto manje nego u prošlom kao i IoU koji je također manji. Odstupanje iznosi 7% iz razloga što dobiveni granični okvir dosta „šeta“ kada je automobil koji se detektira, jako blizu automobilu na kojemu je implementiran algoritam. Iz histograma koji prikazuje IoU stvarnog i dobivenog graničnog okvira može se vidjeti kako je velika većina dobivenih graničnih okvira oko 80% međusobnog preklapanja što je dobro. Na grafu odstupanja (slika 4.4) može se vidjeti kako je svega 8 mjerenja (od ukupnih 222) odstupilo širinu stvarnog i dobivenog graničnog okvira preko ±25% što je također dobro.

Zadnja strana Audi			
False positive	8	16%	
False negative	14	10%	
True positive	129	90%	
True negative	40	83%	
Slike sadrže auto	143		
Slike ne sadrže auto	48		
Precision	94,16%		
Recall	90,21%		
Srednji IoU	72%		
Srednje odstupanje	13%		

Tablica 4.4. Statistika za video „*Zadnja strana Audi*“

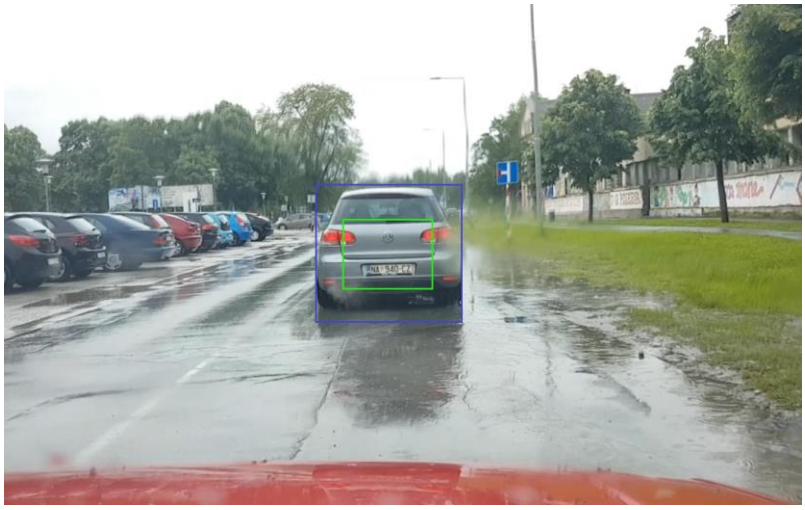
Slika 4.5. Histogram IoU stvarnog i detektiranog graničnog okvira



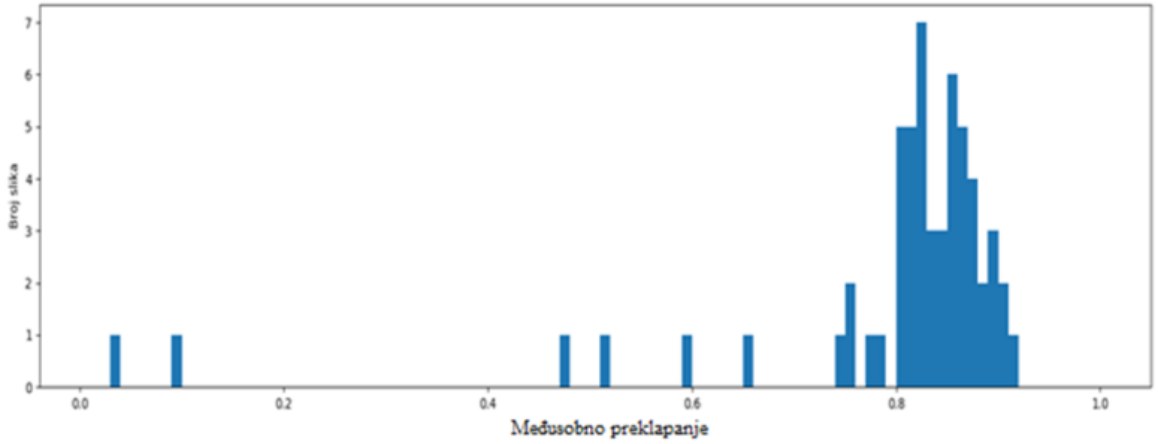
Slika 4.6. Odstupanje širine stvarnog i detektiranog graničnog okvira

Testiranjem video sekvence „*Zadnja strana Audi*“ dobiven je srednji IoU od 72% te je srednje odstupanje širine stvarnog i dobivenog graničnog okvira 13%. Odstupanje iznosi 13% iz razloga što dobiveni granični okvir dosta „šeta“ kada je automobil koji se detektira, jako blizu automobilu na kojemu je implementiran algoritam. Iz histograma koji prikazuje IoU stvarnog i dobivenog graničnog okvira može se vidjeti kako je velika većina dobivenih graničnih okvira oko 80% međusobnog preklapanja no postoje i slučajevi gdje je IoU 20-50%. Na grafu odstupanja (slika 4.6) može se vidjeti kako je svega 12 mjerenja (od ukupnih 143) odstupilo širinu stvarnog i dobivenog graničnog okvira preko ±25%.

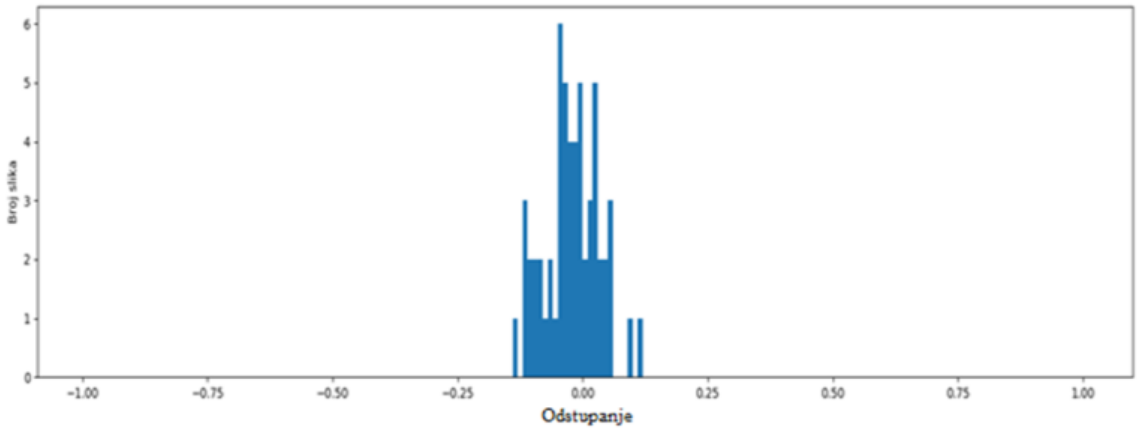
Zadnja strana kiša		
False positive	21	33%
False negative	6	9.6%
True positive	56	90.4%
True negative	0	0%
Slike sadrže auto	62	
Slike ne sadrže auto	0	
Precision	72.73%	
Recall	90.32%	
Srednji IoU	79%	
Srednje odstupanje	-3%	



Tablica 4.5. Statistika za video „*Zadnja strana kiša*“



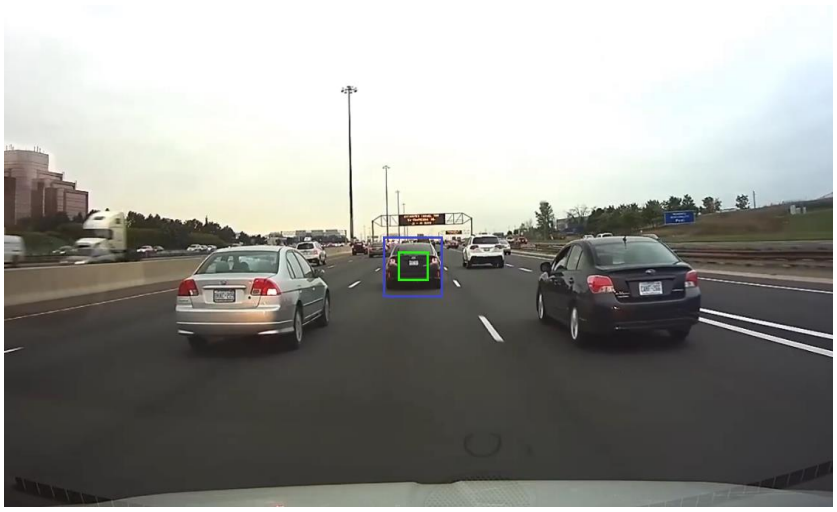
Slika 4.7. Histogram IoU stvarnog i detektiranog graničnog okvira



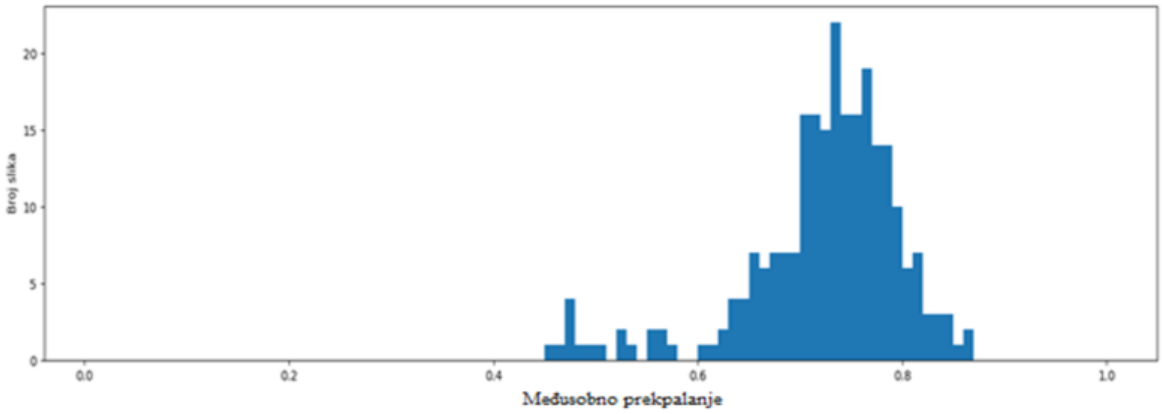
Slika 4.8. Odstupanje širine stvarnog i detektiranog graničnog okvira

Testiranjem video sekvence „*Zadnja strana kiša*“ dobiven je srednji IoU od 79% te je srednje odstupanje širine stvarnog i dobivenog graničnog okvira -3%. Odstupanje iznosi -3% iz razloga što je u cijeloj video sekvenci na kojoj se vršilo testiranje, udaljenost automobila koji se detektira jako malo varira. Iz histograma koji prikazuje IoU stvarnog i dobivenog graničnog okvira može se vidjeti kako je velika većina dobivenih graničnih okvira preko 80% međusobnog preklapanja. Najveću pogrešku prilikom ovog testiranja unosili su brisaći koji područje praćenja pomjere prilikom brisanja vjetrobranskog stakla. Na grafu odstupanja (slika 4.8) može se vidjeti kako su sva odstupanja širine ispod 20%.

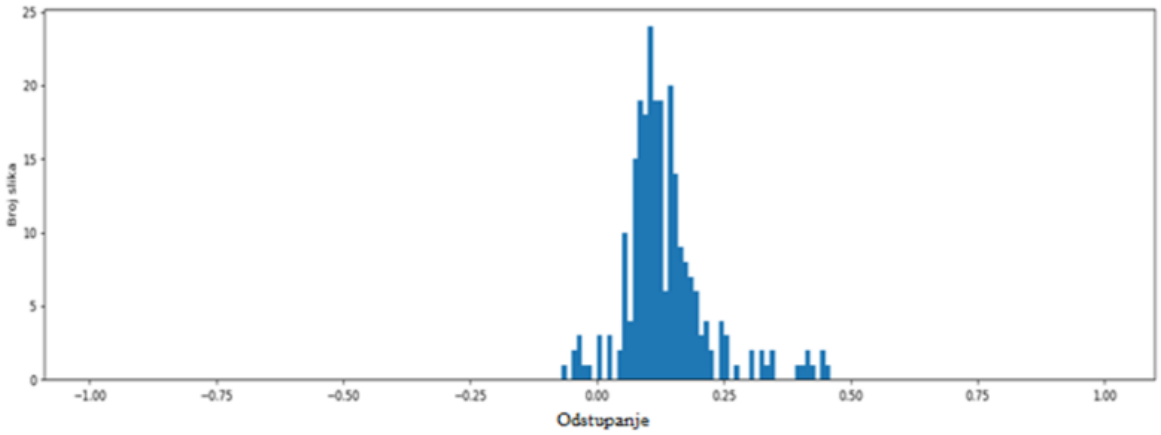
Zadnja strana video s neta		
False positive	40	16%
False negative	8	3%
True positive	244	97%
True negative	61	95%
Slike sadrže auto	251	
Slike ne sadrže auto	64	
Precision	85.92%	
Recall	96.83%	
Srednji IoU	72%	
Srednje odstupanje	13%	



Tablica 4.6. Statistika za video „*Zadnja strana video s neta*“



Slika 4.9. Histogram IoU stvarnog i detektiranog graničnog okvira

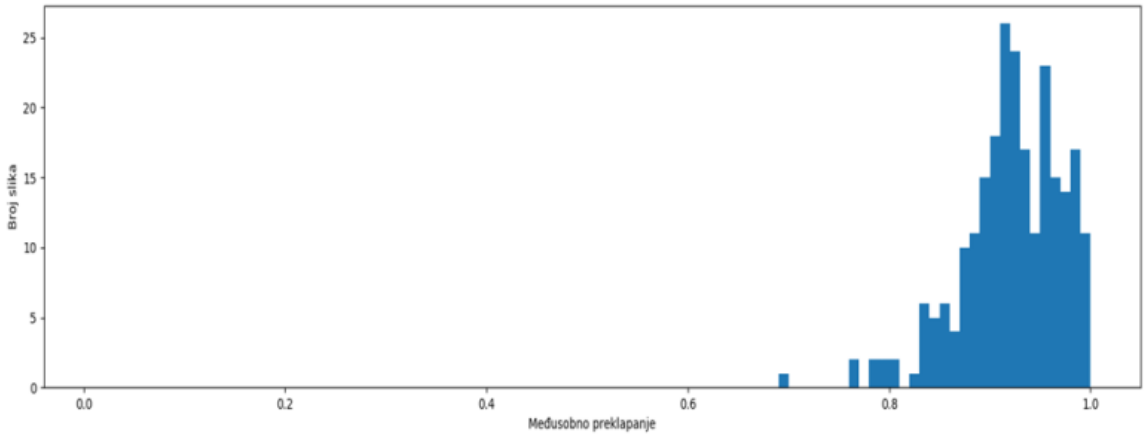


Slika 4.10. Odstupanje širine stvarnog i detektiranog graničnog okvira

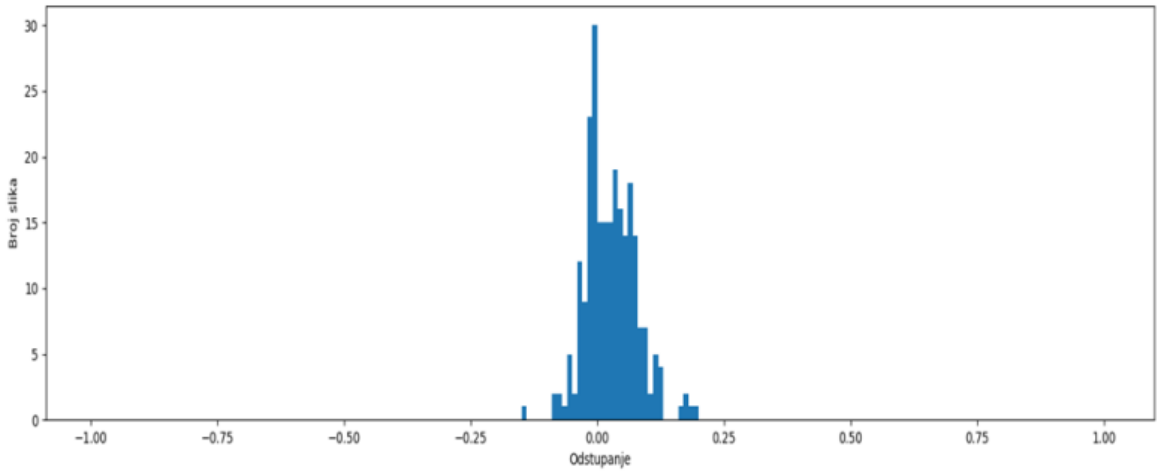
Testiranjem video sekvence „*Zadnja strana video s neta*“ dobiven je srednji IoU od 72% te je srednje odstupanje širine stvarnog i dobivenog graničnog okvira 13%. Odstupanje iznosi 13% iz razloga što dobiveni granični okvir dosta „šeta“ kada je automobil koji se detektira, jako blizu automobilu na kojemu je implementiran algoritam. Iz histograma koji prikazuje IoU stvarnog i dobivenog graničnog okvira može se vidjeti kako je velika većina dobivenih graničnih okvira nešto ispod 80% međusobnog preklapanja. Na grafu odstupanja (slika 4.10) može se vidjeti kako je većina mjerenja odstupilo širinu u pozitivnom smjeru, odnosno, stvarna širina automobila je manja od detektirane što je bolje nego kod slučaja kada je stvarna širina automobila veća od one detektirane. Rezultat krive procjene širine direktno utječe na krivu procjenu udaljenosti automobila.

Zadnja strana sunce_1		
False positive	0	0%
False negative	12	4.8%
True positive	239	95.2%
True negative	0	0%
Slike sadrže auto	251	
Slike ne sadrže auto	0	
Precision	100%	
Recall	95.2%	
Srednji IoU	92%	
Srednje odstupanje	2%	

Tablica 4.7. Statistika za video „*Zadnja strana sunce_1*“



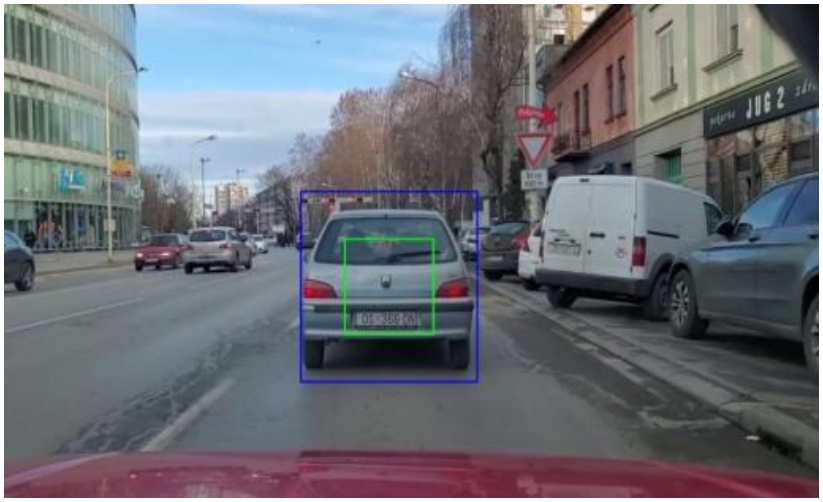
Slika 4.13. Histogram IoU stvarnog i detektiranog graničnog okvira



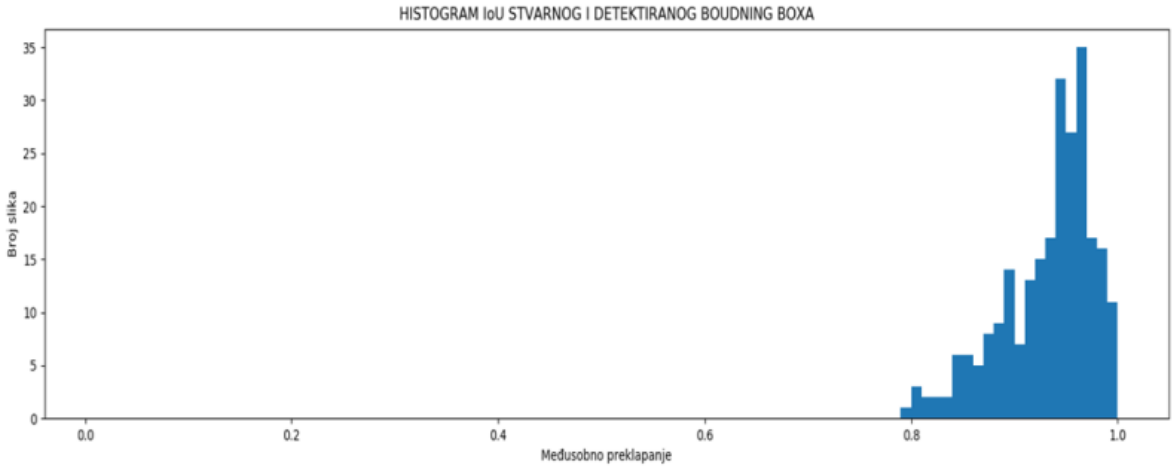
Slika 4.14. Odstupanje širine stvarnog i detektiranog graničnog okvira

Testiranjem video sekvence „*Zadnja strana sunce_1*“ dobiven je srednji IoU od 92% te je srednje odstupanje širine stvarnog i dobivenog graničnog okvira 2%. Odstupanje iznosi 2% iz razloga što je video sekvenca snimljena u izuzetno dobrom osvjetljenju i auto je dobro detektiran od strane detektora. Iz histograma koji prikazuje IoU stvarnog i dobivenog graničnog okvira može se vidjeti kako je velika većina dobivenih graničnih okvira preko 90% međusobnog preklapanja. Odstupanje širine podjednako se rasporedilo na pozitivnu (detektirani automobil veći od stvarnog) i negativnu stranu (detektirani automobil manji od stvarnog) te je zadovoljavajuće malo i iznosi ±2%.

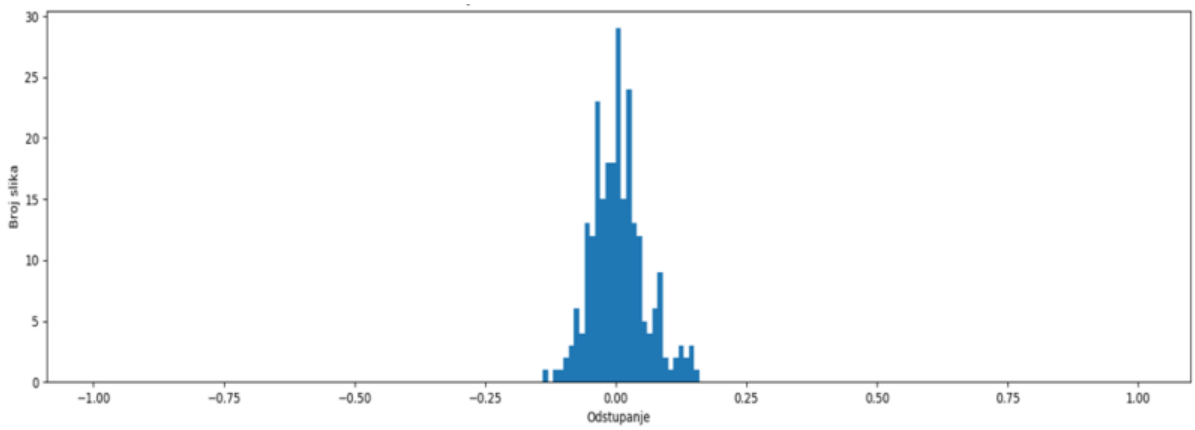
Zadnja strana sunce_2		
False positive	5	1.9%
False negative	19	7.1%
True positive	248	92.9%
True negative	0	0%
Slike sadrže auto	267	
Slike ne sadrže auto	0	
Precision	98%	
Recall	92.9%	
Srednji IoU	93%	
Srednje odstupanje	1%	



Tablica 4.8. Statistika za video „*Zadnja strana sunce_2*“



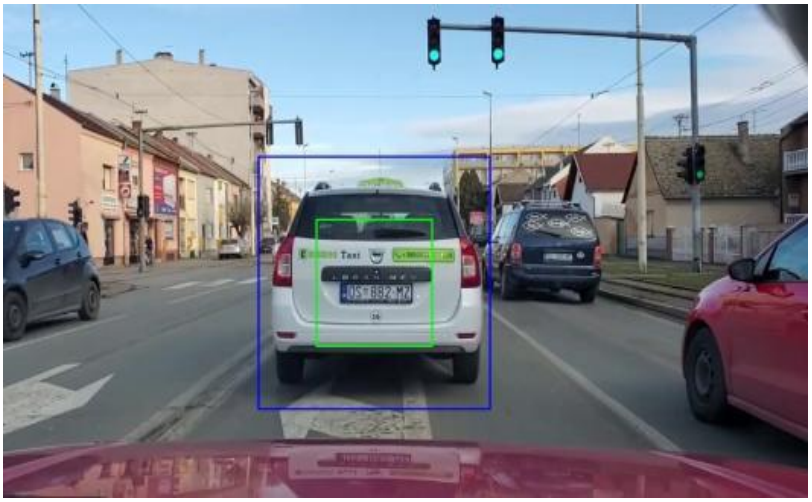
Slika 4.53. Histogram IoU stvarnog i detektiranog graničnog okvira



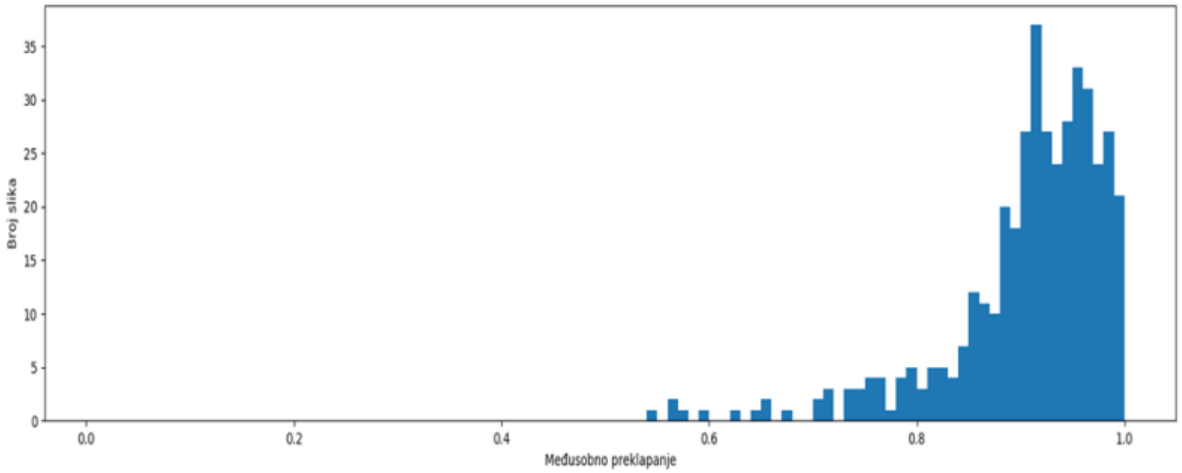
Slika 4.14. Odstupanje širine stvarnog i detektiranog graničnog okvira

Testiranjem video sekvence „*Zadnja strana sunce_2*“ dobiven je srednji IoU od 93% te je srednje odstupanje širine stvarnog i dobivenog graničnog okvira svega 1%. Odstupanje iznosi 1% iz razloga što je video sekvenca snimljena u izuzetno dobrom osvjetljenju i auto je dobro detektiran od strane detektora. Iz histograma koji prikazuje IoU stvarnog i dobivenog graničnog okvira može se vidjeti kako je samo jednom IoU manji od 80% dok su sve ostale detekcije preko 80%. Odstupanje širine podjednako se rasporedilo na pozitivnu (detektirani automobil veći od stvarnog) i negativnu stranu (detektirani automobil manji od stvarnog) te je zadovoljavajuće malo i iznosi ±1%.

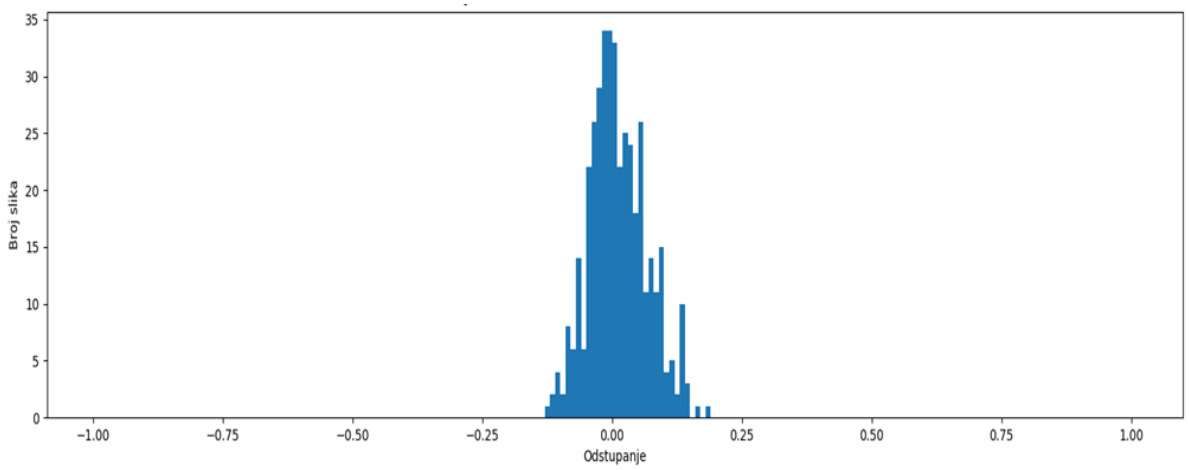
Zadnja strana sunce_3		
False positive	0	0%
False negative	10	2.4%
True positive	405	97.6%
True negative	0	0%
Slike sadrže auto	415	
Slike ne sadrže auto	3	
Precision	100%	
Recall	97.6%	
Srednji IoU	90%	
Srednje odstupanje	1%	



Tablica 4.9. Statistika za video „*Zadnja strana sunce_3*“



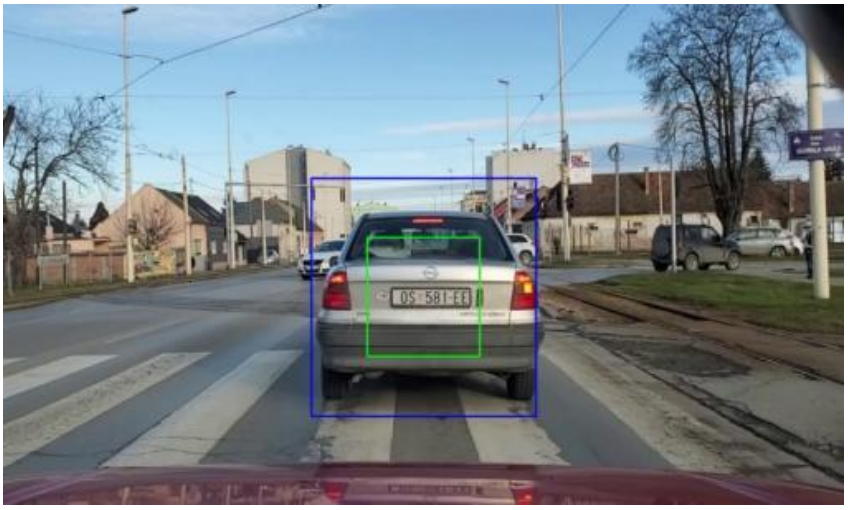
Slika 4.65. Histogram IoU stvarnog i detektiranog graničnog okvira



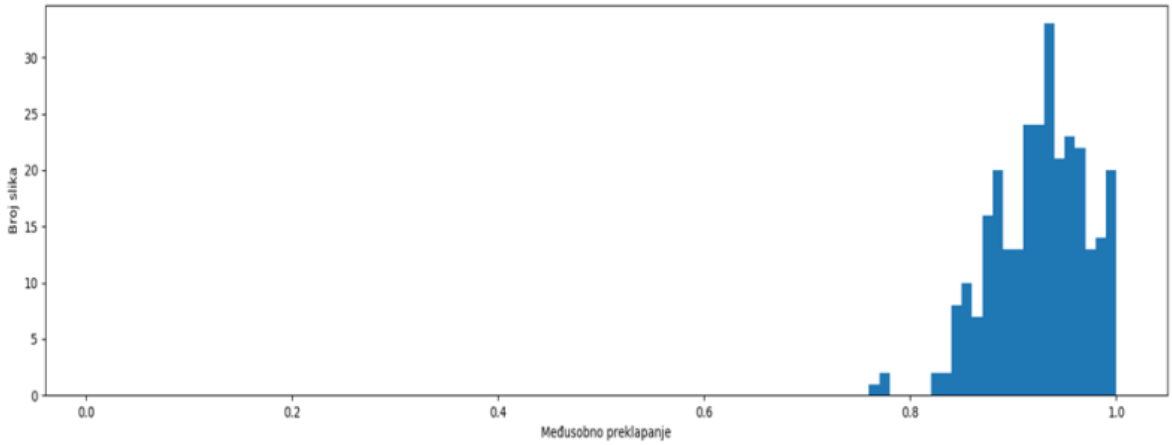
Slika 4.16. Odstupanje širine stvarnog i detektiranog graničnog okvira

Testiranjem video sekvence „*Zadnja strana sunce_3*“ dobiven je srednji IoU od 90% te je srednje odstupanje širine stvarnog i dobivenog graničnog okvira 1%. Odstupanje iznosi 1% iz razloga što je video sekvenca snimljena u izuzetno dobrom osvjetljenju i auto je dobro detektiran od strane detektora. Iz histograma koji prikazuje IoU stvarnog i dobivenog graničnog okvira može se vidjeti kako postoje detekcije koje točno označuju stvarni automobil odnosno IoU iznosi 100%. Odstupanje širine podjednako se rasporedilo na pozitivnu (detektirani automobil veći od stvarnog) i negativnu stranu (detektirani automobil manji od stvarnog) te je zadovoljavajuće malo i iznosi ±1%.

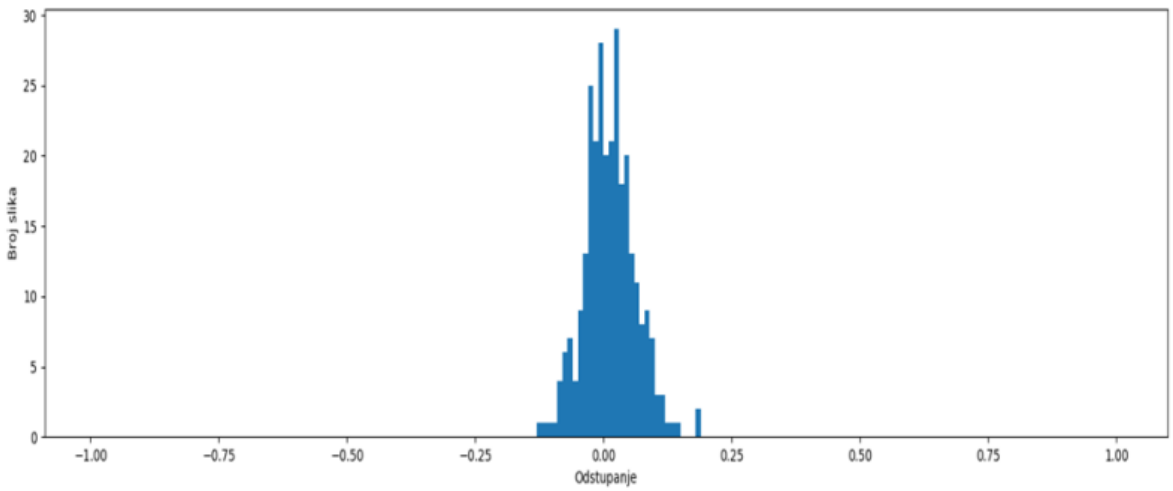
Zadnja strana sunce_4		
False positive	0	0%
False negative	6	2%
True positive	291	98%
True negative	8	100%
Slike sadrže auto	297	
Slike ne sadrže	8	
Precision	100%	
Recall	98%	
Srednji IoU	92%	
Srednje odstupanje	1%	



Tablica 4.10. Statistika za video „*Zadnja strana sunce_4*“



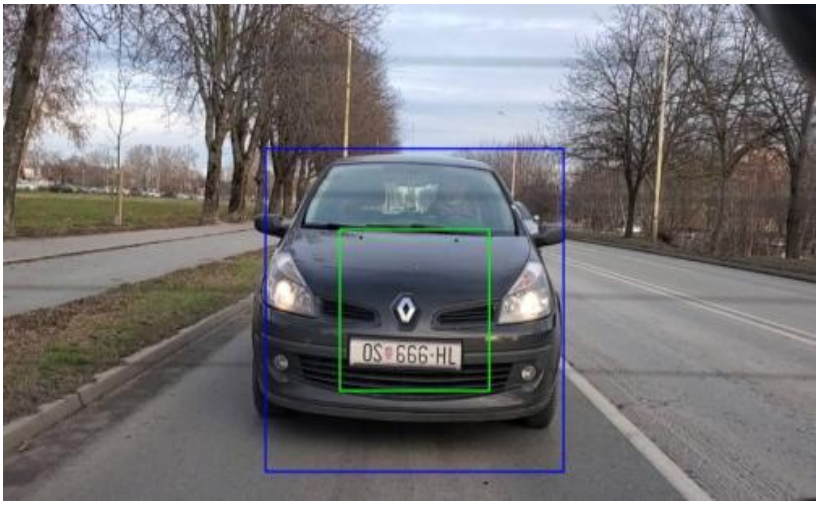
Slika 4.17. Histogram IoU stvarnog i detektiranog graničnog okvira



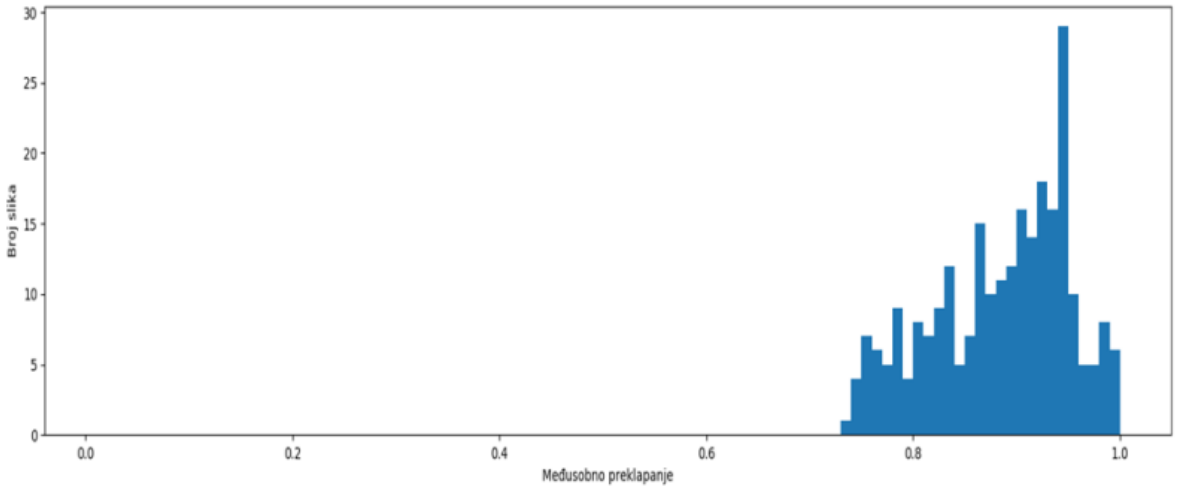
Slika 4.18. Odstupanje širine stvarnog i detektiranog graničnog okvira

Testiranjem video sekvence „*Zadnja strana sunce_4*“ dobiven je srednji IoU od 92% te je srednje odstupanje širine stvarnog i dobivenog graničnog okvira 1%. Odstupanje iznosi 1% iz razloga što je video sekvenca snimljena u izuzetno dobrom osvjetljenju i auto je dobro detektiran od strane detektora. Iz histograma koji prikazuje IoU stvarnog i dobivenog graničnog okvira može se vidjeti kako je kod 294 slike IoU između 80 i 100%. Odstupanje širine podjednako se rasporedilo na pozitivnu (detektirani automobil veći od stvarnog) i negativnu stranu (detektirani automobil manji od stvarnog) te je zadovoljavajuće malo i iznosi ±1%. Svega par detekcija odstupa preko 10% od stvarnog objekta dok je ostatak ispod 10%.

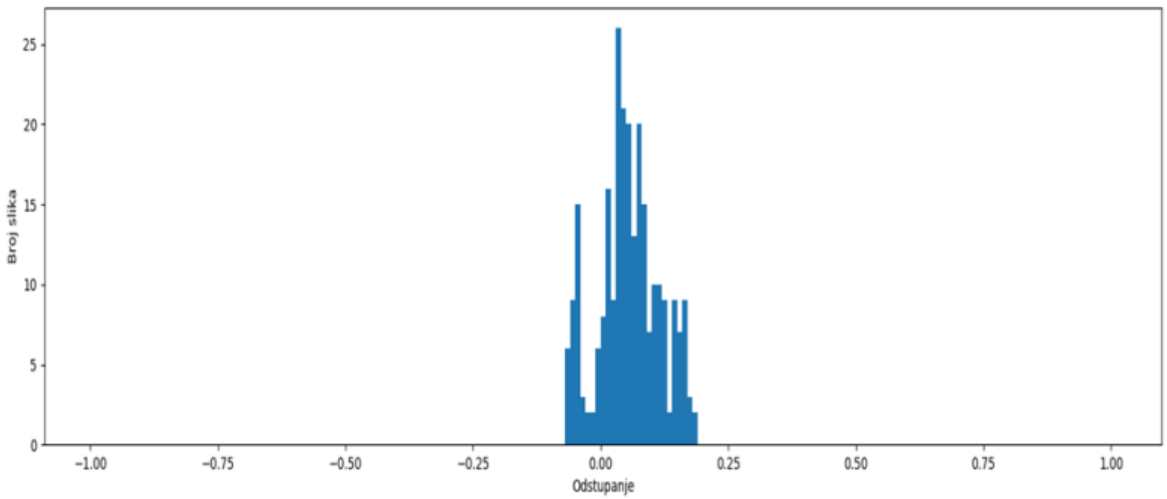
Prednja strana sunce_11		
False positive	0	0%
False negative	56	7.4%
True positive	266	82.6%
True negative	0	0%
Slike sadrže auto	322	
Slike ne sadrže auto	0	
Precision	100%	
Recall	82.6%	
Srednji IoU	88%	
Srednje odstupanje	5%	



Tablica 4.11. Statistika za video „*Prednja strana sunce_11*“



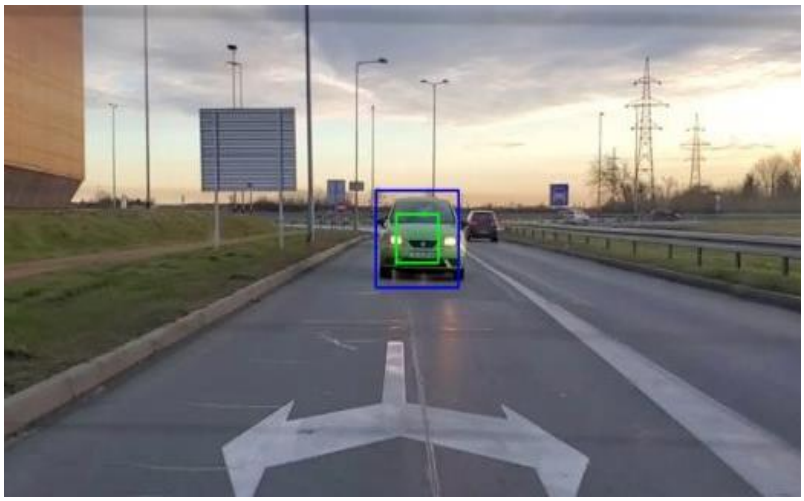
Slika 4.79. Histogram IoU stvarnog i detektiranog graničnog okvira



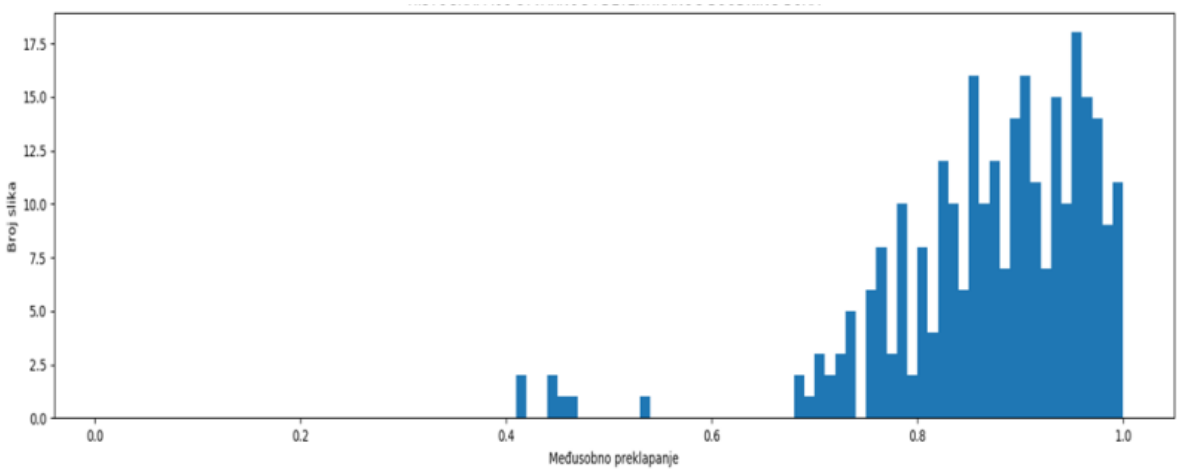
Slika 4.20. Odstupanje širine stvarnog i detektiranog graničnog okvira

Testiranjem video sekvence „*Prednja strana sunce_11*“ dobiven je srednji IoU od 88% te je srednje odstupanje širine stvarnog i dobivenog graničnog okvira 5%. Odstupanje iznosi 5% iz razloga što je video sekvenca snimljena u izuzetno dobrom osvjetljenju i auto je dobro detektiran od strane detektora. Usporede li se rezultati dobiveni detektorom prednje i stražnje strane, može se vidjeti kako je detekcija točnija kada se koristi detektor stražnje strane. Iz histograma koji prikazuje IoU stvarnog i dobivenog graničnog okvira može se vidjeti kako je detekcija u intervalu od 70-100% te nema niti jedne detekcije kod koje je IoU manji od 70%. Odstupanje širine prevagnulo je na pozitivnu stranu što znači da je detektirani automobil širi od stvarnog automobila, te je i udaljenost detektiranog manja od stvarne udaljenosti automobila.

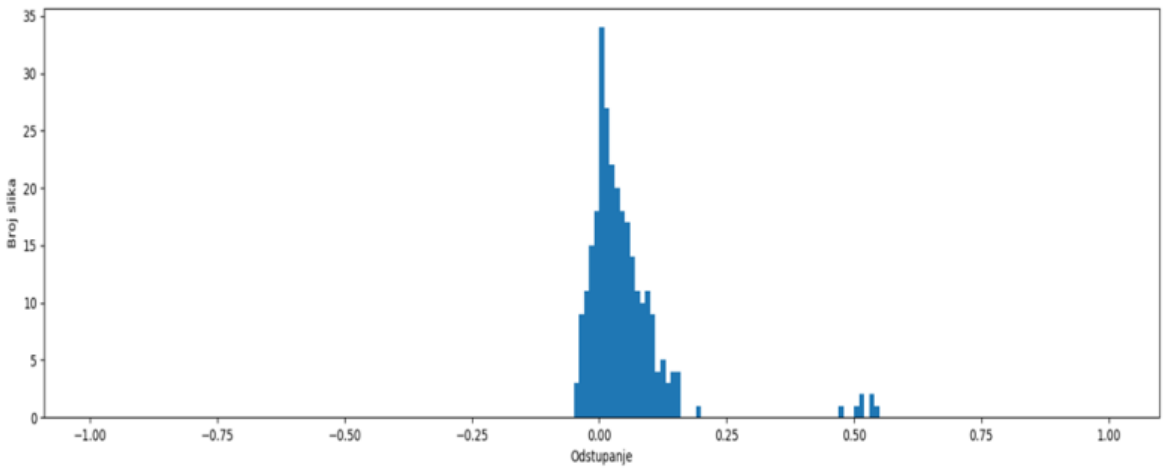
Prednja strana sunce_22		
False positive	0	0%
False negative	30	10.1%
True positive	268	89.9%
True negative	0	0%
Slike sadrže auto	298	
Slike ne sadrže auto	0	
Precision	100%	
Recall	89.9%	
Srednji IoU	87%	
Srednje odstupanje	4%	



Tablica 4.12. Statistika za video „*Prednja strana sunce_22“*



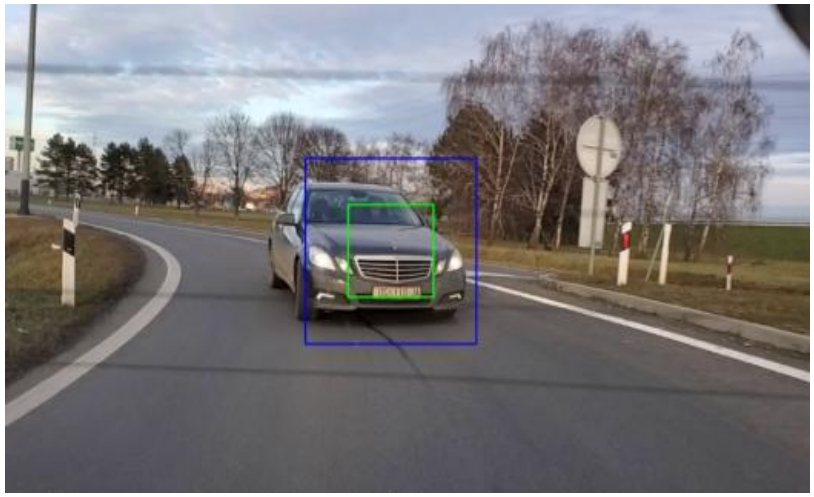
Slika 4.21. Histogram IoU stvarnog i detektiranog graničnog okvira



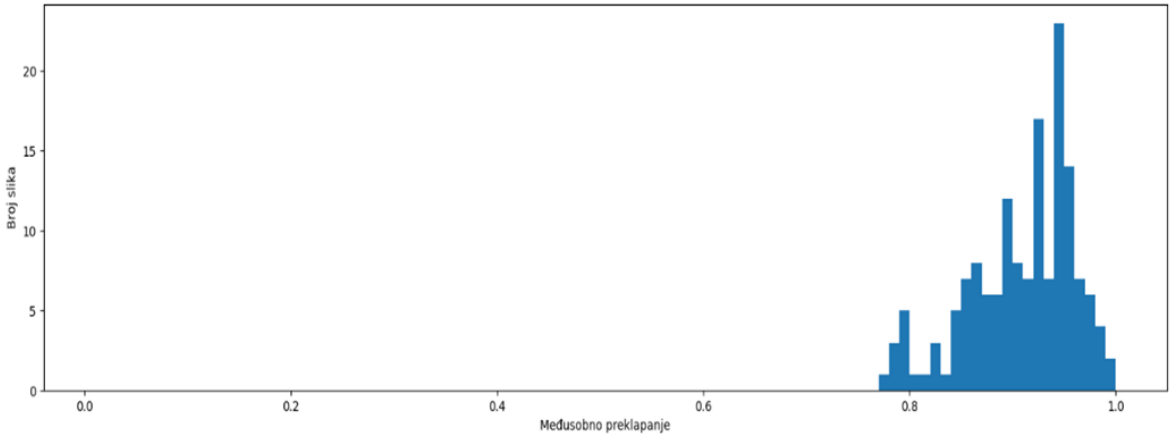
Slika 4.82. Odstupanje širine stvarnog i detektiranog graničnog okvira

Testiranjem video sekvence „*Prednja strana sunce_22*“ dobiven je srednji IoU od 87% te je srednje odstupanje širine stvarnog i dobivenog graničnog okvira 5%. Odstupanje iznosi 4% iz razloga što je video sekvenca snimljena u izuzetno dobrom osvjetljenju i auto je dobro detektiran od strane detektora. Također, usporedbom rezultata dobivenih detektorom prednje i stražnje strane, može se vidjeti kako je detekcija točnija kada se koristi detektor stražnje strane. Iz histograma koji prikazuje IoU stvarnog i dobivenog graničnog okvira može se vidjeti kako je većina detekcija u intervalu od 70-100% te je svega par detekcija kod kojih je IoU manji od 70%. Odstupanje širine prevagnulo je na pozitivnu stranu što znači da je detektirani automobil širi od stvarnog automobila, te je i udaljenost detektiranog manja od stvarne udaljenosti automobila.

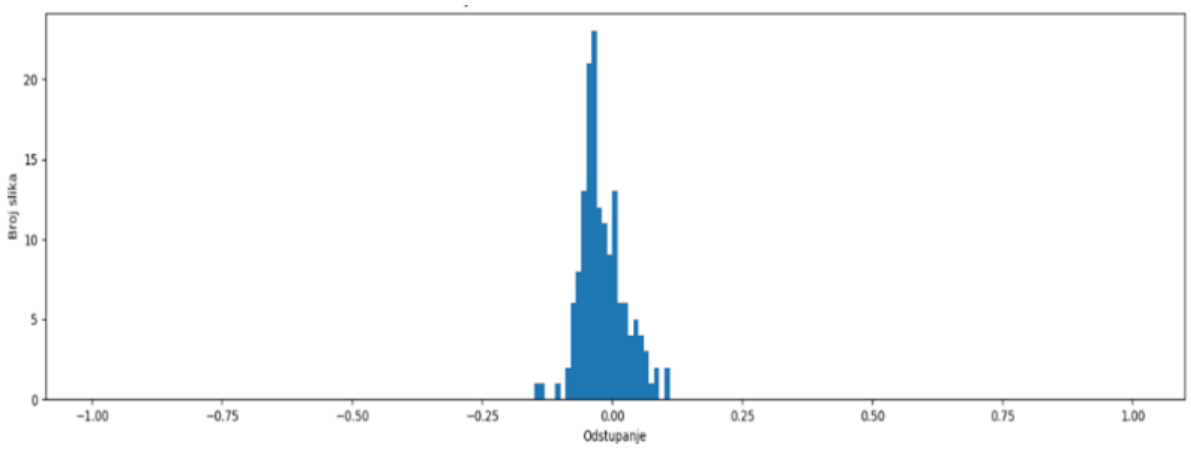
Prednja strana sunce_33		
False positive	0	0%
False negative	11	6.6%
True positive	155	93.4%
True negative	0	0%
Slike sadrže auto	166	
Slike ne sadrže auto	0	
Precision	100%	
Recall	93.4%	
Srednji IoU	90%	
Srednje odstupanje	-2%	



Tablica 4.13. Statistika za video „*Prednja strana sunce_33“*



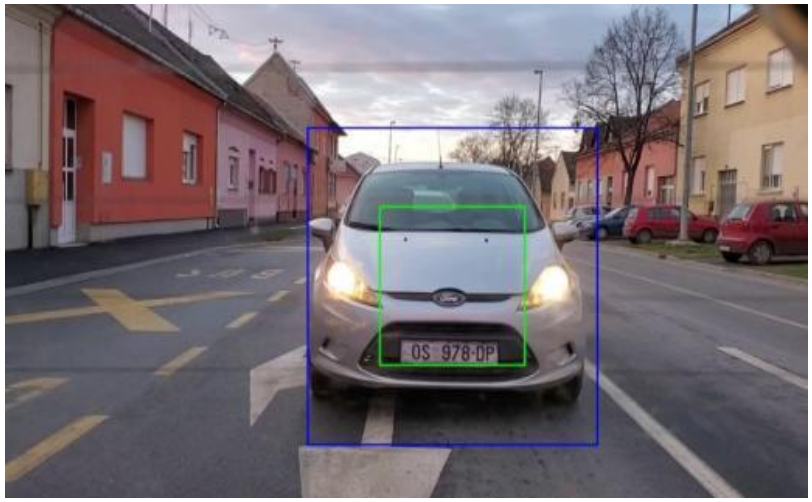
Slika 4.23. Histogram IoU stvarnog i detektiranog graničnog okvira



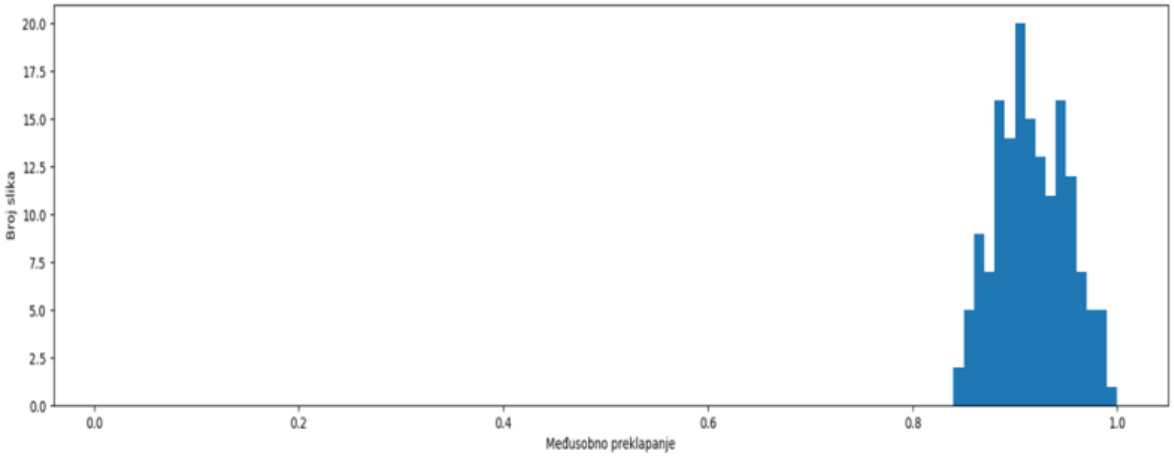
Slika 4.24. Odstupanje širine stvarnog i detektiranog graničnog okvira

Testiranjem video sekvence „*Prednja strana sunce_33*“ dobiven je srednji IoU od 90% te je srednje odstupanje širine stvarnog i dobivenog graničnog okvira -2%. Odstupanje iznosi -2% iz razloga što je video sekvenca snimljena u izuzetno dobrom osvjetljenju i auto je dobro detektiran od strane detektora. Također, usporedbom rezultata dobivenih detektorom prednje i stražnje strane, može se vidjeti kako je detekcija točnija kada se koristi detektor stražnje strane. Iz histograma koji prikazuje IoU stvarnog i dobivenog graničnog okvira može se vidjeti kako je većina detekcija u intervalu od 80-100% te je svega par detekcija kod kojih je IoU manji od 80%. Odstupanje širine prevagnulo je na negativnu stranu što znači da je detektirani automobil uži od stvarnog automobila, te je i udaljenost detektiranog veća od stvarne udaljenosti automobila što može stvarati problem prilikom određivanja poduzimanja radnje i eventualni sudar.

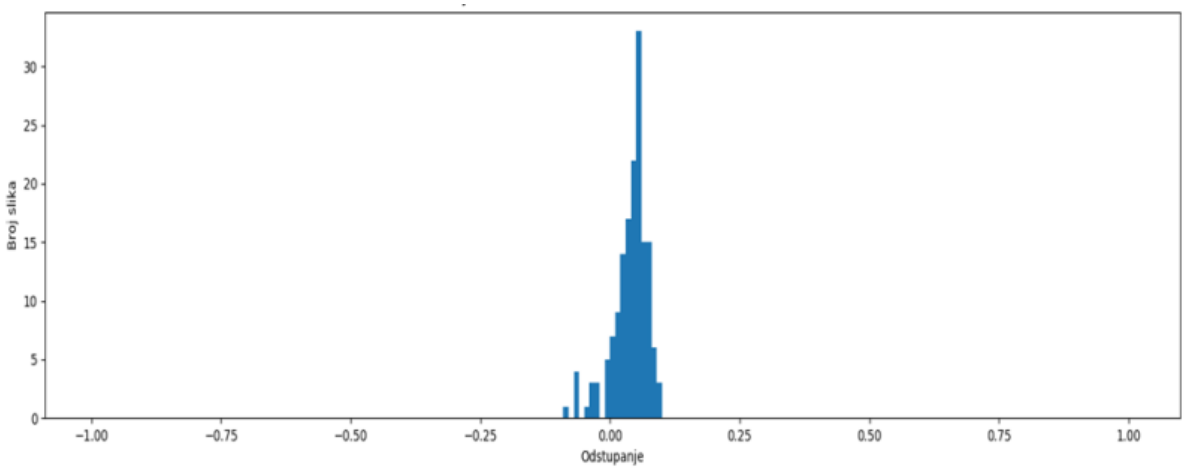
Prednja strana sunce_44		
False positive	0	0%
False negative	4	2.5%
True positive	159	97.5%
True negative	0	0%
Slike sadrže auto	163	
Slike ne sadrže auto	0	
Precision	100%	
Recall	97.5%	
Srednji IoU	91%	
Srednje odstupanje	3%	



Tablica 4.14. Statistika za video „*Prednja strana sunce_44“*

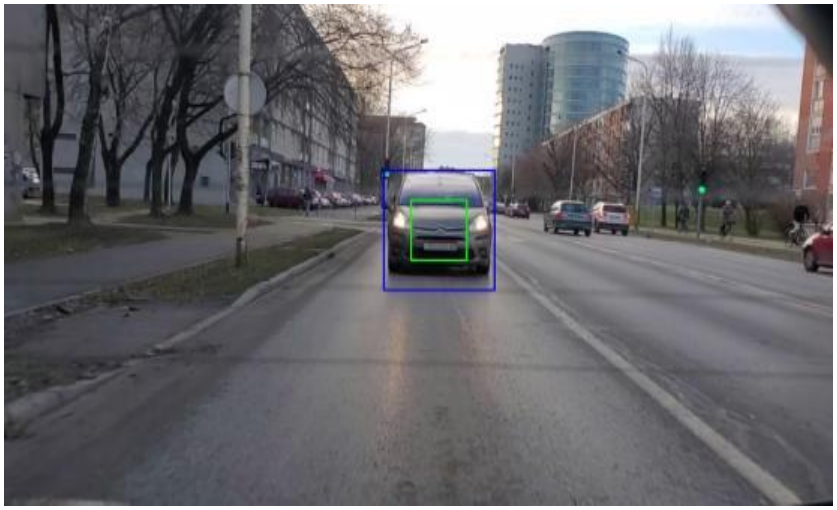


Slika 4.25. Histogram IoU stvarnog i detektiranog graničnog okvira

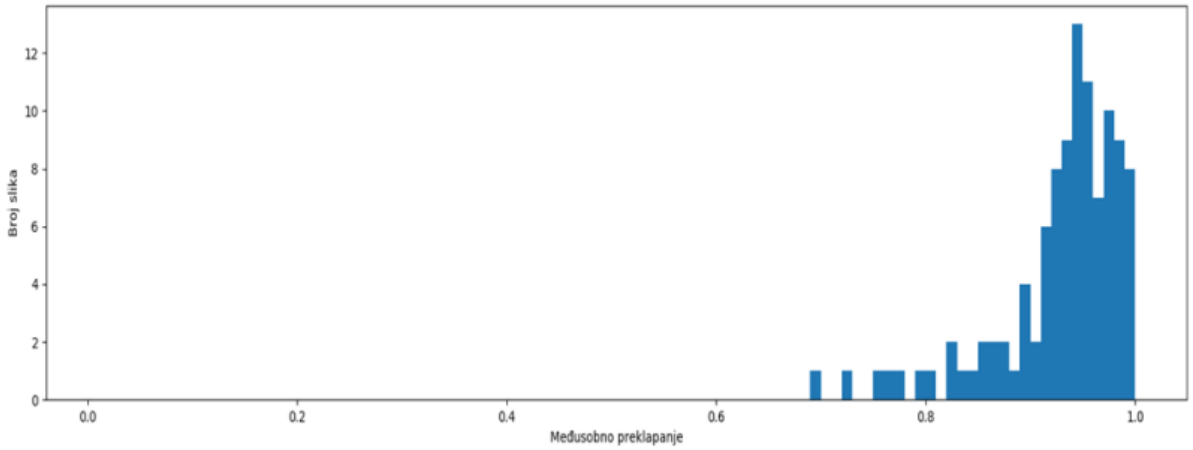


Slika 4.26. Odstupanje širine stvarnog i detektiranog graničnog okvira

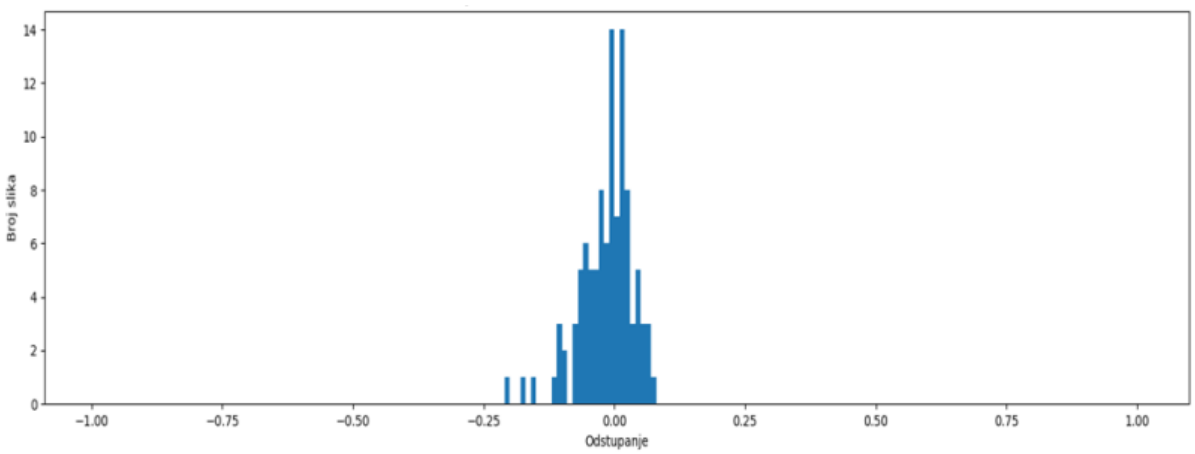
Testiranjem video sekvence „*Prednja strana sunce_44*“ dobiven je srednji IoU od 91% te je srednje odstupanje širine stvarnog i dobivenog graničnog okvira 3%. Odstupanje iznosi 3% iz razloga što je video sekvenca snimljena u izuzetno dobrom osvjetljenju i auto je dobro detektiran od strane detektora. U ovom slučaju, rezultati stražnjeg i prednjeg detektora su vrlo slični te nije nastupila dodatna pogreška prilikom testiranja. Iz histograma koji prikazuje IoU stvarnog i dobivenog graničnog okvira može se vidjeti kako je većina detekcija u intervalu od 70-100% te je svega par detekcija kod kojih je IoU manji od 70%. Odstupanje širine prevagnulo je na pozitivnu stranu što znači da je detektirani automobil širi od stvarnog automobila, te je i udaljenost detektiranog manja od stvarne udaljenosti automobila.

Prednja strana sunce_55			
False positive	0	0%	
False negative	4	3.6%	
True positive	106	96.4%	
True negative	0	0%	
Slike sadrže auto	110		
Slike ne sadrže	0		
Precision	100%		
Recall	96.4%		
Srednji IoU	92%		
Srednje odstupanje	-2%		

Tablica 4.15. Statistika za video „*Prednja strana sunce_55“*

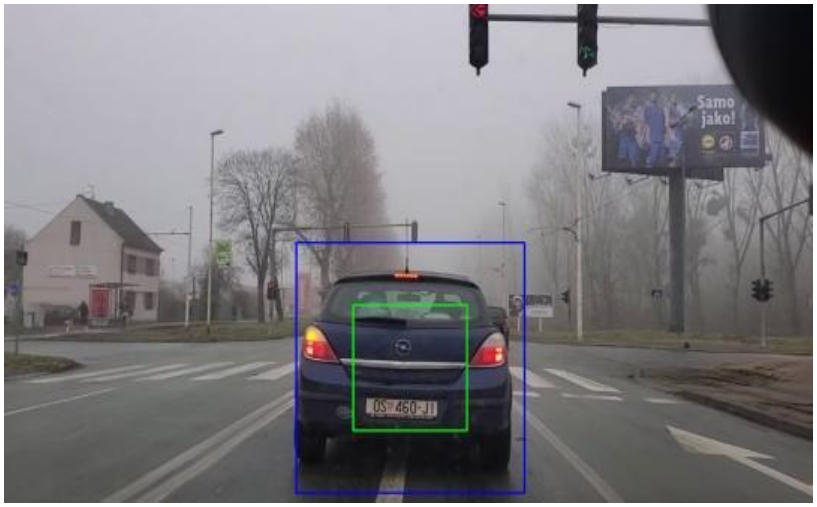


Slika 4.27. Histogram IoU stvarnog i detektiranog graničnog okvira

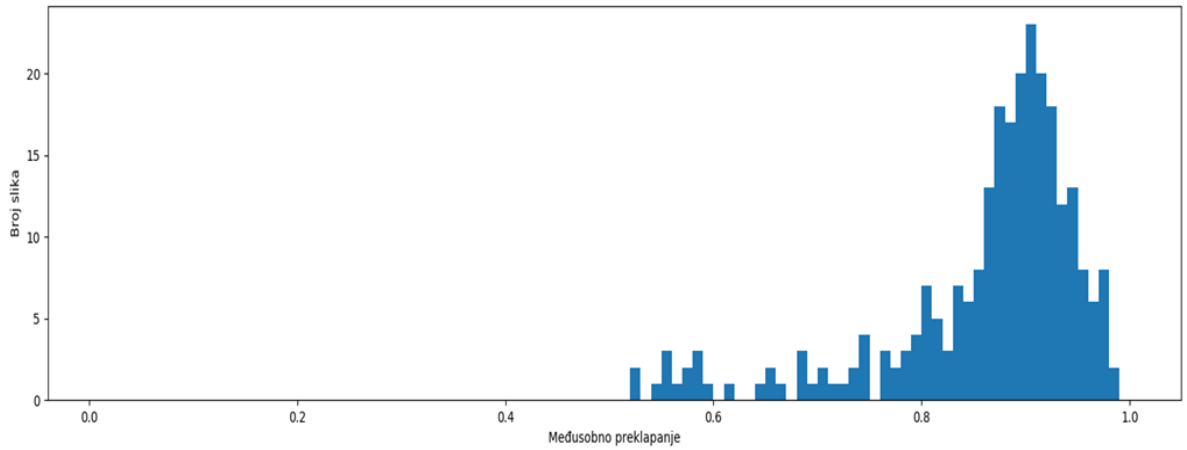


Slika 4.28. Odstupanje širine stvarnog i detektiranog graničnog okvira

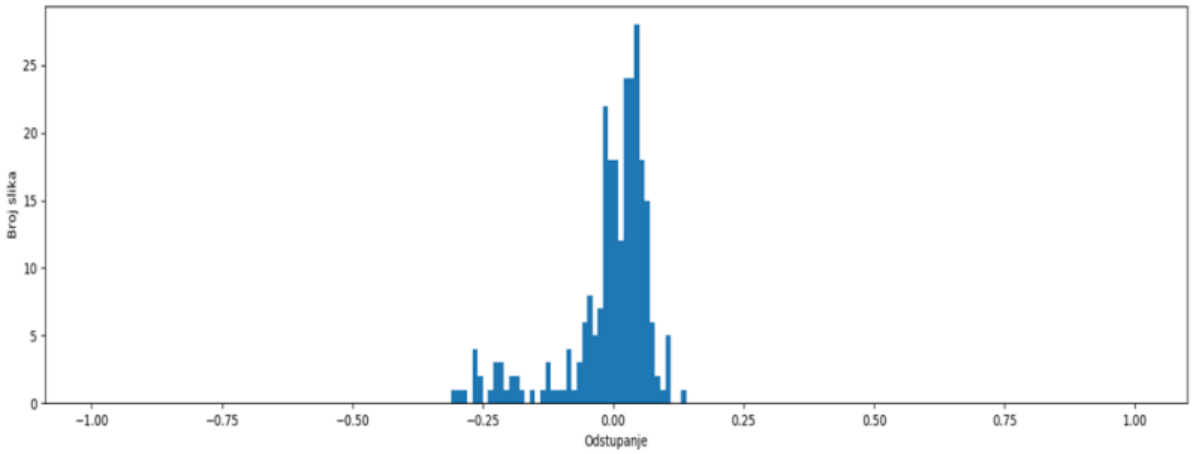
Testiranjem video sekvence „*Prednja strana sunce_55*“ dobiven je srednji IoU od 92% te je srednje odstupanje širine stvarnog i dobivenog graničnog okvira -2%. Odstupanje iznosi -2% iz razloga što je video sekvenca snimljena u izuzetno dobrom osvjetljenju i auto je dobro detektiran od strane detektora. U ovom slučaju, rezultati stražnjeg i prednjeg detektora su vrlo slični te nije nastupila dodatna pogreška prilikom testiranja. Iz histograma koji prikazuje IoU stvarnog i dobivenog graničnog okvira može se vidjeti kako je većina detekcija u intervalu od 80-100% te je svega 5 detekcija kod kojih je IoU manji od 80%. Odstupanje širine prevagnulo je na negativnu stranu što znači da je detektirani automobil uži od stvarnog automobila, te je i udaljenost detektiranog veća od stvarne udaljenosti automobila što može stvarati problem prilikom određivanja poduzimanja radnje i eventualni sudar.

Zadnja strana magla_1			
False positive	4	90%	
False negative	1	0.4%	
True positive	256	99.6%	
True negative	2	40%	
Slike sadrže auto	257		
Slike ne sadrže auto	5		
Precision	94.8%		
Recall	99.6%		
Srednji IoU	86%		
Srednje odstupanje	-1%		

Tablica 4.16. Statistika za video „*Zadnja strana magla_1*“



Slika 4.29. Histogram IoU stvarnog i detektiranog graničnog okvira

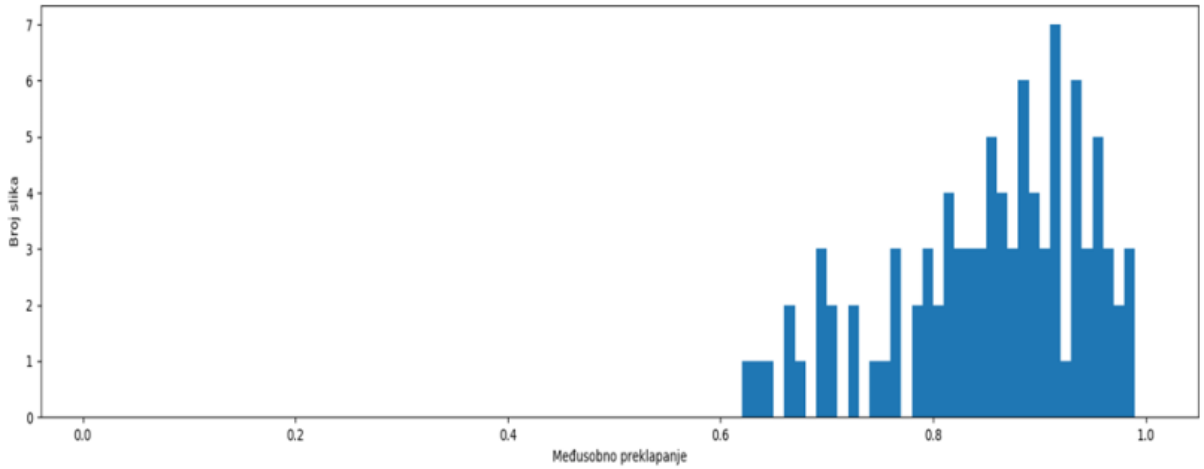


Slika 4.30. Odstupanje širine stvarnog i detektiranog graničnog okvira

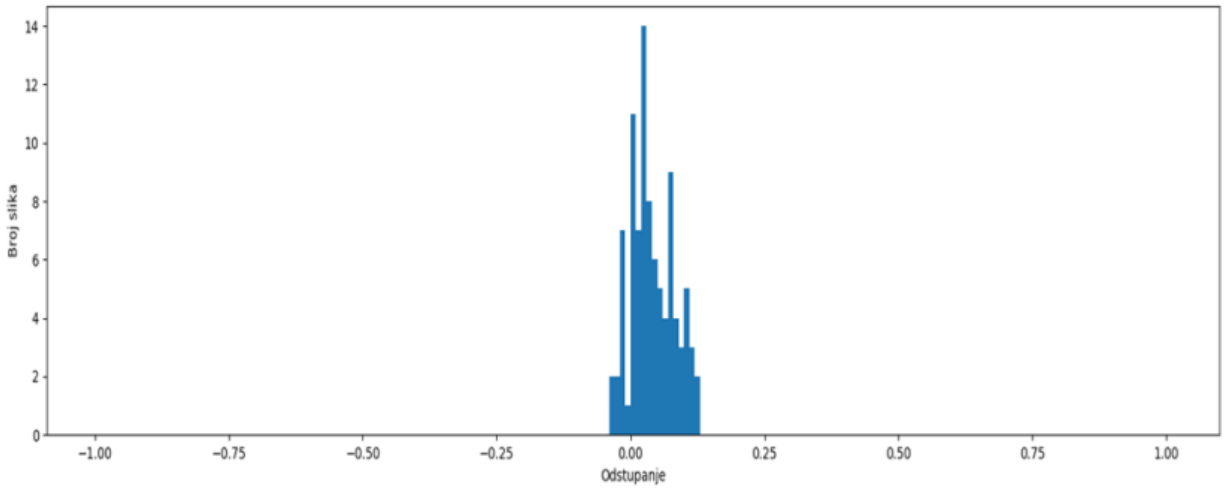
Testiranjem video sekvence „*Zadnja strana magla_1*“ dobiven je srednji IoU od 86% te je srednje odstupanje širine stvarnog i dobivenog graničnog okvira -1%. Odstupanje iznosi -1% te je vrlo dobra s obzirom da je video sekvenca snimljena u lošim uvjetima odnosno u magli. Iz histograma koji prikazuje IoU stvarnog i dobivenog graničnog okvira može se vidjeti kako je IoU poprilično razbacan iz razloga što je video sekvenca snimljena u lošim uvjetima. Odstupanje širine prevagnulo je na negativnu stranu što znači da je detektirani automobil uži od stvarnog automobila, te je i udaljenost detektiranog veća od stvarne udaljenosti automobila što može stvarati problem prilikom određivanja poduzimanja radnje i eventualni sudar.

Zadnja strana magla_2		
False positive	0	0%
False negative	2	2.1%
True positive	94	97.9%
True negative	0	0%
Slike sadrže auto	96	
Slike ne sadrže auto	0	
Precision	100%	
Recall	97.9%	
Srednji IoU	85%	
Srednje odstupanje	4%	

Tablica 4.17. Statistika za video „*Zadnja strana magla_2*“



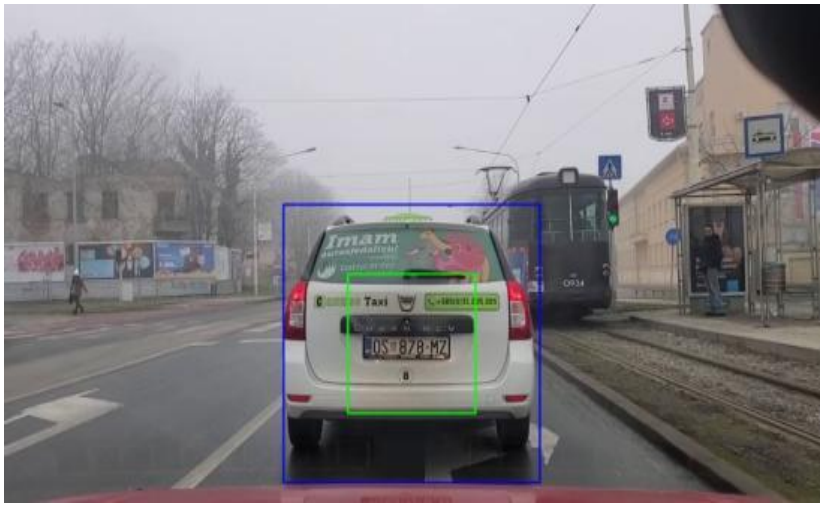
Slika 4.31. Histogram IoU stvarnog i detektiranog graničnog okvira



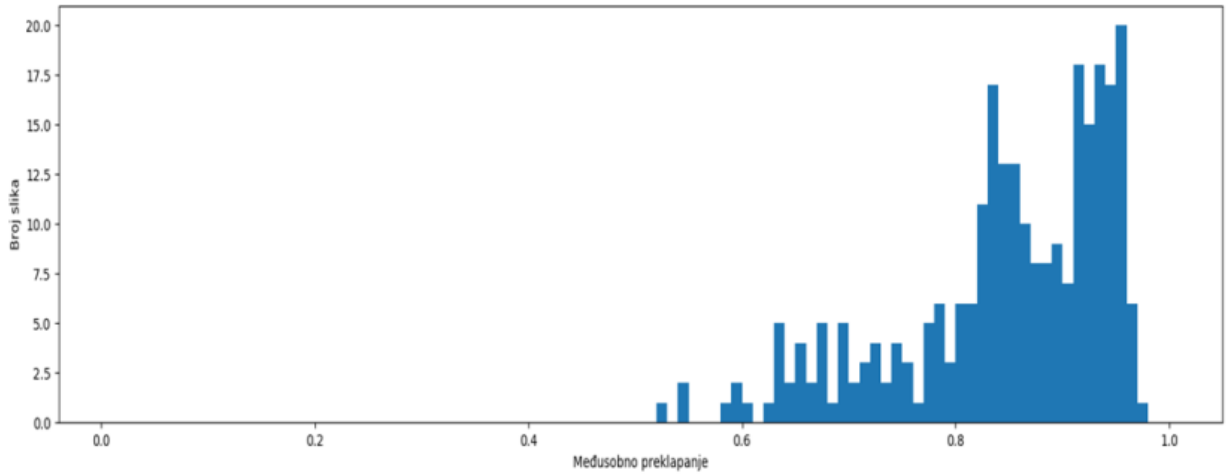
Slika 4.32. Odstupanje širine stvarnog i detektiranog graničnog okvira

Testiranjem video sekvence „*Zadnja strana magla_2*“ dobiven je srednji IoU od 85% te je srednje odstupanje širine stvarnog i dobivenog graničnog okvira 4%. Odstupanje iznosi 4% te je vrlo dobra s obzirom da je video sekvenca snimljena u lošim uvjetima odnosno u magli. Iz histograma koji prikazuje IoU stvarnog i dobivenog graničnog okvira može se vidjeti kako je većina IoU u intervalu od 80-100% što je dobro. Odstupanje širine prevagnulo je na pozitivnu stranu što znači da je detektirani automobil širi od stvarnog automobila, te je i udaljenost detektiranog manja od stvarne udaljenosti automobila. Odstupanje širine stvarnog i graničnog okvira unutar ±15%.

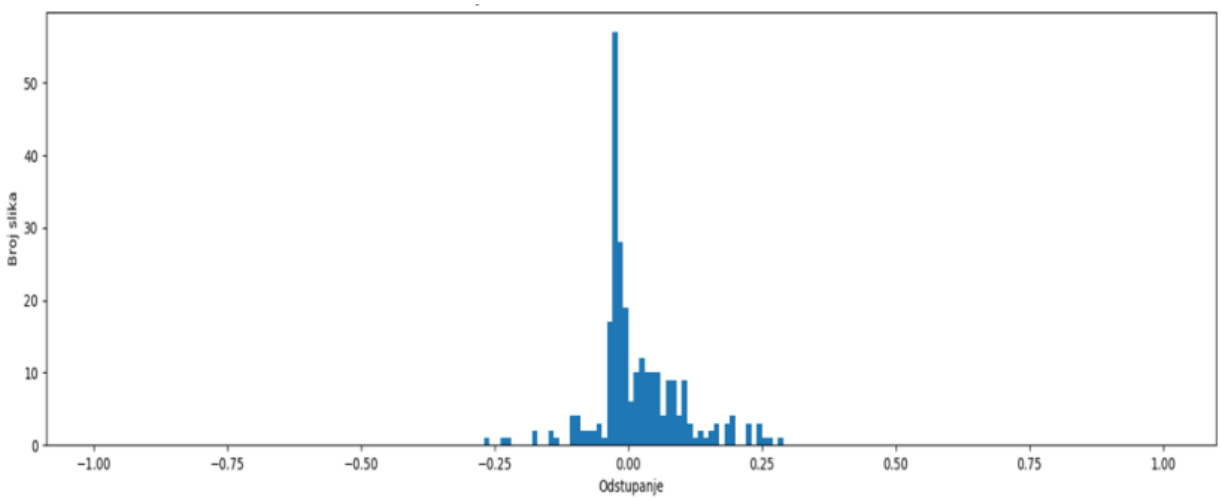
Zadnja strana magla_3		
False positive	0	0%
False negative	2	0.7%
True positive	270	99.3%
True negative	0	0%
Slike sadrže auto	272	
Slike ne sadrže auto	0	
Precision	100%	
Recall	99.3%	
Srednji IoU	84%	
Srednje odstupanje	1%	



Tablica 4.18. Statistika za video „*Zadnja strana magla_3*“



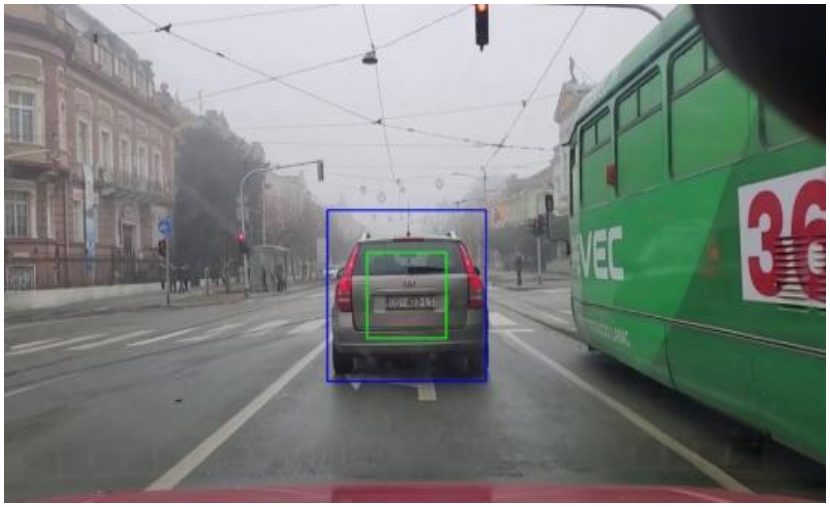
Slika 4.33. Histogram IoU stvarnog i detektiranog graničnog okvira



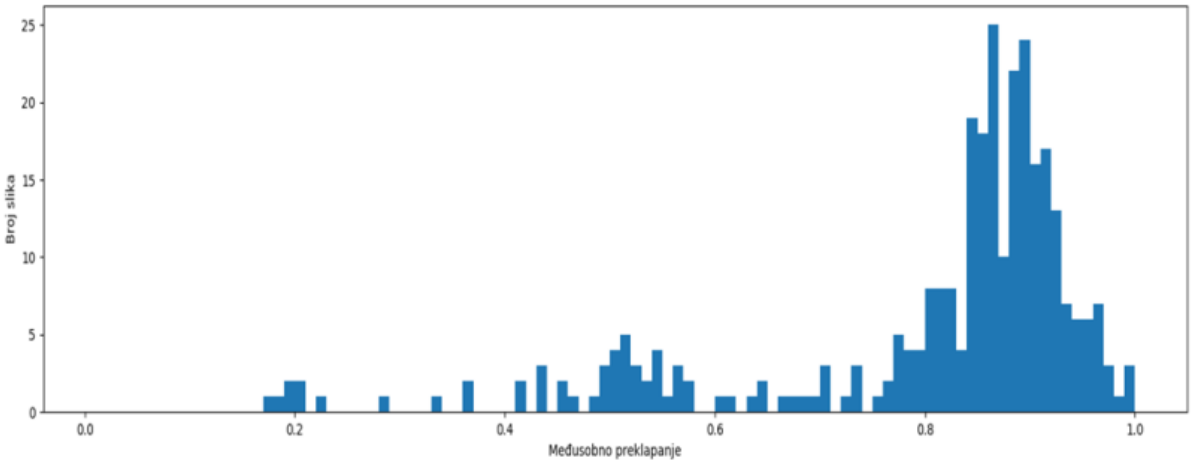
Slika 4.34. Odstupanje širine stvarnog i detektiranog graničnog okvira

Testiranjem video sekvence „*Zadnja strana magla_3*“ dobiven je srednji IoU od 84% te je srednje odstupanje širine stvarnog i dobivenog graničnog okvira 1%. Odstupanje iznosi 1% te je vrlo dobra s obzirom da je video sekvenca snimljena u lošim uvjetima odnosno u magli. Iz histograma koji prikazuje IoU stvarnog i dobivenog graničnog okvira može se vidjeti kako je IoU razbacan u intervalu od 60-100% no većina je preko 80% što je dobro. Odstupanje širine prevagnulo je na pozitivnu stranu što znači da je detektirani automobil širi od stvarnog automobila, te je i udaljenost detektiranog manja od stvarne udaljenosti automobila

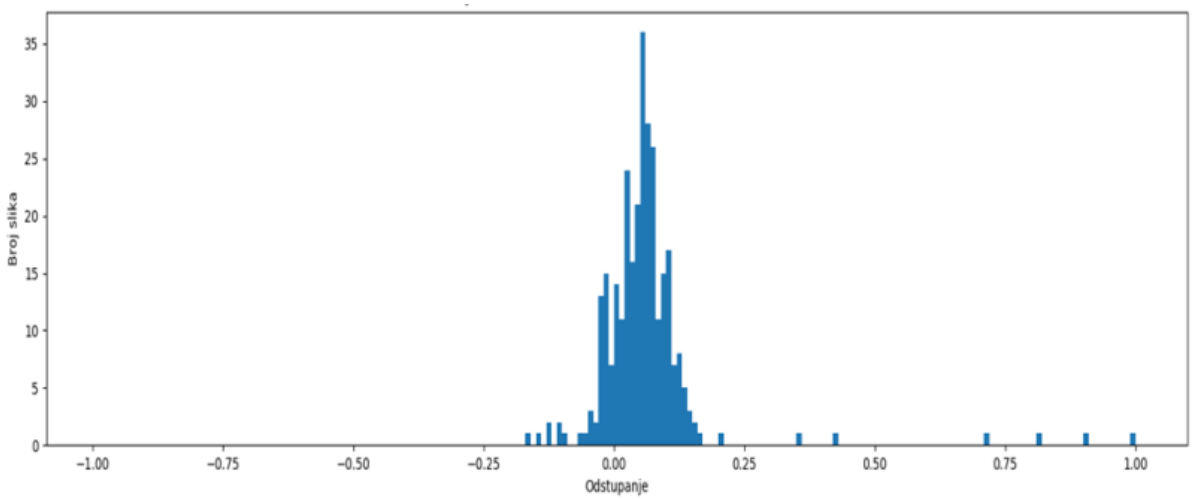
Zadnja strana magla_4		
False positive	10	3.3%
False negative	9	2.9%
True positive	298	97.1%
True negative	8	2.6%
Slike sadrže auto	307	
Slike ne sadrže auto	0	
Precision	96.7%	
Recall	97%	
Srednji IoU	80%	
Srednje odstupanje	7%	



Tablica 4.19. Statistika za video „*Zadnja strana magla_4*“



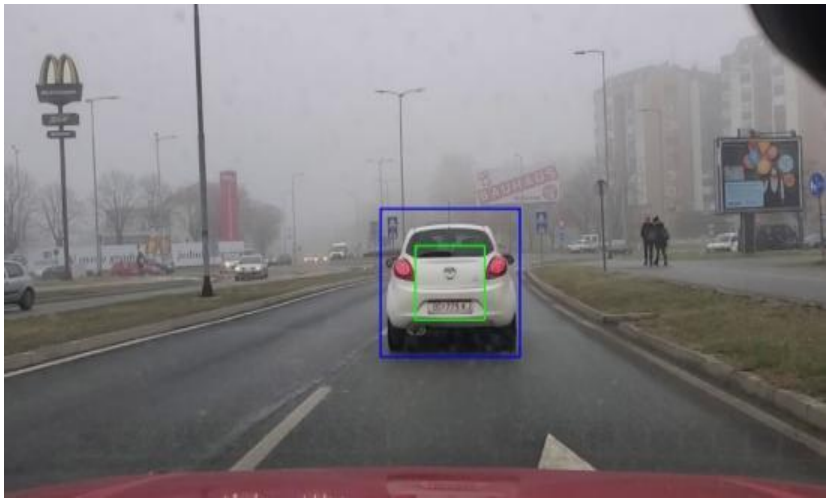
Slika 4.35. Histogram IoU stvarnog i detektiranog graničnog okvira



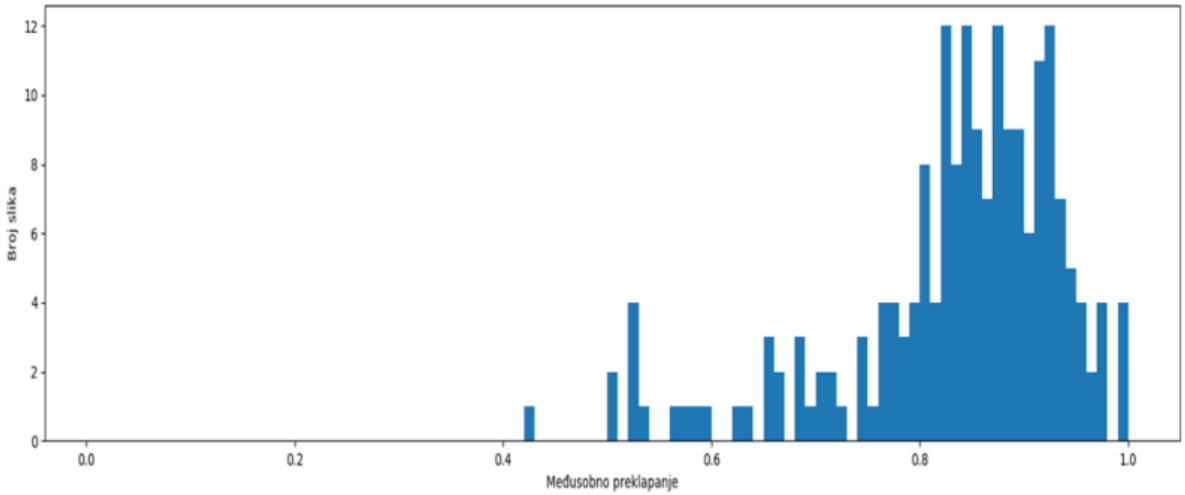
Slika 4.36. Odstupanje širine stvarnog i detektiranog graničnog okvira

Testiranjem video sekvence „*Zadnja strana magla_4*“ dobiven je srednji IoU od 80% te je srednje odstupanje širine stvarnog i dobivenog graničnog okvira 7%. Odstupanje iznosi 7% te je vrlo dobra s obzirom da je video sekvenca snimljena u lošim uvjetima odnosno u magli no lošija je usporedimo li prethodna testiranja u maglovitim uvjetima. Histogram ove video sekvence je poprilično razbacan te bi u dosta video okvira došlo do velike pogreške. Odstupanje širine prevagnulo je na pozitivnu stranu što znači da je detektirani automobil širi od stvarnog automobila, te je i udaljenost detektiranog manja od stvarne udaljenosti automobila. Odstupanje širine stvarnog i graničnog okvira u jednom trenutku iznosi čak 100% što bi rezultiralo jako velikom pogreškom u određivanju udaljenosti a samim time izazvao bi se i sudar.

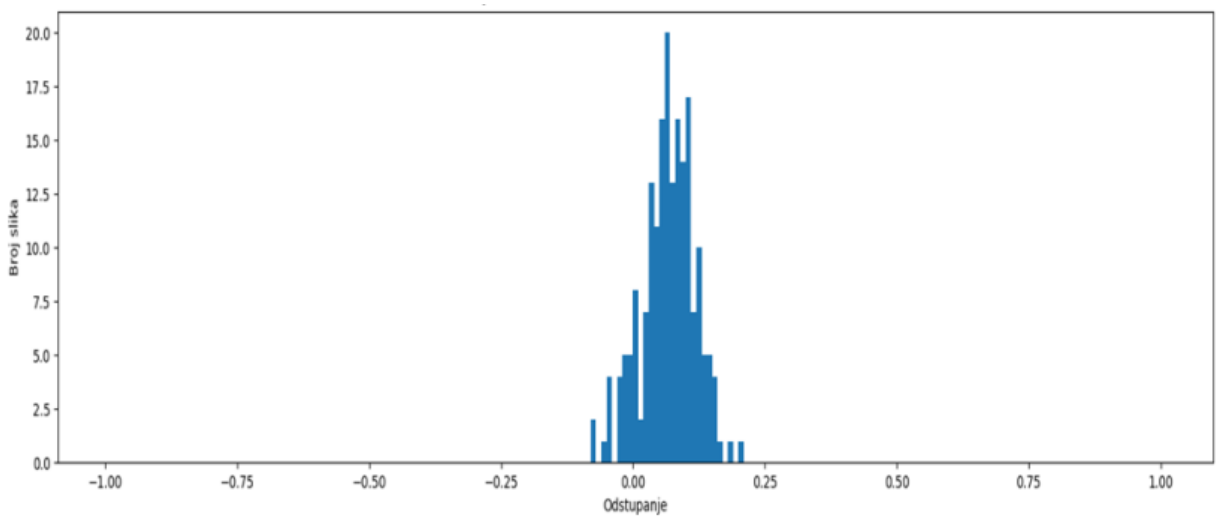
Zadnja strana magla_5		
False positive	3	13%
False negative	10	4%
True positive	192	77.8%
True negative	23	100%
Slike sadrže auto	224	
Slike ne sadrže auto	23	
Precision	98.5%	
Recall	95%	
Srednji IoU	83%	
Srednje odstupanje	6%	



Tablica 4.20. Statistika za video „*Zadnja strana magla_5*“



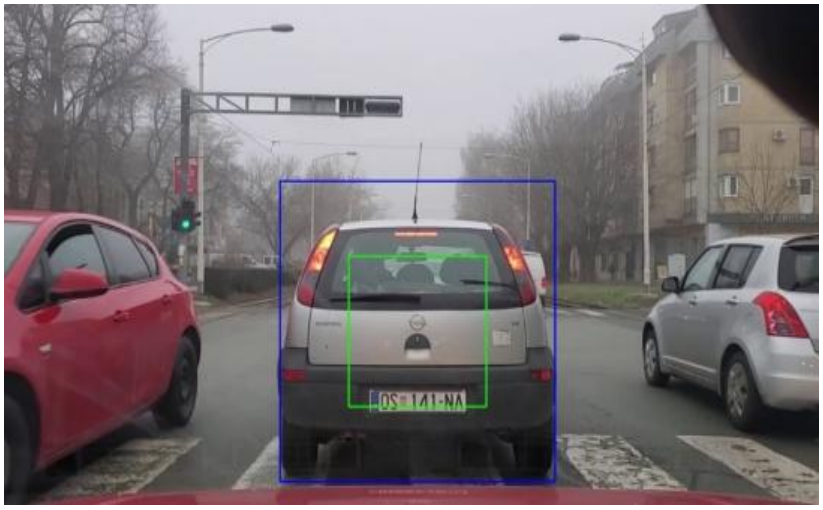
Slika 4.37. Histogram IoU stvarnog i detektiranog graničnog okvira



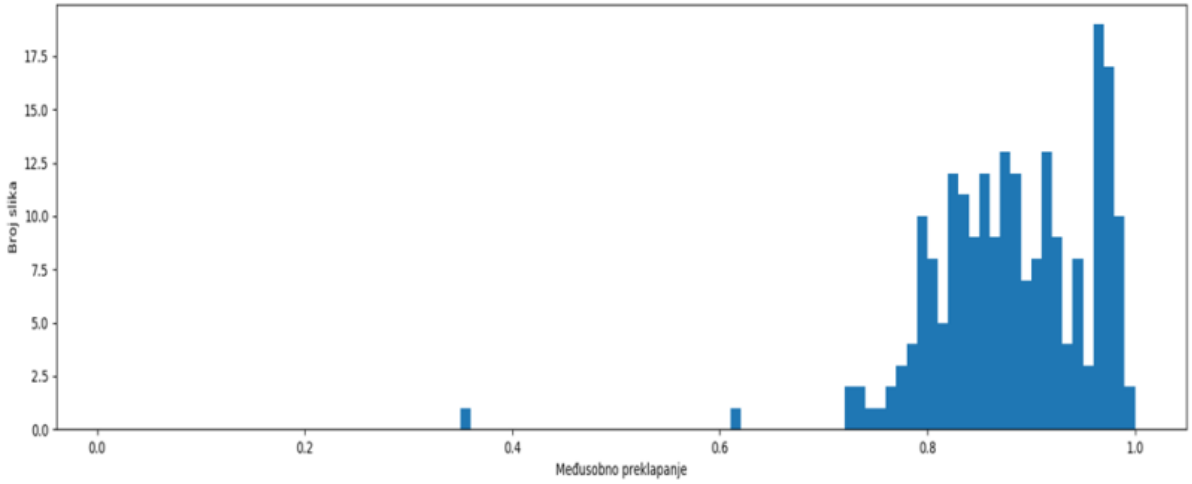
Slika 4.38. Odstupanje širine stvarnog i detektiranog graničnog okvira

Testiranjem video sekvence „*Zadnja strana magla_5*“ dobiven je srednji IoU od 83% te je srednje odstupanje širine stvarnog i dobivenog graničnog okvira 4%. Odstupanje iznosi 6% te je vrlo dobra s obzirom da je video sekvenca snimljena u lošim uvjetima odnosno u magli. Iz histograma koji prikazuje IoU stvarnog i dobivenog graničnog okvira može se vidjeti kako je većina IoU u intervalu od 80-100% no postoje i detekcije koje su unutar intervala 60-80% što unosi pogrešku. Odstupanje širine prevagnulo je na pozitivnu stranu što znači da je detektirani automobil širi od stvarnog automobila, te je i udaljenost detektiranog manja od stvarne udaljenosti automobila. Svega par video okvira odstupa širinu stvarnog i graničnog okvira preko 20% dok je ostatak unutar 20% što je dobro.

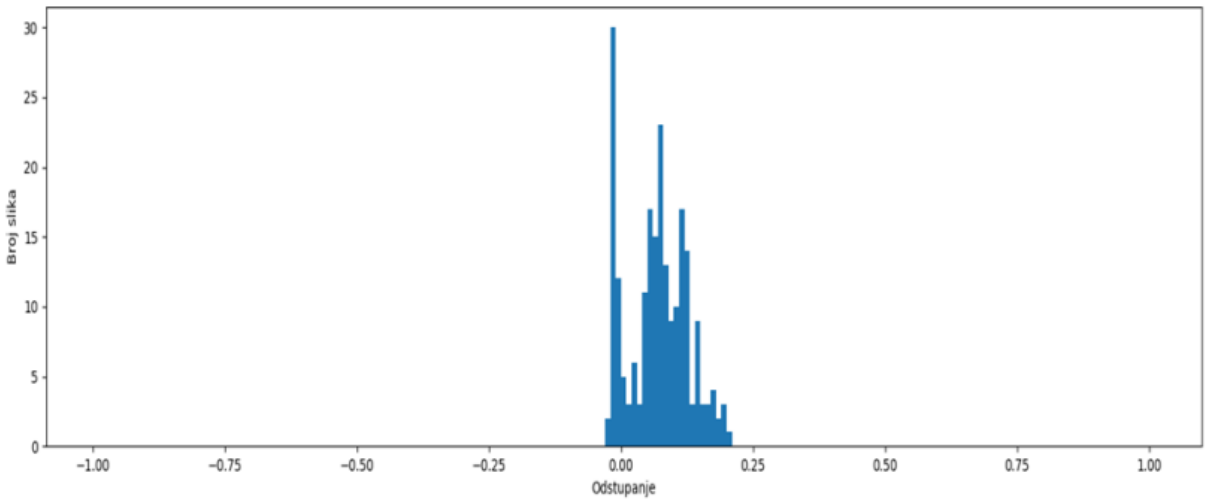
Zadnja strana magla_6		
False positive	0	0%
False negative	4	1.9%
True positive	212	98.1%
True negative	14	73.1%
Slike sadrže auto	216	
Slike ne sadrže auto	19	
Precision	100%	
Recall	98.1%	
Srednji IoU	88%	
Srednje odstupanje	6%	



Tablica 4.21. Statistika za video „*Zadnja strana magla_6*“



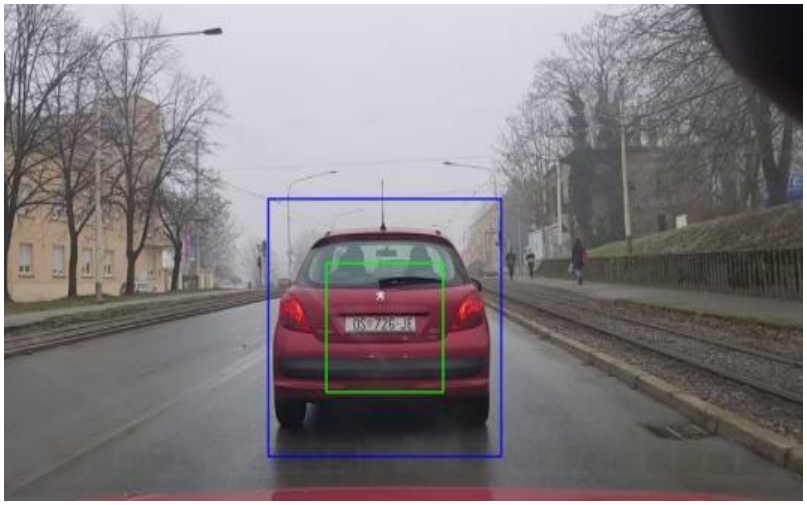
Slika 4.39. Histogram IoU stvarnog i detektiranog graničnog okvira



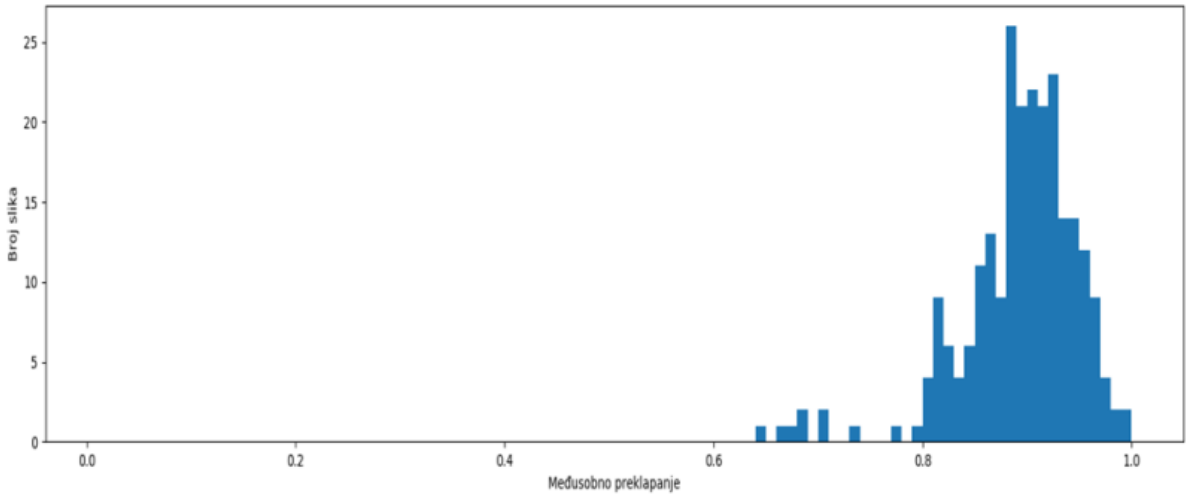
Slika 4.40. Odstupanje širine stvarnog i detektiranog graničnog okvira

Testiranjem video sekvence „*Zadnja strana magla_6*“ dobiven je srednji IoU od 88% te je srednje odstupanje širine stvarnog i dobivenog graničnog okvira 6%. Odstupanje iznosi 6% te je vrlo dobra s obzirom da je video sekvenca snimljena u lošim uvjetima odnosno u magli. Iz histograma koji prikazuje IoU stvarnog i dobivenog graničnog okvira može se vidjeti kako je većina IoU u intervalu od 80-100% što je dobro. Odstupanje širine prevagnulo je na pozitivnu stranu što znači da je detektirani automobil širi od stvarnog automobila, te je i udaljenost detektiranog manja od stvarne udaljenosti automobila. Odstupanje širine stvarnog i graničnog okvira unutar -5% i 20%.

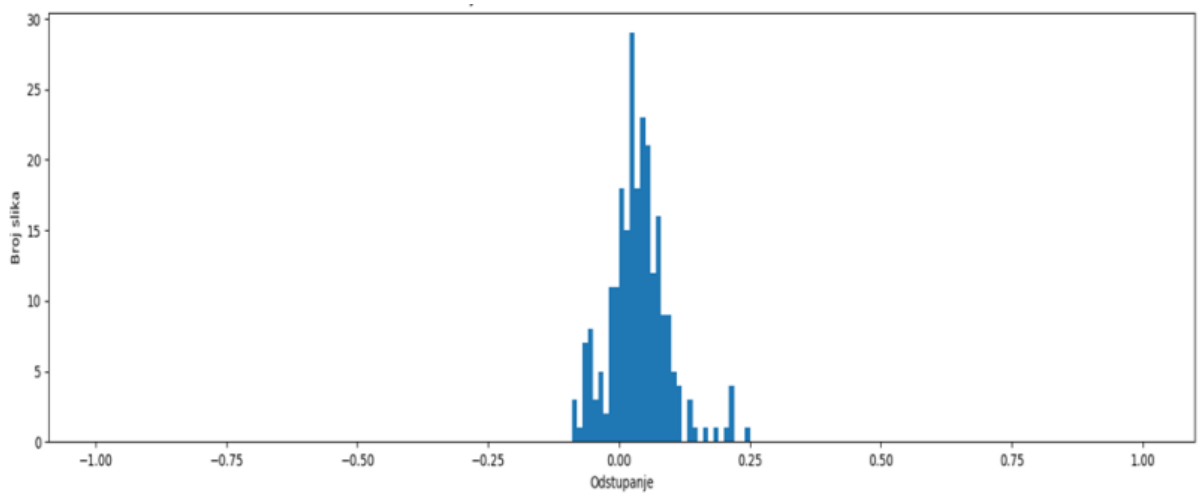
Zadnja strana magla_7		
False positive	0	0%
False negative	2	0.8%
True positive	243	99.2%
True negative	0	0%
Slike sadrže auto	245	
Slike ne sadrže auto	0	
Precision	100%	
Recall	99.2%	
Srednji IoU	89%	
Srednje odstupanje	3%	



Tablica 4.22. Statistika za video „*Zadnja strana magla_7*“



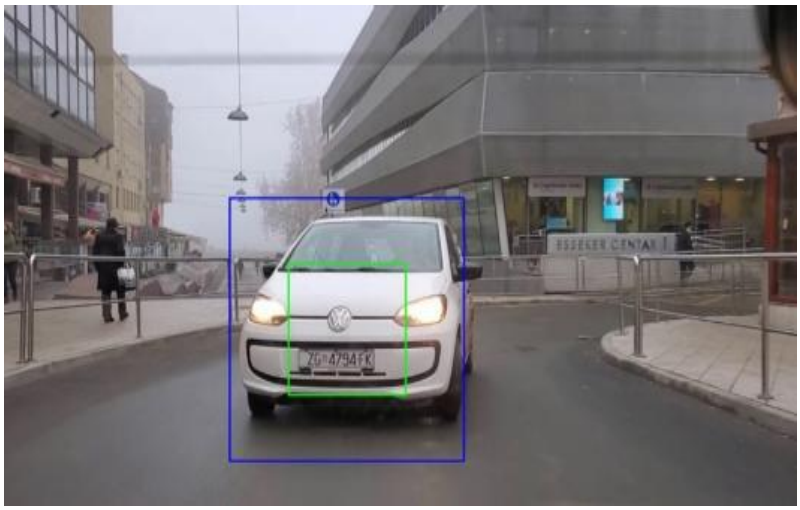
Slika 4.49. Histogram IoU stvarnog i detektiranog graničnog okvira



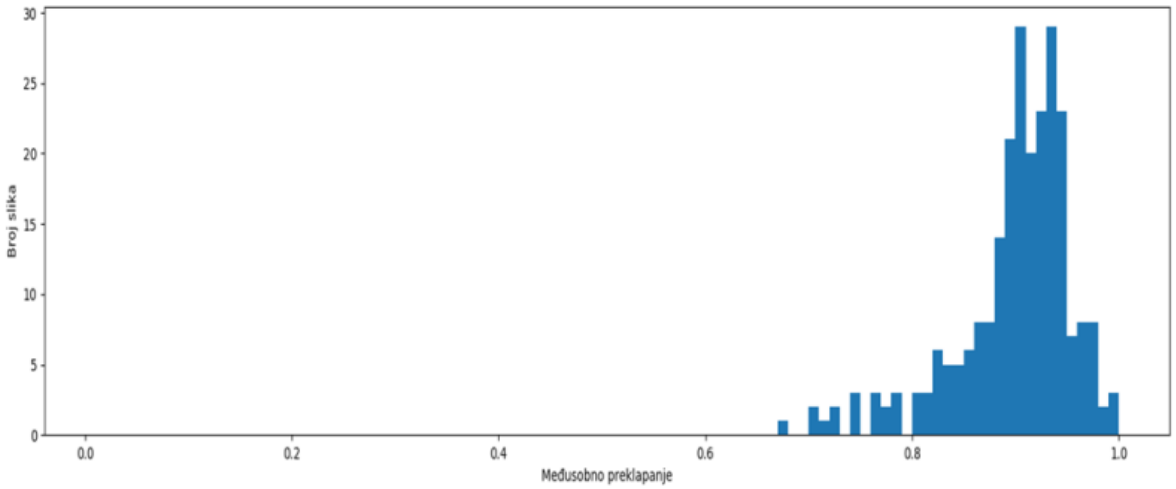
Slika 4.410. Odstupanje širine stvarnog i detektiranog graničnog okvira

Testiranjem video sekvence „*Zadnja strana magla_7*“ dobiven je srednji IoU od 89% te je srednje odstupanje širine stvarnog i dobivenog graničnog okvira 3%. Odstupanje iznosi 3% te je vrlo dobra s obzirom da je video sekvenca snimljena u lošim uvjetima odnosno u magli. Iz histograma koji prikazuje IoU stvarnog i dobivenog graničnog okvira može se vidjeti kako je većina IoU u intervalu od 80-100% te je svega par video okvira unutar intervala 60-80%. Odstupanje širine prevagnulo je na pozitivnu stranu što znači da je detektirani automobil širi od stvarnog automobila, te je i udaljenost detektiranog manja od stvarne udaljenosti automobila. Odstupanje širine stvarnog i graničnog okvira unutar -10% i 25%.

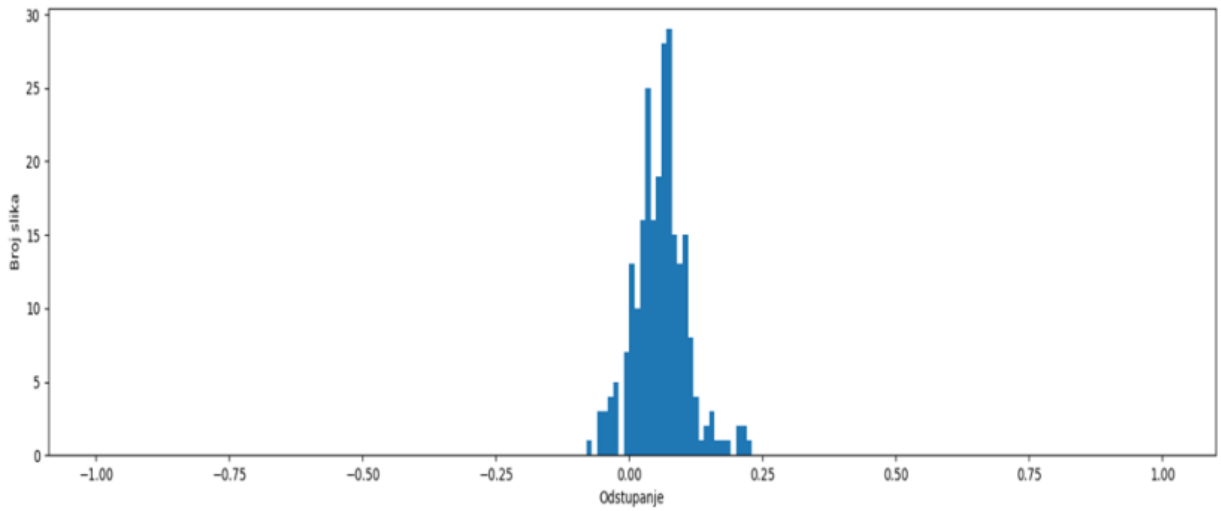
Prednja strana magla_11		
False positive	23	40.3%
False negative	65	20.7%
True positive	249	79.3%
True negative	47	82.5%
Slike sadrže auto	314	
Slike ne sadrže auto	57	
Precision	91.5%	
Recall	79.3%	
Srednji IoU	89%	
Srednje odstupanje	5%	



Tablica 4.23. Statistika za video „*Prednja strana magla_11*“



Slika 4.43. Histogram IoU stvarnog i detektiranog graničnog okvira

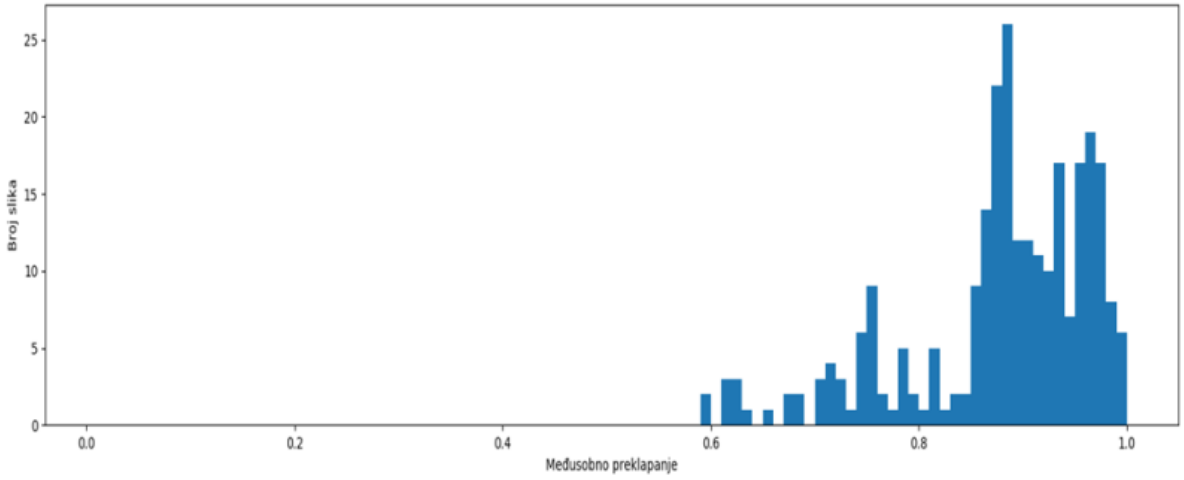


Slika 4.44. Odstupanje širine stvarnog i detektiranog graničnog okvira

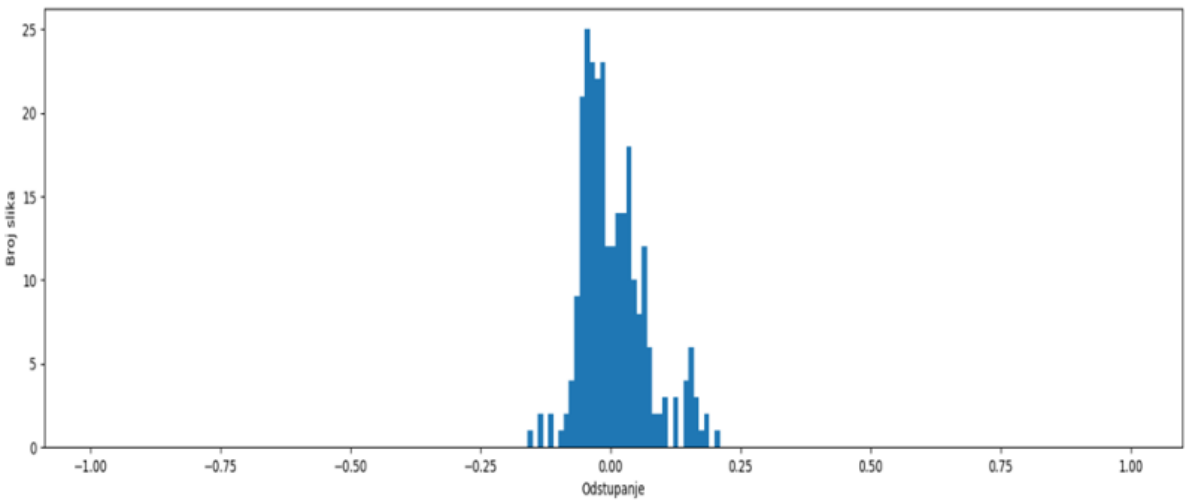
Testiranjem video sekvence „*Prednja strana magla_11*“ dobiven je srednji IoU od 89% te je srednje odstupanje širine stvarnog i dobivenog graničnog okvira 5%. Odstupanje iznosi 5% te je vrlo dobra s obzirom da je video sekvenca snimljena u lošim uvjetima odnosno u magli. Najveća pogreška u ovoj video sekvenci unesena je iz tog razloga što vozilo krivuda po zavoju te se IoU i odstupanje dosta razlikuje (auto se iskosi a granični okvir i dalje označava konstantnu širinu automobila). Iz histograma koji prikazuje IoU stvarnog i dobivenog graničnog okvira može se vidjeti kako je većina IoU u intervalu od 80-100% što je dobro. Odstupanje širine prevagnulo je na pozitivnu stranu što znači da je detektirani automobil širi od stvarnog automobila, te je i udaljenost detektiranog manja od stvarne udaljenosti automobila.

Prednja strana magla_22		
False positive	0	0%
False negative	14	5.2%
True positive	257	94.8%
True negative	0	0%
Slike sadrže auto	271	
Slike ne sadrže auto	0	
Precision	100%	
Recall	94.8%	
Srednji IoU	87%	
Srednje odstupanje	1%	

Tablica 4.24. Statistika za video „*Prednja strana magla_22*“



Slika 4.45. Histogram IoU stvarnog i detektiranog graničnog okvira

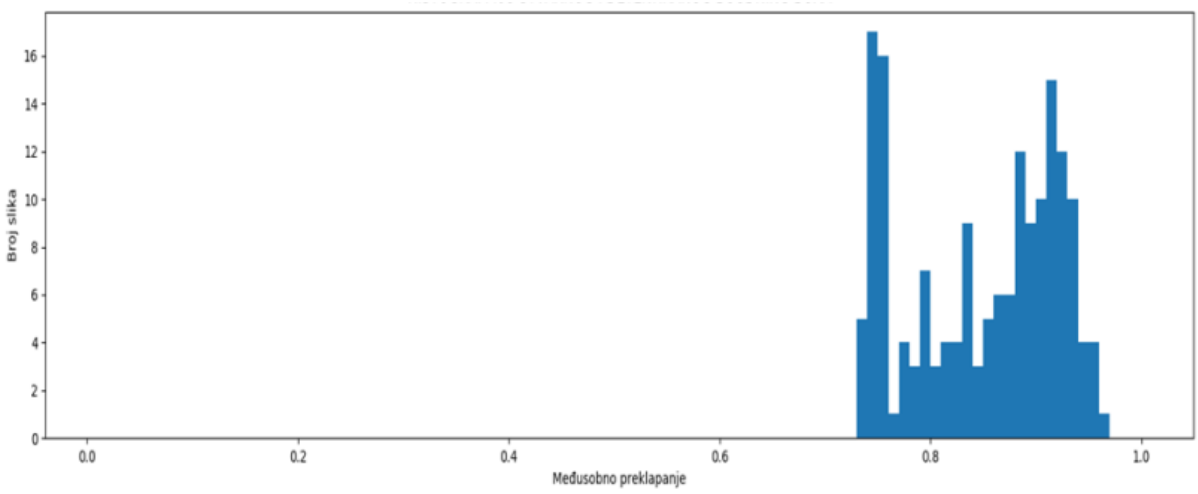


Slika 4.46. Odstupanje širine stvarnog i detektiranog graničnog okvira

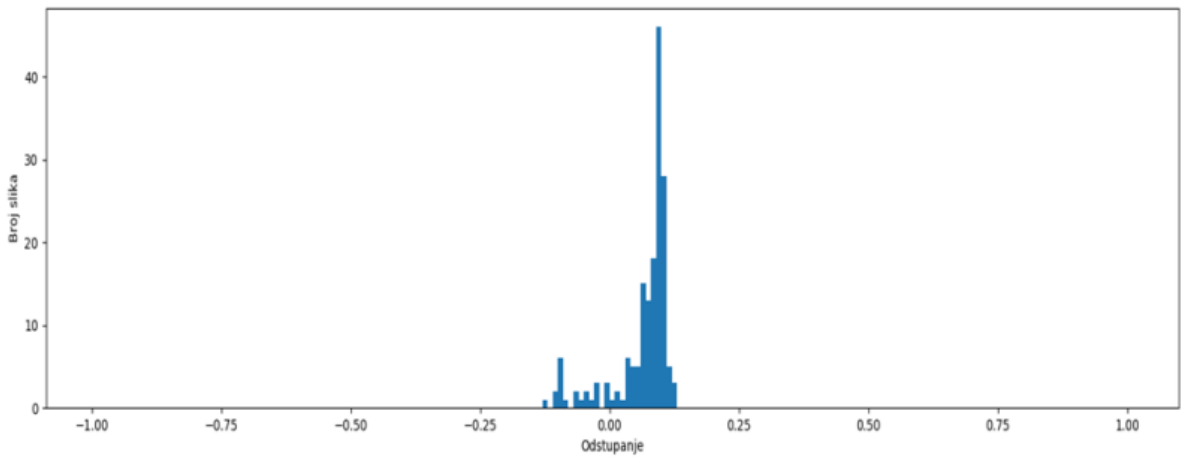
Testiranjem video sekvence „*Prednja strana magla_22*“ dobiven je srednji IoU od 87% te je srednje odstupanje širine stvarnog i dobivenog graničnog okvira 1%. Odstupanje iznosi 1% te je vrlo dobra s obzirom da je video sekvenca snimljena u lošim uvjetima odnosno u magli. Iz histograma koji prikazuje IoU stvarnog i dobivenog graničnog okvira može se vidjeti kako je većina IoU u intervalu od 80-100% što je dobro. Odstupanje širine prevagnulo je na pozitivnu stranu (za 1%) što znači da je detektirani automobil širi od stvarnog automobila, te je i udaljenost detektiranog manja od stvarne udaljenosti automobila. Odstupanje širine stvarnog i graničnog podjednako odstupa u pozitivnu i negativnu stranu zbog čega srednja vrijednost odstupanja i iznosi 1%.

Prednja strana magla_33		
False positive	0	0%
False negative	2	0.2%
True positive	171	98.8%
True negative	0	0%
Slike sadrže auto	173	
Slike ne sadrže auto	0	
Precision	100%	
Recall	98.8%	
Srednji IoU	84%	
Srednje odstupanje	6%	

Tablica 4.25. Statistika za video „*Prednja strana magla_33*“

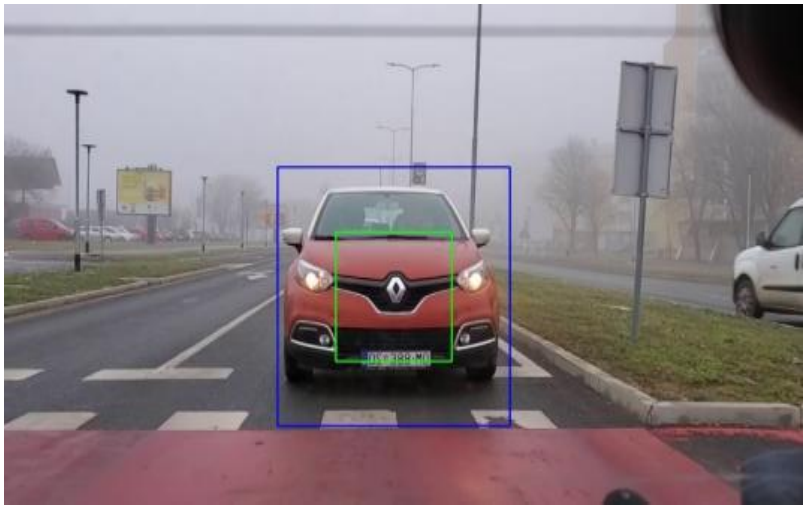


Slika 4.47. Histogram IoU stvarnog i detektiranog graničnog okvira

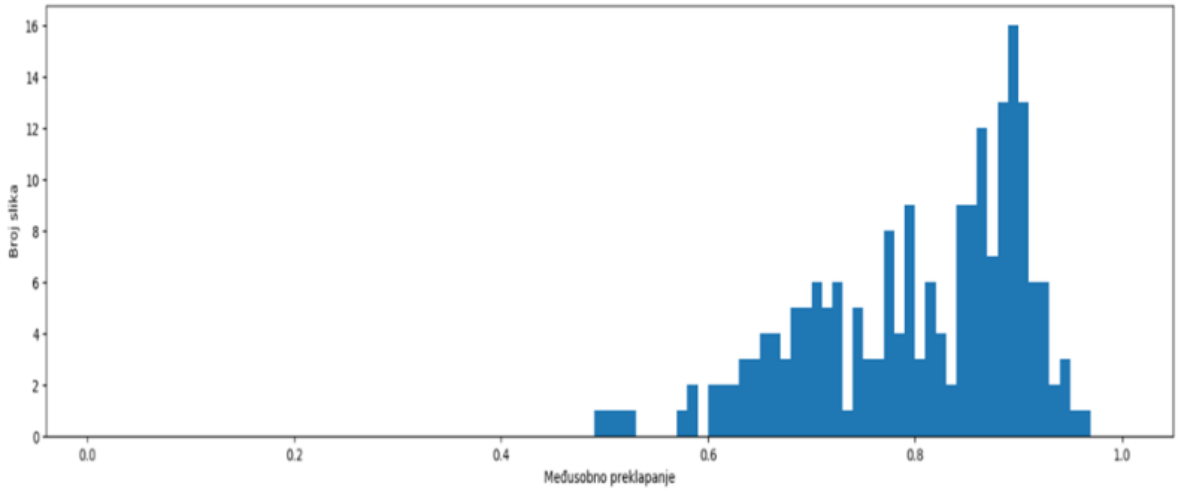


Slika 4.48. Odstupanje širine stvarnog i detektiranog graničnog okvira

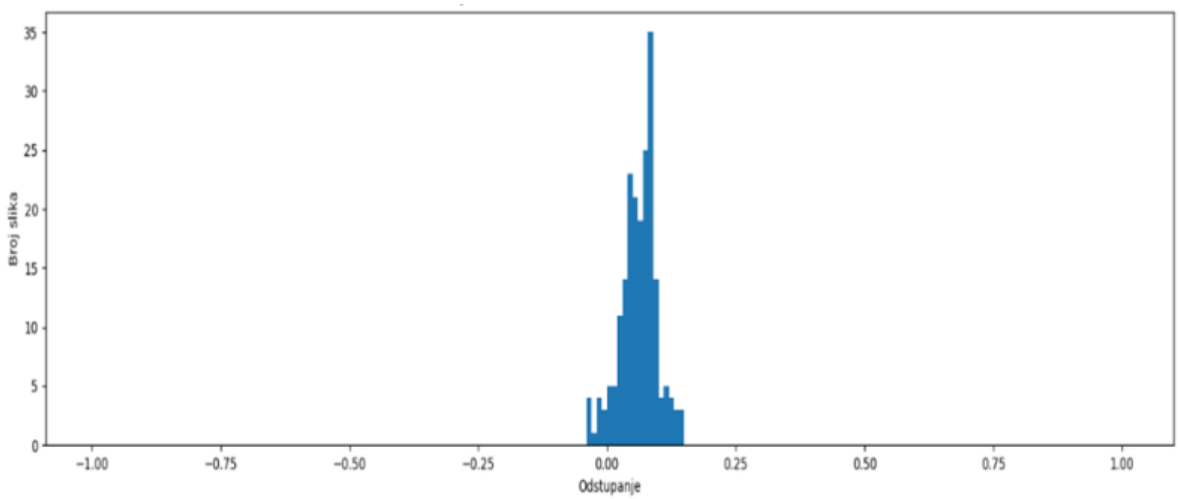
Testiranjem video sekvence „*Prednja strana magla_33*“ dobiven je srednji IoU od 84% te je srednje odstupanje širine stvarnog i dobivenog graničnog okvira 6%. Odstupanje iznosi 6% te je vrlo dobra s obzirom da je video sekvenca snimljena u lošim uvjetima odnosno u magli. Usporedi li se opet ovo testiranje sa testiranjima po suncu, da se zaključiti kako se zbog uvjeta snimanja unosi pogreška – što su uvjeti snimanja lošiji, pogreška je veća. Iz histograma koji prikazuje IoU stvarnog i dobivenog graničnog okvira može se vidjeti kako je većina IoU u intervalu od 70-95% što je dobro. Odstupanje širine prevagnulo je na pozitivnu stranu što znači da je detektirani automobil širi od stvarnog automobila, te je i udaljenost detektiranog manja od stvarne udaljenosti automobila. Pretpostavka je da se odstupanje propagira većinom oko 10% zbog automobila koji je na video sekvenci.

Prednja strana magla_44			
False positive	0	0%	
False negative	15	5.6%	
True positive	254	94.4%	
True negative	0	0%	
Slike sadrže auto	221		
Slike ne sadrže auto	48		
Precision	100%		
Recall	94.4%		
Srednji IoU	80%		
Srednje odstupanje	6%		

Tablica 4.26. Statistika za video „*Prednja strana magla_44*“



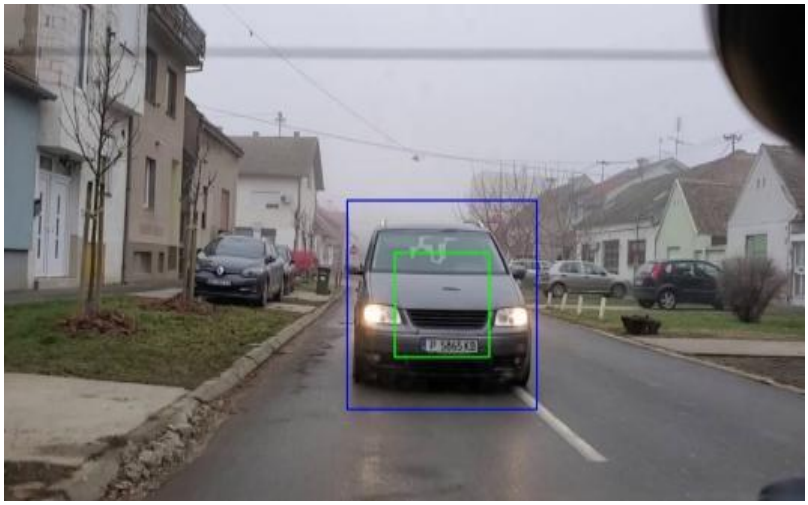
Slika 4.49. Histogram IoU stvarnog i detektiranog graničnog okvira



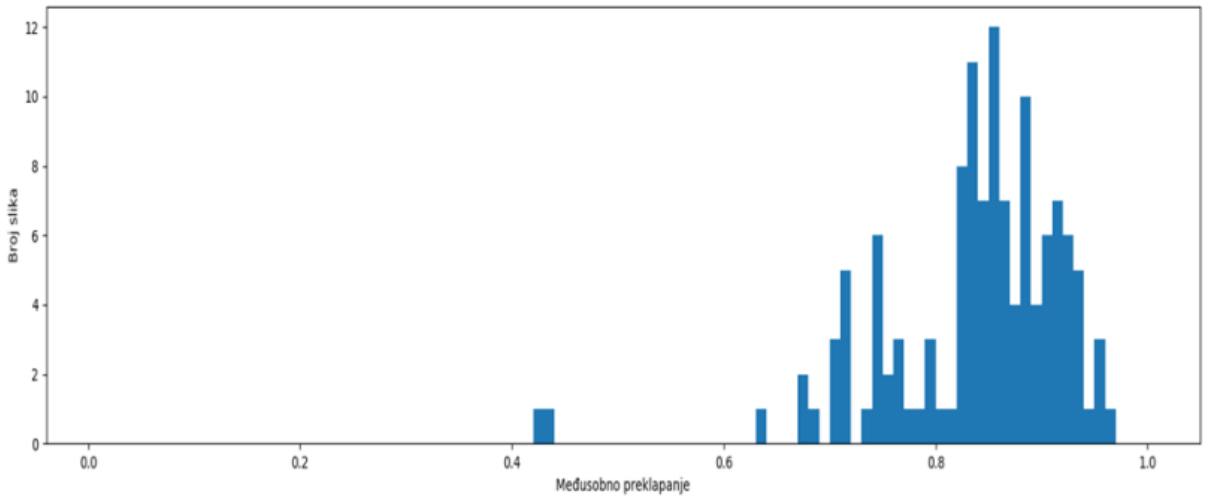
Slika 4.50. Odstupanje širine stvarnog i detektiranog graničnog okvira

Testiranjem video sekvence „*Prednja strana magla_44*“ dobiven je srednji IoU od 80% što je najlošiji rezultat u cijelom testiranju, te je srednje odstupanje širine stvarnog i dobivenog graničnog okvira 6%. Odstupanje iznosi 6% te je vrlo dobra s obzirom da je video sekvenca snimljena u lošim uvjetima odnosno u magli. Iz histograma koji prikazuje IoU stvarnog i dobivenog graničnog okvira može se vidjeti kako vrijednost IoU poprima vrijednosti u intervalu od 60-95% što je poprilično širok raspon. Odstupanje širine prevagnulo je na pozitivnu stranu što znači da je detektirani automobil širi od stvarnog automobila, te je i udaljenost detektiranog manja od stvarne udaljenosti automobila. Odstupanje širine stvarnog i graničnog okvira unutar -5% i 20%.

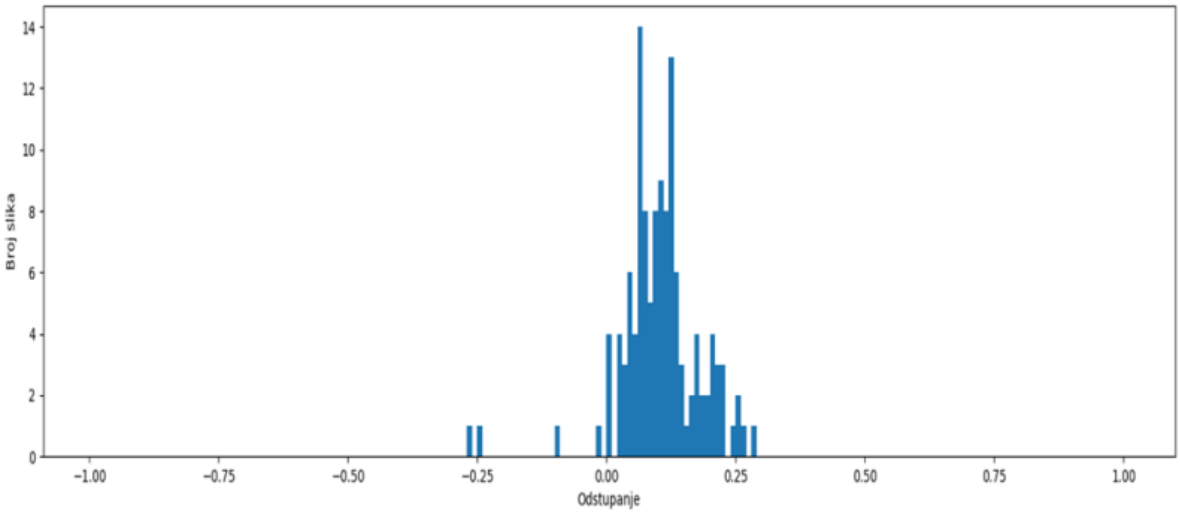
Prednja strana magla_55		
False positive	0	0%
False negative	15	5.6%
True positive	254	94.4%
True negative	0	0%
Slike sadrže auto	221	
Slike ne sadrže auto	48	
Precision	100%	
Recall	94.4%	
Srednji IoU	80%	
Srednje odstupanje	6%	



Tablica 4.27. Statistika za video *„Prednja strana magla_55*“



Slika 4.51. Histogram IoU stvarnog i detektiranog graničnog okvira



Slika 4.52. Odstupanje širine stvarnog i detektiranog graničnog okvira

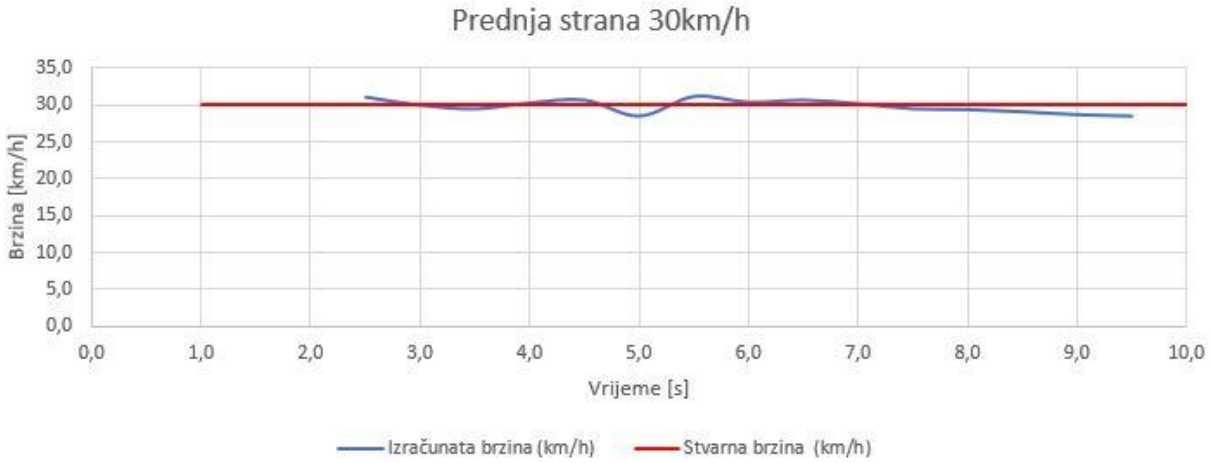
Testiranjem video sekvence „*Prednja strana magla_55*“ dobiven je srednji IoU od 80% kao i u prošlom slučaju. te je srednje odstupanje širine stvarnog i dobivenog graničnog okvira 6%. Odstupanje iznosi 6% te je vrlo dobra s obzirom da je video sekvenca snimljena u lošim uvjetima odnosno u magli. Histogram je nešto bolji od prošlog testiranja no i dalje je među lošijima od ostatka testiranja i većina vrijednosti se kreće u intervalu od 80-100% no postoje i par video okvira gdje je IoU u intervalu od 30-80%. Odstupanje širine znatno je prevagnulo na pozitivnu stranu što znači da je detektirani automobil širi od stvarnog automobila, te je i udaljenost detektiranog manja od stvarne udaljenosti automobila. Odstupanje širine stvarnog i graničnog okvira unutar -27% i 30%.

Iz danih rezultata vidljivo je kako algoritam daje najbolje rezultate prilikom dobrog osvjetljenja odnosno u dobrim vremenskim uvjetima (što su uvjeti lošiji to i algoritam lošije radi). Što se tiče razlike između detekcije prednje i stražnje strane automobila, ne može se sa sigurnošću reći jer niti jedan video ne prikazuje detektirani automobil i sa prednjom i stražnjom kamerom u istim uvjetima no razlika u radu algoritma jer minimalna (u nekim slučajevima manji IoU ili veće odstupanje širine detektiranog i stvarnog graničnog okvira)

4.2. Testiranje točnosti izračunate brzine

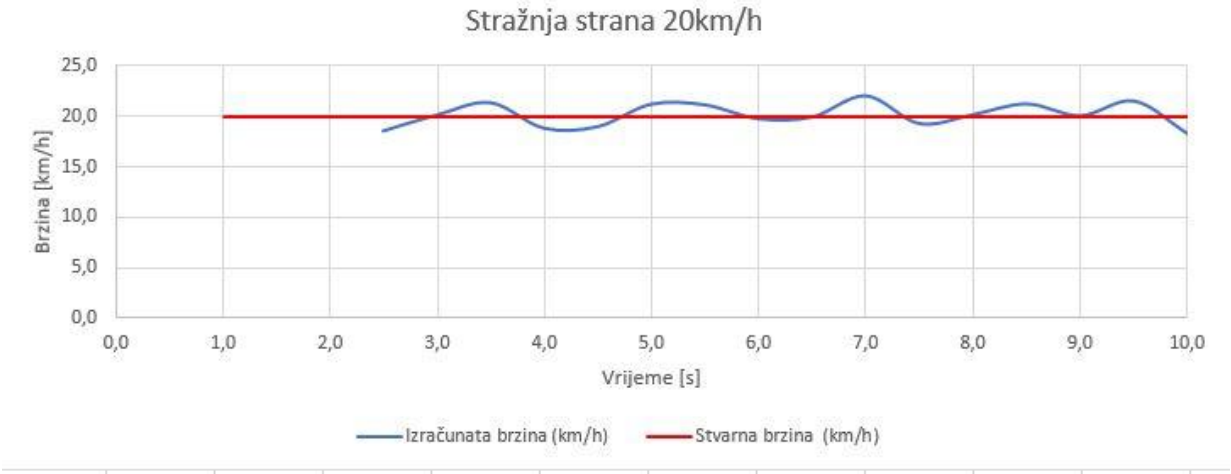
U ovom potpoglavlju napravljena je analiza izračunate brzine. Testiranje je provedeno na 6 video sekvenci (3 prednja strana automobila, 3 stražnja strana automobila) za 3 različite brzine: 20km/h, 30km/h i 50km/h. Video sekvence se nalaze priložene uz DVD i nalaze se u mapi Prilog 4.3. Na sljedećim slikama možemo vidjeti ovisnost stvarne brzine i brzine koju izračunava algoritam u ovisnosti o vremenu.

Slika 4.53. Prikaz brzine za prednju stranu automobila pri brzini 20km/h

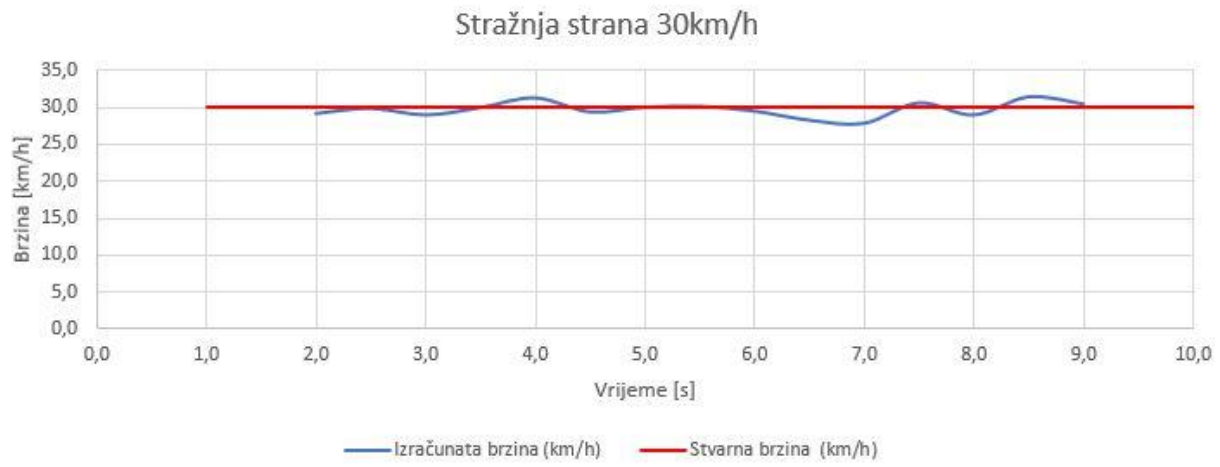


Slika 4.54. Prikaz brzine za prednju stranu automobila pri brzini 30km/h

Slika 4.55. Prikaz brzine za prednju stranu automobila pri brzini 50km/h



Slika 4.56. Prikaz brzine za stražnju stranu automobila pri brzini 20km/h



Slika 4.57. Prikaz brzine za stražnju stranu automobila pri brzini 30km/h

Slika 4.58. Prikaz brzine za stražnju stranu automobila pri brzini 50km/h

Iz prethodnih slika može se vidjeti kako je točnost brzine manja što je udaljenost automobila veća, najveća točnost algoritma postiže se kada je on jako blizu (20ak metara). Oscilacija brzine se dešava zbog pogreške u proračunu udaljenosti odnosno zbog pogreške koju unosi detektor *Viola-Jones*.

U sljedećoj tablici se nalazi srednja vrijednost brzine i standardna devijacija odstupanja za svaku video sekvencu.

Tablica 4.28. Prikaz srednje vrijednosti i standardne devijacije brzine za svaki od videa

Naziv videa	Srednja vrijednost brzine [km/h]	Standardna devijacija brzine [km/h]
Prednja strana 20km/h	19,6	1,11
Prednja strana 30km/h	29,8	0,90
Prednja strana 50km/h	49,8	1,60
Stražnja strana 20km/h	20,2	1,08
Stražnja strana 30km/h	29,8	1,00
Stražnja strana 50km/h	49,9	1,30

5. ZAKLJUČAK

U ovom radu predložen je algoritam koji se sastoji u tri glavna dijela: detekcija vozila koje se nalazi ispred/iza vozila na kojemu je implementiran algoritam, proračun udaljenosti vozila ispred/iza te proračun brzine vozila ispred/iza. Algoritam uvelike pridonosi autonomnosti vozila jer vozačevu subjektivnu procjenu zamjenjuje algoritam kojemu je odziv znatno brži. Algoritam radi u stvarnom vremenu što bi značilo da ga se može implementirati u stvarno vozilo kako bi se smanjio rizik od sudara. Greška koja nastaje prilikom procjene udaljenosti vozila jest unesena namjerno jer je uzeta prosječna širina vozila, te je prisutna i ne namjerna greška koju unosi detektor koji ne označava stvarnu širinu vozila na pojedinom video okviru. Također, u svakom trenutku algoritam mora imati informaciju o brzini vozila na kojemu je implementiran kako bi se izvršio točan proračun brzine vozila koje se nalazi ispred/iza.

Testiranjem algoritma pokazano je da algoritam radi s određenom pogreškom, postoje pogreške prilikom klasifikacije (*false pos/neg, true pos/neg*) te postoje pogreške prilikom određivanja udaljenosti i brzine. Testiranje je provedeno na oko 7000 video okvira koji su snimljeni u različitim vremenskim uvjetima (kiša, sunce, magla, lošije osvjetljenje, bolje osvjetljenje) te na različitim automobilskim markama. Prilikom testiranja, korištena su oba slučaja postavljanja kamere (kamera postavljena na stražnjoj strani vozila, kamera postavljena na prednjoj strani vozila) te je omjer videa prednje i stražnje strane otprilike podjednak. Algoritam nije bilo moguće implementirati u stvarno vozilo te ga isprobati, stoga je testiranje napravljeno isključivo na video okvirima.

LITERATURA

[1] Global Health Observatory (GHO), data, https://www.who.int/gho/road_safety/mortality/en/ , pristup ostvaren [15.02.2020.]

[2] Racing Elite Home, https://racing-elite.com/2019/08/14/f1-test-test-your-reaction-time-at-the-start-of-the-f1-race/ ,pristup ostvaren dana [15.02.2020.]

[3] Z. Czapla, Vehicle Speed Estimation with the Use of Gradient-Based Image Conversion into Binary Form, Signal Processing Algorithms, Architectures, Arrangements, and Aplications, 2017.

[4] OpenCV-Python tutorials, https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_canny/py_canny.html, pristup ostvaren dana [15.02.2020.]

[5] OpenCV-Python tutorials,https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_houghlines/py_houghlines.html, pristup ostvaren dana [15.02.2020.]

[6] OpenCV Object Tracking, https://www.pyimagesearch.com/2018/07/30/opencv-object-tracking/, pristup ostvaren dana [18.02.2020.]

[7] K. Giseok, C. Jae-Soo, Vision-based vehicle detection and inter-vehicle distance estimation, Department of Computer Science and Engineering, Korea University of Technology and Education, Cheonan, Korea, 2012

[8] Face Detection using Haar Cascades, https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_objdetect/py_face_detection/py_face_detection.html, pristup ostvaren dana[18.02.2020.]

[9] OpenCV-Cascade Classifier, https://docs.opencv.org/3.4/db/d28/tutorial_cascade_classifier.html, pristup ostvaren dana [18.02.2020.]

[10] T. Babb, How a Kalman filter works in pictures, https: www.bzarg.com/p/how-a-kalmanfilter-works-in-pictures/, pristup ostvaren dana [18.02.2020.]

[11] Car speed detector, https://gregtinkers.wordpress.com/2016/03/25/car-speed-detector/, pristup ostvaren dana [01.03.2020.]

[12] Flickr photos, https://www.flickr.com/photos/150908913@N06/35458941320/, pristup ostvaren dana [01.03.2020.]

[13] M. Vranješ, Predavanje s kolegija „*Digitalna obrada slike i videa – Detekcija lica, pješaka i automobila“*, 2018.

[14] Computer Science Source, https://computersciencesource.wordpress.com/2010/09/03/computer-vision-the-integral-image/, pristup ostvaren dana [01.03.2020.]

[15] Cascade Classifier Training, https://docs.opencv.org/master/dc/d88/tutorial_traincascade.html , pristup ostvaren dana [01.03.2020.]

[16] Intersection over Union (IoU) for object detection, https://www.pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/ , pristup ostvaren dana [01.03.2020.]

[17] Device specification, https://www.devicespecifications.com/en/model/82354f1f, pristup ostvaren dana [02.03.2020.]

[18] Samsung CMOS, https://en.wikipedia.org/wiki/Samsung_CMOS , pristup ostvaren dana [02.03.2020.]

[19] Car size comparison with dimensions of length, width and height, https://www.automobiledimension.com/car-comparison.php, pristup ostvaren dana [02.03.2020.]

SAŽETAK

U ovom diplomskim radom razvijen je algoritam koji na temelju udaljenosti vozila koje se nalazi ispred ili iza vozila na kojemu je implementiran algoritam, računa brzinu te poduzima potrebnu radnju. U predloženom algoritmu korišten je *Viola-Jones* detektor koji služi za detekciju vozila nakon čega se proračunava udaljenost vozila. *Viola-Jones* detektor je istreniran na ranije prikupljenim slikama nakon čega je testiran na novom skupu slika. Iz rezultata testiranja može se vidjeti kako algoritam radi prilično dobro (vrlo rijetko se dešavaju netočne detekcije i detekcije kojima je IoU manji od 70%).

Ključne riječi: Viola-Jones, detekcija rizika od sudara i pametna kontrola brzine, izračun udaljenosti, izračun brzine, detekcija vozila, autonomija vožnje

ABSTRACT

Crash risk detection and smart speed control

In this graduate thesis, an algorithm was developed which, based on the distance of the vehicle which stands in front of or behind the vehicles on which the algorithm was implemented, calculate speed and takes some action. In the proposed algorithm, the Viola-Jones detector for vehicle detection is used. The Viola-Jones detector was trained on previously acquired images, after which is tested on a new set of images. The test results show that the algorithm works quite well (very rarely there are inaccurate detections and detections with IoU less than 70%).

Keywords: Viola-Jones, crash risk detection and smart speed control, distance calculation, speed calculation, vehicle detection, driving autonomy

ŽIVOTOPIS

Eugen Šimara rođen je 20.02.1996. godine u Osijeku. U Viljevu završava "Osnovna škola Ante Starčevića – Viljevo", nakon koje upisuje srednju školu zanimanja elektrotehničara u Slatini "Srednja škola Marka Marulića Slatina". Tokom srednjoškolskog obrazovanja sudjeluje u županijskom natjecanju iz fizike. Postiže 1. mjesto na županijskom natjecanju u Virovitici te 1. mjesto na regionalnom natjecanju u Varaždinu u atletici, disciplina – bacanje kugle. Po završetku srednjoškolskog obrazovanja, upisuje preddiplomski studij elektrotehnike na Fakultetu elektrotehnike, računarstva i informacijskih tehnologija u Osijeku te se na diplomskom studiju opredjeljuje za smjer Automobilsko računarstvo i komunikacije.

U Osijeku, ožujak 2020. Eugen Šimara

_____

PRILOZI

- **P.3.1.** Skripta u kojoj je pohranjen kod algoritma koji je izrađen (priloženo na DVD-u uz rad)
- **P.3.2.** Slike korištenje u procesu treniranja *Viola-Jones* algoritma koji služi za detekciju objekata (priloženo na DVD-u uz rad)
- **P.3.3.** Datoteka u kojoj se nalaze *info.lst, bg.txt* i *positives.vec* potrebne za proces treniranja (priloženo na DVD-u uz rad)
- **P.3.4.** Skripta pomoću koje su sortirane koordinate točaka (priloženo na DVD-u uz rad)
- **P.3.5.** Skripta pomoću koje je vršeno označavanje stvarnih graničnih okvira (priloženo na DVD-u uz rad)
- **P.3.6.** Skripta pomoću koje je parsirana *.xml* datoteka kako bi se dobile koordinate točaka (priloženo na DVD-u uz rad)
- **P.4.1.** Video sekvence korištene za testiranje izrađenog algoritma snimljene u gradu Osijeku (priloženo na DVD-u uz rad)
- **P.4.2.** Skripta pomoću koje je računato prosječno odstupanje širine i IoU (priloženo na DVD-u uz rad)
- **P.4.3.** Video sekvence korištene za testiranje točnosti brzine (priloženo na DVD-u uz rad)