

# Semantička segmentacija slika prometnica generiranih CARLA simulatorom

---

**Aničić, Simon**

**Master's thesis / Diplomski rad**

**2023**

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **University of Pula / Sveučilište Jurja Dobrile u Puli**

*Permanent link / Trajna poveznica:* <https://um.nsk.hr/um:nbn:hr:137:228791>

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

*Download date / Datum preuzimanja:* **2024-11-23**



*Repository / Repozitorij:*

[Digital Repository Juraj Dobrila University of Pula](#)



Sveučilište Jurja Dobrile u Puli  
Fakultet informatike u Puli

**SIMON ANIČIĆ**

**SEMANTIČKA SEGMENTACIJA SLIKA PROMETNICA GENERIRANIH CARLA  
SIMULATOROM**

Diplomski rad

Pula, rujan, 2023. godine.

Sveučilište Jurja Dobrile u Puli  
Fakultet informatike u Puli

**SIMON ANIČIĆ**

**SEMANTIČKA SEGMENTACIJA SLIKA PROMETNICA GENERIRANIH CARLA  
SIMULATOROM**

Diplomski rad

**JMBAG:** 0242046952, redoviti student

**Studijski smjer:** Sveučilišni diplomski studij Informatika

**Predmet:** Neuronske mreže i duboko učenje

**Znanstveno područje:** Društvene znanosti

**Znanstveno polje:** Informatičke i komunikacijske znanosti

**Znanstvena grana:** Informatički sustavi i informatologija

**Mentor:** doc. dr. sc. Goran Oreški

Pula, rujan, 2023. godine



## IZJAVA O AKADEMSKOJ ČESTITOSTI

Ja, dolje potpisani **Simon Aničić**, kandidat za magistra **informatike, mag. inf.** ovime izjavljujem da je ovaj Diplomski rad rezultat isključivo mogega vlastitog rada, da se temelji na mojim istraživanjima te da se oslanja na objavljenu literaturu kao što to pokazuju korištene bilješke i bibliografija. Izjavljujem da niti jedan dio Diplomskog rada nije napisan na nedozvoljen način, odnosno da je prepisan iz kojega necitiranog rada, te da ikoji dio rada krši bilo čija autorska prava. Izjavljujem, također, da nijedan dio rada nije iskorišten za koji drugi rad pri bilo kojoj drugoj visokoškolskoj, znanstvenoj ili radnoj ustanovi.

Student

---

U Puli, rujan, 2023. godine



IZJAVA  
o korištenju autorskog djela

Ja, Simon Aničić dajem odobrenje Sveučilištu Jurja Dobrile u Puli, kao nositelju prava iskorištavanja, da moj diplomski rad pod nazivom Semantička segmentacija slika prometnica generiranih CARLA simulatorom koristi na način da gore navedeno autorsko djelo, kao cjeloviti tekst trajno objavi u javnoj internetskoj bazi Sveučilišne knjižnice Sveučilišta Jurja Dobrile u Puli te kopira u javnu internetsku bazu završnih radova Nacionalne i sveučilišne knjižnice (stavljanje na raspolaganje javnosti), sve u skladu s Zakonom o autorskom pravu i drugim srodnim pravima i dobrom akademskom praksom, a radi promicanja otvorenoga, slobodnoga pristupa znanstvenim informacijama.

Za korištenje autorskog djela na gore navedeni način ne potražujem naknadu.

U Puli, 01.09.2023. godine

Potpis

---

## Sadržaj

|   |    |
|---|----|
| 1. Uvod .....   | 1  |
| 2. Semantička segmentacija.....   | 3  |
| 2.1. Semantička segmentacija temeljena na regiji.....                             | 4  |
| 2.2. Semantička segmentacija temeljena na potpuno povezanoj mreži .....           | 5  |
| 2.2.1. FPN.....   | 5  |
| 2.2.2. DeepLabV3Plus .....  | 6  |
| 2.2.3. U-net .....  | 6  |
| 2.3. Slabo nadzirana semantička segmentacija .....                                | 7  |
| 3. Funkcije gubitka.....  | 9  |
| 4. Evaluacijske mjere.....  | 11 |
| 4.1. Preciznost.....  | 11 |
| 4.2. Odziv .....  | 12 |
| 4.3. F1 mjera .....   | 12 |
| 4.4. IoU mjera.....   | 13 |
| 5. Tehnike augmentacije podataka.....   | 14 |
| 6. CARLA simulator .....  | 15 |
| 6.1. Prikupljanje slika za skup podataka.....                                     | 16 |
| 7. Usporedba funkcija gubitka za semantičku segmentaciju objekata u prometu ..... | 18 |
| 7.1. Uvod .....   | 18 |
| 7.2. Pregled literature .....   | 19 |
| 7.3. Funkcije gubitka .....   | 20 |
| 7.3.1. <i>Cross entropy</i> .....   | 21 |
| 7.3.2. Dice.....  | 21 |
| 7.3.3. <i>Jaccard/Intersection over union (IoU)</i> .....                         | 22 |
| 7.3.4. <i>Focal</i> .....   | 22 |

|        |  |    |
|--------|--|----|
| 7.3.5. | Tversky .....  | 23 |
| 7.4.   | Eksperiment i rezultati analize .....  | 23 |
| 7.4.1. | Skup podataka iz simulatora CARLA .....  | 24 |
| 7.4.2. | Modeli .....   | 25 |
| 7.4.3. | Evaluacijske mjere .....   | 25 |
| 7.4.4. | Rezultati .....  | 26 |
| 7.5.   | Zaključak .....  | 28 |
| 8.     | Usporedba tehnika za augmentaciju podataka za semantičku segmentaciju objekata u prometu ..... | 29 |
| 8.1.   | Uvod .....   | 29 |
| 8.2.   | Pregled literature .....   | 30 |
| 8.3.   | Augmentacija podataka i tehnike .....  | 32 |
| 8.3.1. | Nasumična rotacija .....   | 32 |
| 8.3.2. | Nasumična inverzija boja .....   | 33 |
| 8.3.3. | Nasumično vertikalno zaokretanje .....   | 33 |
| 8.3.4. | Nasumično horizontalno zaokretanje .....   | 33 |
| 8.3.5. | Nasumično brisanje dijelova slike .....  | 34 |
| 8.3.6. | Nasumično pretvaranje slike u sive nijanse .....   | 34 |
| 8.4.   | Eksperiment i rezultati analize .....  | 34 |
| 8.4.1. | Skup podataka iz simulatora CARLA .....  | 35 |
| 8.4.2. | Modeli .....   | 35 |
| 8.4.3. | Evaluacijske mjere .....   | 36 |
| 8.4.4. | Rezultati .....  | 36 |
| 8.5.   | Zaključak .....  | 48 |
| 9.     | Zaključak diplomskog rada .....  | 50 |
|        | Popis literature .....   | 51 |
|        | Popis slika .....  | 59 |

|                                   |    |
|-----------------------------------|----|
| Popis tablica .....               | 59 |
| Popis jednadžbi .....             | 60 |
| Sažetak na hrvatskom jeziku ..... | 61 |
| Abstract .....                    | 62 |



## 1. Uvod

Danas je autonomna vožnja sve važniji čimbenik u životu pojedinca, a samim time povećava se i važnost razvijanja modela dubokog učenja koji će obavljati takav zadatak. Autonomna vožnja ne uključuje samo kontrolu i upravljanje određenim vozilom već zahtijeva i određenu razinu sigurnosti i sposobnosti sustava da precizno predviđa i očekuje moguće događaje na prometnicama, prati okolinu te da u svakom trenutku donosi brze i ispravne odluke kako bi se osigurala sigurnost svih sudionika u prometu.

Tehnologija koja omogućuje takav način vožnje uvelike bi povećala sigurnost, efikasnost i mobilnost na način da se umjesto oslanjanja na vozača, ljudi oslanjaju na sama vozila da ih vode kroz promet (Beiker, 2012). Sigurnost nije jedini razlog za korištenje sustava autonomne vožnje, iako je po *National Highway Traffic Safety Administration* 94% svih prometnih nesreća zapravo uzrokovano ljudskom pogreškom (Singh, 2015). Neki od razloga su također i smanjenje štetnih emisija, prijevoz osoba s invaliditetom ili nemogućnost vožnje automobila i smanjenje stresa koji je nastao zbog vožnje.

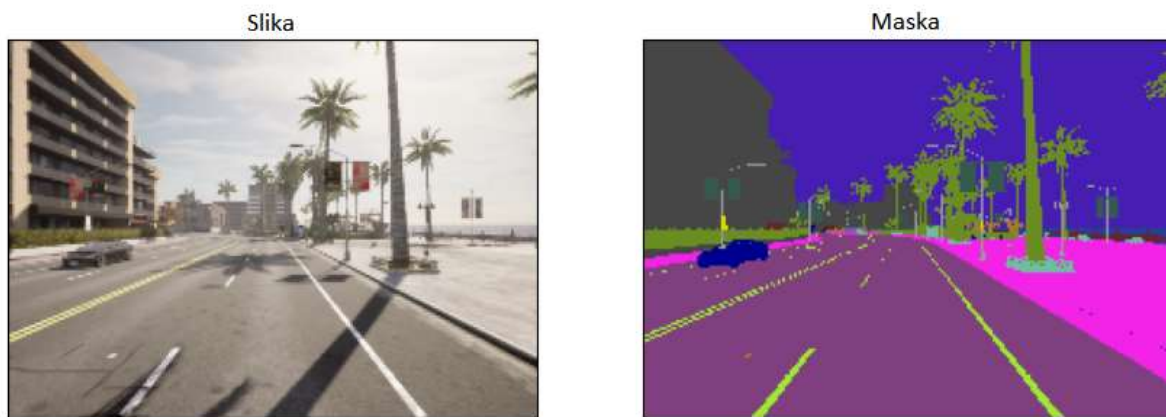
Stvaranje modela dubokog učenja rješava takav zadatak. Adekvatnim treniranjem model može postati dovoljno precizan da ga se koristi u tu svrhu, odnosno da može u vrlo kratkom vremenu donijeti ispravnu odluku u prometu. Kako bi model imao mogućnost predviđanja različitih događaja u prometnicama mora imati skup podataka za trening u kojem se nalaze događaji i scenariji koji bi se mogli dogoditi u svakodnevnom prometu. Kreiranje skupa podataka je dugotrajan i financijski skup proces, a posebice kod označavanja podataka koji se koriste u svrhu treniranja takvog modela na kojem se vrši nadzirano učenje (Wang et al., 2020). Skup podataka je jedan od osnovnih koraka koji treba biti zadovoljen kako bi se izvršilo treniranje jer bez skupa podataka model nema izvor iz kojega može *učiti* o mogućim događajima na prometnicama te objektima koji se mogu pojaviti. Korištenje simulatora gradske vožnje omogućuje stvaranje skupa podataka bez velikih financijskih troškova, a ovisno o simulatoru nudi i široki raspon vozila, pješaka, objekata, prometnih znakova i ostalih elemenata koji se mogu pojaviti u svakodnevnom odvijanju prometa. Također može nuditi i postavljanje drugih elemenata kao naprimjer doba dana ili vremenskih uvjeta (kiša, snijeg, sunce, magla, itd). Model mora imati sposobnost razlikovanja različitih

elemenata, odnosno segmentacije objekata po vrstama (klasama), a takva tehnika naziva se semantička segmentacija.

Ovaj rad sastoji se od teorijskog dijela te dva znanstvena rada, od kojih je jedan usmjeren na usporedbu različitih funkcija gubitka, a drugi na usporedbu različitih tehnika za augmentaciju podataka.

## 2. Semantička segmentacija

Semantička segmentacija dugo je bila jedan od velikih izazova u računalnom vidu (Guo et al., 2018). U suštini, semantička segmentacija nastoji dodijeliti svaki piksel dane slike određenoj klasi, odnosno svaki piksel u semantičkoj segmentaciji pripada nekoj od klasa (Wang et al., 2018). Naprimjer ukoliko model kao ulazni podatak dobije sliku koja na sebi sadržava samo dva automobila i zgradu, svaki piksel koji tvori automobile biti će klasificiran u jednu klasu dok će svaki piksel koji tvori zgradu biti klasificiran u drugu klasu.



*Slika 1. Slika i maska semantičke segmentacije*

Na Slici 1 prikazane su dvije slike, prva slika prikazuje generiranu sliku bez segmentacije, a druga slika je maska. Prva slika je ulazni podatak dan modelu na kojoj se nalaze cesta, vozila, zgrade, prometni znakovi, stabla i razni drugi objekti. Maska je semantički segmentirani prikaz slike na kojoj jedna boja predstavlja jednu klasu, a objekti koji pripadaju istoj klasi obojani su u istu boju.

Klasificiranje svakog piksela obilježje je semantičke segmentacije, no postoje razni drugi zadaci u dubokom učenju kojima je zadatak prepoznavanje objekata (engl. *object recognition*). Prepoznavanje objekata je zadatak koji za dani ulazni podatak traži objekt (ili objekte) za čije je prepoznavanje istreniran te samo njih prepoznaje i označava. Semantička segmentacija nastoji i uspijeva svaki piksel dane slike kategorizirati neovisno o uspješnosti prepoznavanja objekta. Kod semantičke segmentacije nepoznati objekti se također označavaju i kategoriziraju u zasebnu kategoriju. Takav

način percipiranja i odvajanja objekata može se prepoznati i kod ljudi, iako čovjek ne zna sve objekte na slici moći će prepoznati granice tog objekta i svrstati ga u zasebnu kategoriju od onih objekata koje prepoznaje.

Segmentacija se može koristiti u raznim slučajevima, a njenom uporabom se stvaraju bolji i robusniji algoritmi. Uporaba segmentacije mogla bi se primijeniti na različitim problemima npr. kod očuvanja kulturnih dobara pa sve do analize satelitskih slika i interakcije između čovjeka i računala. Uporabom segmentacije omogućilo bi se da se tijekom postavljanja slike u bazu podataka ona segmentira kako bi se korisniku omogućilo da tijekom pretraživanja baze podataka lakše pronađe željenu sliku. Pojmovi, odnosno ključne riječi slike ili kategorija, ne bi trebale biti unesene ručno već bi ih model tijekom segmentacije sam odredio. Proces tradicionalne segmentacije slika uključivao je grupiranje piksela (engl. *clustering*) te su se često uz grupiranje piksela upotrebljavale i informacije o konturama i rubovima. Danas se segmentacijske metode mogu podijeliti u tri glavne kategorije: semantička segmentacija temeljena na regiji (engl. *region-based semantic segmentation*), semantička segmentacija temeljena na potpuno povezanoj mreži (engl. *fully convolutional network-based semantic segmentation*) i slabo nadzirana semantička segmentacija (engl. *weakly supervised semantic segmentation*) (Guo et al., 2018).

## **2.1. Semantička segmentacija temeljena na regiji**

Metode semantičke segmentacije temeljene na regiji prate proces koji najprije uključuje prepoznavanje regija na slici. Nakon prepoznavanja regija one se klasificiraju, a tijekom testa se predikcije na razini regija pretvaraju u predikcije na razini piksela. Regije s CNN značajkama (engl. *convolutional neural network features*) jedna je od metoda te je usko vezana uz detekciju objekata, odnosno provođenje semantičke segmentacije ovisno je o rezultatima detekcije objekata. Segmentacijski proces stvara regije čije granice vrlo dobro prate granice objekta (Caesar, Uijlings i Ferrari, 2016). Proces ove metode je takav da se najprije selektivnim pretraživanjem traže mogući objekti za koje se izračunavaju CNN značajke te se za svaku regiju koristi linearni SVM (engl. *Support vector machine*) model u svrhu klasifikacije (Uijlings et al., 2013). Bitno

je napomenuti kako postoje tri glavna nedostatka korištenja ove metode u svrhu semantičke segmentacije:

1. značajka nije kompatibilna sa zadatkom semantičke segmentacije,
2. značajka ne sadrži dovoljno prostornih informacija za precizno generiranje granica,
3. generiranje prijedloga temeljenih na segmentima zahtijeva vrijeme i uvelike bi utjecalo na konačnu izvedbu.

## **2.2. Semantička segmentacija temeljena na potpuno povezanoj mreži**

Metode temeljene na potpuno povezanoj mreži (engl. *FCN-based*) omogućavaju da CNN kao ulazni podatak može primiti sliku proizvoljne veličine, dok klasični CNN ne može primiti sliku proizvoljne veličine jer sadržava potpuno povezani sloj (engl. *fully connected layer*) koji je fiksne veličine. Metode temeljene na potpuno povezanoj mreži sadržavaju samo konvolucijske (engl. *convolutional*) i objedinjavajuće slojeve (engl. *pooling layer*) te iz tog razloga mogu primati i stvarati predikcije na slikama proizvoljne veličine (Guo et al., 2018). Bitno je napomenuti kako se za danu sliku (ulazni podatak) određene veličine stvara izlazna slika (izlazni podatak) koja je iste veličine kao i dana slika pa je samim time prigodna za semantičku segmentaciju (Long, Shelhamer i Darrell, 2015), ali ne i za detekciju objekata (Ren et al., 2015).

Nedostatak takve metode je to što propagiranjem kroz više konvolucijskih i objedinjavajućih slojeva dovodi do toga da izlazne mape značajki (engl. *output feature maps*) budu niske kvalitete, što zatim rezultira nejasnim granicama objekta.

Neke od arhitektura temeljene na potpuno povezanoj mreži su: FPN (engl. *feature pyramid network*), DeepLabV3Plus i U-net.

### **2.2.1. FPN**

FPN je akronim za *Feature Pyramid Network*, odnosno Mreža piramida značajki. Ova mreža koristi piramidalnu hijerarhiju konvolucijskih mreža za stvaranje piramida značajki (Seferbekove et al., 2018). Piramide značajki su osnovne komponente u

sustavima prepoznavanja koji služe u svrhu detekcije objekata na različitim skalama. Koristi *top-down* arhitekturu kako bi se izgradile visokokvalitetne mape značajki na svim skalama, odnosno ova arhitektura kao ulazni podatak uzima sliku proizvoljne veličine te kao izlazni podatak daje proporcionalne mape značajki na svim skalama (Lun et al., 2017).

### 2.2.2. DeepLabV3Plus

DeepLab je serija mreža za semantičku segmentaciju kreirana od strane Google-a, a DeepLabV3Plus jedna je od tih kreiranih mreža. Arhitektura mreže je kompleksna, ali nema jasnih nedostataka kod korištenja za zadatak semantičke segmentacije. Enkoder i dekoder su dijelovi ove mreže, a u enkoderskom dijelu *backbone* mreža koristi se s većim konvolucijskim slojem kako bi se maksimiziralo receptivno područje i povećala sposobnost izvlačenja značajki. Značajke niže razine ulaze u dekoder dok su značajke visoke razine spojene s ASPP (engl. *Atrous Spatial Pyramid Pooling*) modulom u svrhu prikupljanja i globalnih povećanja semantičke informacije slike. Nakon toga se rezultat tog spajanja u dekoderu spaja s prethodno spomenutim značajkama niske razine te ona prolazi kroz niz konvolucijskih slojeva i slojeva za povećanje (engl. *upsampling*) kako bi izlazna slika imala istu rezoluciju kao ulazna (Wang et al., 2022).

### 2.2.3. U-net

U-net je arhitektura temeljena na FPN-u te je stvorena primarno za korištenje u medicinskim analizama, točnije za segmentaciju slika u medicinskim analizama. Iako se vrlo često koristi baš za zadatke u tom području, naprimjer za analizu CT (engl. *computerized tomography*) skeniranja, MRI (engl. *magnetic resonance imaging*) te mikroskopije ona se koristi i u drugim područjima, a sve u svrhu stvaranja dobrog modela semantičke segmentacije. Struktura ove arhitekture je simetrična te tvori slovo U, a može se podijeliti na dva glavna puta, prvi put je put analize ili enkoder, a drugi put je put sinteze ili dekoder. Put analize (enkoder) je zapravo regularna klasifikacijska mreža, dok se put sinteze (dekoder) sastoji od *upsampling* slojeva i značajki iz

prethodnog puta, a time se omogućava mreži da dobije informacije ne samo o klasifikaciji već i o lokalizaciji, odnosno mjestu na kojem se nalazi određeni objekt. Prvi put se još zove i skupljajući put (engl. *contracting path*), a drugi se još zove i proširujući put (engl. *expanding path*) (Siddique et al., 2020).

### **2.3. Slabo nadzirana semantička segmentacija**

Metode slabo nadzirane semantičke segmentacije koriste se kako bi se izbjegao veliki trošak prilikom stvaranja skupa podataka. One koriste oznake na razini slika i CAM (engl. *class activation map*) za semantičku segmentaciju. CAM se koristi kako bi se pronašle one značajke na slici koje su važne za stvaranje predikcija za određenu klasu, odnosno on označava regije na slici gdje se te značajke pojavljuju. Metode slabo nadzirane semantičke segmentacije prate proces koji najprije uključuje model za klasifikaciju na razini slike (slika se označava s jednom klasom), CAM zatim pronalazi ključne regije koje se zatim koriste kao početne oznake za potpuno nadzirane segmentacijske metode (Syed et al., 2023).

Stvaranje skupa podataka je financijski skup i dugotrajan proces, a jedan od procesa kod stvaranja skupa podataka je i anotiranje. Ono se provodi na svakoj slici, odnosno za svaki piksel, te je iz tog razloga to vrlo zahtjevan zadatak. Vodeći se time stvorene su metode koje omogućuju treniranje modela semantičke segmentacije sa skupom podataka koji ima oznake na razini slika. Svaka slika u trening skupu podataka sadržava oznake (klase) kojima pripada, no točna lokacija tog objekta na slici nije poznata. Iz tog je poznat i glavni problem koji se pojavljuje u ovakvom zadatku segmentacije: kako točno dodijeliti klase točnom pikselu na slici (Huang et al., 2018)? Dai, He i Sun (2015) su predložili korištenje graničnog okvira (engl. *bounding box*) kao tehniku nadziranja treninga te su ostvarili srednju prosječnu preciznost od 62% na validacijskom skupu podataka. Papandreou et al. (2015) su predložili tehniku očekivanje-maksimiziranje (engl. *expectation-maximization*) za treniranje. Usporedili su korištenje skupa podataka sa slikama koje imaju oznake na razini slika te korištenje graničnih okvira. Rezultati korištenja skupa podataka s oznakama na razini slika nisu bili zadovoljavajući za kvalitetni model semantičke segmentacije, dok je korištenje graničnih okvira rezultiralo dobrim i kompetitivnim modelom. Kombiniranjem obje

metode postižu se još bolji rezultati. Khoreva et al. (2017) su pristupili tom problemu kao problemu smetnje ulaznih podataka te su koristili rekurzivno treniranje kao metodu smanjenja smetnji. Koristeći takav pristup i postavljanje ulaznih oznaka iz graničnih okvira ostvarili su čak 95% točnosti u usporedbi s modelom koji je u potpunosti nadziran, a korištena je ista procedura treninga.



### 3. Funkcije gubitka

Funkcija gubitka jedna je od najvažnijih stavki kod kreiranja i treniranja modela jer se uz pomoć nje dobiva informacija o napretku povećanja performansi modela te se uz tu informaciju upravlja treniranjem modela (Jadon, 2020). Izbor neadekvatne funkcije gubitka rezultirati će smanjenom performansom modela (Wang et al., 2020).

Funkcije gubitaka se po Liu, Doroslovačkom i Loewu (2020) mogu podijeliti u tri kategorije: funkcije gubitaka temeljene na vjerojatnosti (engl. *probability-based loss functions*), funkcije gubitaka temeljene na energiji (engl. *energy-based loss functions*) i učenje temeljeno na metrici (engl. *metric-based learning*). Oni također navode kako funkcije gubitaka temeljene na vjerojatnosti nastoje povećati vjerojatnost točne predikcije s obzirom na istinu iz trening skupa podataka, a kao primjer funkcije gubitaka navode *Cross entropy* funkciju gubitka. Za funkcije gubitaka temeljene na energiji objašnjavaju kako one mjere prikladnost svake konfiguracije ulaznih podataka i oznaka, a vrijednost rezultata funkcije gubitka prikazuje kompatibilnost između vrijednosti ulaznih podataka i oznaka, LeCun et al. (2006) kao primjer daju *Energy loss* funkciju gubitka. Za funkcije gubitaka temeljene na metrici navode kako se oslanjaju na metriku udaljenosti ili sličnosti koja enkodira korelaciju i varijaciju varijabli, a kao primjer daju *Mean squared error* funkciju gubitka. Ma (2020) je podijelio funkcije za segmentaciju u sljedeće kategorije: gubitak temeljen na distribuciji (engl. *distribution-based loss*), gubitak temeljen na regiji (engl. *region-based loss*), gubitak temeljen na granici (engl. *boundary-based loss*) i složeni gubitak (engl. *compounded loss*).

Gubitak temeljen na distribuciji sadržava one funkcije gubitaka koje kao zadatak imaju smanjiti razliku između dvije distribucije, a temeljna funkcija gubitaka je *Cross entropy* koja je nastala iz Kullback-Leibler divergencije. Smanjenje Kullback-Leibler divergencije rezultira i smanjenjem rezultata *Cross entropy* funkcije gubitka. Od ove funkcije gubitaka nastale su i TopK te *Focal* funkcije gubitka. *Cross entropy* i *Focal* funkcija gubitka su detaljnije opisani u poglavlju 7.3.

Gubitak temeljen na regiji sadržava one funkcije gubitaka koje nastoje čim više smanjiti neusklađenost ili čim više povećati regije preklapanja između istinite i predviđene maske. Funkcije gubitaka koje spadaju u ovu kategoriju su, između ostalog, Dice

funkcija gubitka IoU te Tversky. Spomenute funkcije gubitaka su također detaljnije opisane u poglavlju 7.3.

Gubitak temeljen na granici sadržava funkcije gubitaka koje su slične kao kod onih u kategoriji gubitka temeljenog na regiji, ali je razlika u tome da one koje pripadaju ovoj kategoriji nastoje minimizirati udaljenost između istinite i predviđene maske. Neke od funkcija gubitaka koje spadaju u ovu kategoriju su i *Boundary* te Hausdorff udaljenost.

Složeni gubitak sadržava funkcije gubitka koje se sastoje od više funkcija gubitaka (Ma, 2020).

## 4. Evaluacijske mjere

Evaluacijske mjere se često koriste u strojnom učenju u svrhu procjene uspješnosti modela. Njihovim korištenjem omogućava se uvid u napredak treniranja, a samim time omogućava se podešavanje hiperparametara kako bi se poboljšao trening te spriječilo pretreniranje. Neke od najčešće korištenih mjera ću opisati u nastavku poglavlja. Matrice zabune (engl. *confusion matrices*) koriste se za procjenu greške u problemu klasifikacije te je njihova uporaba tipična za nadzirano strojno učenje. Ovisno o ishodu klasifikacije razlikujemo četiri različita rezultata: istinito pozitivni (engl. *true positive*), istinito negativni (engl. *true negative*), lažno pozitivni (engl. *false positive*) i lažno negativni (engl. *false negative*). Istinito pozitivni označavaju točno klasificirane slučajeve, istinito negativni označavaju ispravno odbačene slučajeve, lažno pozitivni označavaju neklasificirane ili propuštene slučajeve te lažno negativni označavaju klasificirane slučajeve, ali pogrešno (Beauxis i Hardman, 2014). Goutte i Gaussier (2005) su kreirali sljedeću tablicu:

|            |   | Dodjela $z$ |    |
|------------|---|-------------|----|
|            |   | +           | -  |
| Oznaka $l$ | + | TP          | FN |
|            | - | FP          | TN |

Tablica 1. Konfuzijska matrica (Izvor: Goutte i Gaussier, 2005)

Tablica 1 prikazuje konfuzijsku matricu u kojoj simbol “+” predstavlja relevantno, a simbol “-“ nerelevantno. Sljedeće evaluacijske mjere imaju polazište iz konfuzijske matrice: preciznost (engl. *Precision*), odziv (engl. *Recall*), F1 mjera (engl. *F1 score*) (Beauxis i Hardman, 2014).

### 4.1. Preciznost

Preciznost ili povjerenje (engl. *Confidence*) se može definirati kao udio ispravno klasificiranih slučajeva u sveukupnom skupu pozitivno klasificiranih (Powers, 2020). Također se može definirati i kao udio slučajeva koje sustavi vrati, a točno su klasificirani (Derczynski, 2016).

Goutte i Gaussier (2005) su kreirali formulu za preciznost koja je prikazana na Jednadžbi 1.

$$p = \frac{TP}{TP + FP}$$

Jednadžba 1. Preciznost (Izvor: Goutte i Gaussier, 2005)

Korištenje ove evaluacijske mjere u semantičkoj segmentaciji prikazuje proporciju točno predviđenih piksela (TP) u odnosu na sve piksele koje je model predvidio kao pozitivne (TP + FP).

## 4.2. Odziv

Odziv ili osjetljivost (engl. *Sensitivity*) se može definirati kao udio ispravno klasificiranih slučajeva u sveukupnom skupu stvarno pozitivnih (Powers, 2020). Također se može definirati i kao udio slučajeva koji bi trebali biti pronađeni (Derczynski, 2016).

Goutte i Gaussier (2005) su kreirali formulu za odziv koja je prikazana na Jednadžbi 2.

$$r = \frac{TP}{TP + FN}$$

Jednadžba 2. Odziv (Izvor: Goutte i Gaussier, 2005)

Korištenje ove evaluacijske mjere u semantičkoj segmentaciji slično je kao i korištenje preciznosti, jedina razlika je u tome što odziv prikazuje proporciju točno predviđenih piksela (TP) u odnosu na sve piksele koji su stvarno pozitivni (TP + FN), a ne one koje je model predvidio kao pozitivne.

## 4.3. F1 mjera

F1 mjera ili F mjera je izvedena iz preciznosti i odziva te njena skala varira od nula do jedan (Derczynski, 2016). Ona preferira preciznost kada je hiperparametar beta ( $\beta$ ) veći od jedan, a u ostalim slučajevima preferira odziv (Sokolova, Japkowicz i Szpakowicz, 2006).

Goutte i Gaussier (2005) su kreirali formulu za F1 mjeru koja je prikazana na Jednadžbi 3.

$$F_{\beta} = (1 + \beta^2) \frac{pr}{r + \beta^2 p} = \frac{(1 + \beta^2)TP}{(1 + \beta^2)TP + \beta^2 FN + FP}$$

*Jednadžba 3. F1 mjera (Izvor: Goutte i Gaussier, 2005)*

Korištenje ove evaluacijske mjere u semantičkoj segmentaciji obuhvaća preciznost i odziv te se daje harmonijska sredina obje mjere, a to je korisno kod neuravnotežene distribucije klasa koja se često može dogoditi na slikama prometnica. Jednadžba 3 prikazuje formulu F1 mjere u kojoj p označava preciznost, r odziv, a  $\beta$  označava betu koja se koristi za upravljanje preferencijom između preciznosti i odziva. Nakon znaka jednakosti prikazana je ista formula koristeći rezultate iz konfuzijske matrice (TP, FN i FP) te beta hiperparametar.

#### 4.4. IoU mjera

*Intersection over Union* ili Jaccard indeks se koristi za usporedbu dva skupa u strojnom učenju, točnije za usporedbu sličnosti dva skupa te je jedan od bitnijih evaluacijskih matrica između ostalog i u semantičkoj segmentaciji. Također se može definirati i kao preklapanje između dva različita skupa (Yu et al., 2021).

(Yu et al., 2021) su kreirali formulu za IoU mjeru koja je prikazana na Jednadžbi 4.

$$IoU = \frac{A \cap B}{A \cup B}$$

*Jednadžba 4. IoU mjera (Izvor: Yu et al., 2021)*

Korištenje ove evaluacijske mjere u semantičkoj segmentaciji prikazuje preklapanje između objekta na predviđenoj i istinitoj maski. Na Jednadžbi 4 A predstavlja piksele objekta predviđene maske, a B predstavlja piksele objekta na istinitoj maski. Formula prikazuje omjer piksela gdje se predviđena i istinita maska preklapaju ( $A \cap B$ ) te svih piksela objekta koje je model predvidio i koji su dio istinite maske ( $A \cup B$ ).

## 5. Tehnike augmentacije podataka

Augmentacija podataka je povećanje obujma skupa podataka s pretpostavkom da se više informacija može izvući iz originalnog skupa podataka. Povećavanje generalizacije (engl. *generalizability*) je jedan od zadataka kod računalnog vida, a odnosi se na razliku u performansi modela kada se evaluira na podacima koje je ranije vidio (trening podacima) u odnosu na podatke koje ranije nije vidio (validacijski ili test podaci). Razlog niske razine generalizacije je pretreniranje (engl. *overfitting*). Augmentacija podataka uspješno smanjuje pretreniranost, odnosno greška na validacijskom skupu podataka se mora smanjivati zajedno s greškom na trening skupu. Augmentirani podaci omogućuju opsežniji skup što rezultira smanjenjem udaljenosti dva skupa (trening i validacijskog skupa) te samim time i nekog budućeg skupa za testiranje. Tehnike augmentacije podataka se mogu podijeliti u dvije kategorije: iskrivljenje podataka (engl. *data warping*) i prekomjerno uzorkovanje (engl. *oversampling*).

Kategorija iskrivljenje podataka sadržava one augmentacijske tehnike kod kojih se oznaka (klasa) sačuva te se transformiraju već postojeće slike. Neke od takvih tehnika su nasumično brisanje i inverzija boja.

Kategorija prekomjerno uzorkovanje sadržava one augmentacijske tehnike koje kreiraju nove instance te ih dodaju u skup podataka. Neke od takvih tehnika su miješanje slika i povećanje značajki prostora (engl. *feature space augmentations*) (Shorten i Khoshgoftaar, 2019).

PyTorch je biblioteka koja omogućuje ubrzano razvijanje modela dubokog učenja (Paszke et al., 2017). Na internetskim stranicama Pytorch-a (pristupljeno 16. travnja 2023.) navodi se kako biblioteka također nudi paket Torchvision koji sadržava razne tehnike za augmentaciju podataka. Tehnike augmentacije korištene u radu su detaljno opisane u poglavlju 8.3.

## 6. CARLA simulator

CARLA simulator je simulator otvorenog koda koji je stvoren u svrhu treniranja, validacije i kreiranja sustava za autonomnu vožnju. Simulator sadržava mnoge statičke i dinamičke objekte koji se mogu susresti u svakodnevnom prometu, naprimjer zgrade, vozila, pješaci, cesta, prometni znakovi i mnogi drugi objekti. Simulator također nudi mnogo kontrola za upravljanje ostalim vozilima pa se tako može postaviti praćenje linija, poštivanje prometnih znakova i oznaka te održavanje propisanog ograničenja brzine. Dinamički objekti (vozila i pješaci) su kreirani na način da se mogu prepoznati i izbjegavati. Radi diverzifikacije između objekata, kreiranje pješaka nasumično izabire odjevne predmete iz predefinirane kolekcije odjevnih predmeta te se nasumično dodaju neki objekti pješacima (naprimjer pametni mobitel, torbe za trgovinu, kofer za gitaru, putni kofer i slično). Vozila su također drugačija te sadržavaju druge boje. Radi boljeg skupa podataka moguće je dodavanje i drugih varijabilnih elemenata poput doba dana te vremenskih uvjeta, naprimjer kiše, magle, sunca i slično.

Simulator nudi mnoge vrste kamere koje se mogu postaviti na agenta (vozilo korisnika), kamere koje su korisne za kreiranje skupa podataka za zadatak ovog rada su kamera s crvenim, zelenim i plavim (RGB) kanalima te kamera semantičke segmentacije. Kamerama se mogu namjestiti parametri: pozicija, orijentacija, rezolucija, tip te vidno polje (Dosovitskiy et al., 2017). Kamera semantičke segmentacije odvaja sve objekte u klase, a raspon klasa u koje će se spremi ovisi o broju dostupnih klasa u verziji CARLA-e.

Klase su spremljene u crvenom kanalu slike na točnom mjestu na kojem se nalazi piksel koji pripada toj određenoj klasi. CARLA simulator trenutno sadržava dvadeset i tri klase (od kojih je jedna nedefinirana) te ukoliko kamera spremi sliku koja sadržava sve dvadeset i tri klase znači da će slika sadržavati prazni zeleni i plavi kanal, a u crvenom će se pojavljivati vrijednosti od nule do dvadeset i dva. U kanalima se pojavljuju vrijednosti do dvjesto pedeset i pet, veća brojka označava veći intenzitet boje na tom pikselu. Kako su u maski vrijednosti do dvadeset i dva, takva slika neće biti prikazana jasno, već će se činiti crnom zbog niske razine vrijednosti (intenziteta) u crvenom kanalu. CARLA simulator iz tog razloga nudi funkciju *cityScapesPalette* koja za svaku klasu iz crvene klase pretvara tu vrijednost u druge vrijednosti u crvenom,

zelenom i plavom kanalu kako bi ljudsko oko moglo jasnije razumjeti sliku semantičke segmentacije, odnosno masku semantičke segmentacije.

## 6.1. Prikupljanje slika za skup podataka

Za prikupljanje skupa podataka najprije je potrebno pokrenuti CARLA simulator, a nakon pokretanja mogu se postaviti svi varijabilni elementi. CARLA nudi mogućnost upravljanja raznim elementima, ali također nudi i *generate\_traffic.py* dokument koji ima predefinirane elemente. U spomenutom dokumentu su postavljeni samo varijabilni elementi, ali ne i kamere kojima se može prikupiti skup podataka. Iz tog razloga je u ovom radu dorađena skripta na način da se CARLA pokrene na sinkroni način kako bi se mogla iskoristiti funkcija *tick* koju ona nudi. Tom funkcijom moguće je zaustaviti CARLA-u da čeka na odgovor klijenta, a to je iskorišteno kako bi se CARLA zaustavila dok nisu obadvije kamere spremile slike na istoj lokaciji. Bez uporabe ove funkcije, kamere ne bi uvijek spremile iste slike, a to bi rezultiralo jako lošim skupom podataka jer istinita maska ne bi odgovarala slici.

```
# Postavljanje kamere za sliku
camera_bp = blueprint_library.find('sensor.camera.rgb')
camera_transform = carla.Transform(carla.Location(x=1.5, z=2.4))
camera = world.spawn_actor(camera_bp, camera_transform, attach_to=vehicle)
image_queue = queue.Queue()
camera.listen(image_queue.put)
actor_list.append(camera)

# Postavljanje kamere za masku
camera_semseg = blueprint_library.find('sensor.camera.semantic_segmentation')
camera_transform = carla.Transform(carla.Location(x=1.5, z=2.4))
camera_seg = world.spawn_actor(camera_semseg, camera_transform, attach_to=vehicle)
image_queue_seg = queue.Queue()
camera_seg.listen(image_queue_seg.put)
actor_list.append(camera_seg)

for i in tqdm.tqdm(range(args['num_frames'])):
    world.tick()
    image = image_queue.get()
    image_seg = image_queue_seg.get()
    if i%args['sample_every']==0:
        image.save_to_disk("dataset/images/%s_%06d.png" %(args['name'], image.frame))
        image_seg.save_to_disk("dataset/masks/%s_%06d.png" %(args['name'], image_seg.frame), carla.ColorConverter.CityScapesPalette)
    image_queue.queue.clear()
    image_queue_seg.queue.clear()
```

Slika 2: Kod za generiranje skupa podataka

Na Slici 2 vidljiv je dio koda gdje se stvaraju kamere za sliku i masku, da nakon njihovog kreiranja potrebno je postaviti kameru na vozilo i postaviti ih u red. Tijekom prolaska po svakoj sličici (engl. *frame*) iz reda se uzimaju stvorene kamere te se njima kreiraju slike koje se spremaju u zasebne datoteke u skup podataka. Nakon uporabe se kamere brišu iz memorije kako ne bi došlo do curenja memorije (engl. *memory leak*).



U sljedeća dva poglavlja nalaze se znanstveni radovi koji opisuju različite funkcije gubitaka i tehnike za augmentaciju podataka.

## 7. Usporedba funkcija gubitka za semantičku segmentaciju objekata u prometu<sup>1</sup>

Autonomna vožnja postaje sve važniji čimbenik svakodnevnog života, čime raste značaj dubokog učenja na kojem se takvi sustavi temelje. Posljednjih godina predložene su mnoge funkcije gubitka za problem semantičke segmentacije, čiji odabir ima direktan utjecaj na sposobnost modela da uči i uspješno generira predikcije. Ovo istraživanje analizira utjecaj različitih funkcija gubitka na performanse semantičke segmentacije na 23 različita objekta prisutna u prometu. Istraživanje je provedeno na skupu podataka prikupljenom na CARLA simulatoru prometa u urbanom okruženju. Istraživanje obuhvaća sve najčešće korištene funkcije gubitka: *Binary Cross entropy*, *Dice*, *Focal*, *Jaccard* te *Tversky*, kako bi analizirali njihov utjecaj na konvergenciju modela. Usporedba je izvršena na tri različite arhitekture mreža: U-net, DeepLabV3Plus i FPN. Istraživanje donosi preporuke prikladnosti pojedinih funkcija na problemu semantičke segmentacije objekata u prometu.

### 7.1. Uvod

Danas sve više proizvođača automobila stavlja fokus na autonomnu vožnju, koja se brzo razvija i obećava revoluciju u životima pojedinaca. Stvaranje autonomnog sustava vožnje uključuje ne samo kontrolu i upravljanje vozilom, nego zahtijeva da vozilo i njegov autonomni sustav vožnje precizno percipiraju okolinu, očekuju i predviđaju različite pojave i događaje na cesti te donose brze i točne odluke u stvarnom vremenu, kako bi putnicima i drugim sudionicima u prometu omogućili sigurno putovanje. U rješavanju izazova autonomne vožnje trenutno se u svijetu nameću modeli dubokog učenja. Takvi modeli su sposobni predvidjeti različita ponašanja vozila i okoline s iznimnom točnošću. Duboko učenje i računalni vid zahtijevaju korištenje skupa podataka koji je dovoljno velik i sadrži različite scenarije, tako da model može naučiti razlikovati različite obrasce i pojave na cesti. Stvaranje skupa podataka je dugotrajan i skup proces. To se pogotovo odnosi na označavanje podataka za treniranje modela

---

<sup>1</sup> Poglavlje prezentirano kao znanstveni rad na MIPRO 2023 konferenciji (Oreški i Aničić, 2023)

s nadziranim učenjem (Wang et al., 2020). CARLA simulator za gradsku vožnju predstavlja alternativu klasičnom načinu prikupljanju podataka. Simulator nudi niz različitih elemenata u svakodnevnom prometu, kao što su: pješaci, vozila, zgrade, ulični znakovi te drugi elementi. Također nudi mogućnost promjene doba dana, vremena, senzora, brzine, ubrzanja i drugih varijabilnih elemenata (Yuan i Xu, 2022). Najveća prednost takvih simulatora je mogućnost automatskog generiranja labela, čime se značajno ubrzava proces stvaranja skupa podataka. U ovom istraživanju smo iskoristili CARLA simulator da bi istražili performanse različitih funkcija gubitka na problemu semantičke segmentacije u prometu. Funkcija gubitka je osnovni element svakog modela. Ključna je u procesu treninga te stoga njezin odabir predstavlja bitnu odluku.

## 7.2. Pregled literature

Iako je problem autonomne vožnje danas vrlo popularan postoji malo literature vezane uz funkcije gubitka, pogotovo za semantičku segmentaciju. Jedan od razloga može biti ranije spomenuta cijena prikupljanja takvih skupova podataka. U nastavku donosimo istraživanja koja su povezana s navedenom temom. S. Malec je istražio semantičku segmentaciju za treniranje modela autonomne vožnje proširenu s podacima generiranim iz CARLA simulatora. U istraživanju je korištena isključivo *Cross entropy* funkcija gubitka (Malec, 2021). M. Boudissa et al. su korištenjem U-net modela nastojali stvoriti model za korištenje u prometu, *odnosno* prepoznavanje objekata te su koristili dvije funkcije gubitka: *Dice* i *Focal*. *Dice* se pokazao kao bolji izbor zbog performansi detekcije rubova objekata (Boudissa et al., 2022). O. Natan i J. Miura su uz pomoć CARLA simulatora razvili model za semantičku segmentaciju i procjenu dubine. Tijekom treninga semantičke segmentacije koristili su *Dice* funkciju kombiniranu s *pixel-wise* binarnom *Cross-entropy* funkcijom u svrhu postizanja veće raznolikosti pogreške (Natan i Miura, 2022). F. Nesti et al. su usporedili standardnu *Cross entropy* funkciju s vlastitom, nadograđenom funkcijom koja bolje sprječava *Adversarial Patch* napade tijekom semantičke segmentacije (Nesti et al., 2022).

J. Ma et al. su analizirali veliki broj funkcija gubitka za semantičku segmentaciju (više od 10) na medicinskim slikama. U istraživanju se zaključuje da su najrobustnije

složene funkcije gubitka između kojih je i Dice funkcija (Ma et al., 2021). M. Yeung et al. su predložili novu funkciju, *Unified Focal*, koja generalizira dvije najčešće funkcije gubitka za zadatak segmentacije: Dice i *Cross-entropy*. Istraživanje je također provedeno na medicinskim slikama (Yeung et al., 2022). H Ghandorh et al. su istražili način kako uz pomoć semantičke segmentacije prepoznati cestu na visokokvalitetnim satelitskim slikama, kao funkciju gubitka koristili su težinsku *Cross entropy* te *Focal Tversky* funkciju gubitka (Ghandorh et al., 2022). J. Su et al. su usporedili različite funkcije gubitaka za nebalansiranu semantičku segmentaciju te su predložili *Cross Dropout Focal* funkciju gubitka. Koristeći istu ostvareno je 2.5% veća točnost u usporedbi sa standardnom *Cross-entropy* funkcijom (Su et al., 2022). H.X. Cheng et al. su predložili novi model koji smanjuje računsku i vremensku kompleksnost semantičke segmentacije. Koristili su kombinaciju funkcija gubitka: težinski *Cross-entropy*, Lovasz-Softmax te *Boundary*. Kombinacijom su nastojali riješiti izazove: nebalansiranih podataka, optimizacije *Intersection over Union* (IoU) mjere i nejasnih granica segmentacije (Cheng et al., 2022). S. Jadon je napravio iscrpnu analizu nekoliko funkcija gubitaka te ih usporedio na primjeru semantičke segmentacije. Identificirao je u kojim situacijama je jedna prikladnija od druge. Istraživanje je proveo na medicinskim slikama (Jadon, 2020). Ovim istraživanjem donosimo sustavnu usporedbu različitih funkcija gubitka za semantičku segmentaciju na tri najpopularnije arhitekture modela. Na podacima iz prometa smo ispitali njihovu prikladnost i donijeli preporuke za korištenje. U analizu smo uključili najpopularnije funkcije spomenute u pregledu literature.

### **7.3. Funkcije gubitka**

Algoritmi dubokog učenja počivaju na mjeri koja određuje kvalitetu njihove izvedbe, odnosno koja računa veličinu pogreške između predviđenih i stvarnih (ciljnih) podataka. Funkcija gubitka predstavlja taj izračun te se koristi kao polazište za prilagođavanje parametara modela kako bi se smanjila pogreška te poboljšale performanse modela. Izbor funkcije gubitka može uvelike utjecati na performanse algoritma, samim time neprikladna funkcija će negativno utjecati na performanse i konvergenciju modela. U nastavku donosimo kratki opis funkcija obuhvaćenih ovim

istraživanjem. Uz svaku funkciju dana je i njezina formula, čije detaljno objašnjenje se može pronaći u referenciranim istraživanjima.

### 7.3.1. *Cross entropy*

*Cross entropy* funkcija gubitka je definirana kao negativni logaritam predviđene vjerojatnosti za ispravnu klasu (Li et al., 2020) te je najčešće korištena funkcija gubitka za treniranje klasifikacijskih modela. Funkcija se pokazala uspješnom na brojnim problemima te se često koristi za semantičku segmentaciju. Njezin nedostatak je što sve piksele tretira podjednako te veliki objekti dominiraju mrežom stoga su performanse na malim objektima loše. Formula kategorijske *Cross entropy* funkcije za C klasa je dana Jednadžbom 5 gdje je  $t_i$  oznaka istinite klase a  $s_i$  Softmax vjerojatnost za  $i$ -tu klasu.

$$CE = - \sum_i^C t_i \log(f(s_i))$$

*Jednadžba 5. Cross entropy funkcija (Izvor: izrada autora prema Jadon, 2020)*

*Binary Cross entropy* je varijanta navedene funkcije koja se koristi u binarnoj klasifikaciji.

$$BCE = - \sum_i^{C=2} t_i \log(f(s_i))$$

*Jednadžba 6. Binary Cross entropy funkcija (Izvor: izrada autora prema Jadon, 2020)*

### 7.3.2. *Dice*

Dice funkcija gubitka bazirana je na Sørensen–Dice koeficijentu ili Tversky indeksu (Kervadec et al., 2019). Lažno pozitivni i lažno negativni primjeri su jednake važnosti te je ona imunija na problem nebalansiranosti (Li et al., 2019). Postoji nekoliko nedostataka, odnosno ograničenja kod treniranja neuronskih mreža s ovom funkcijom: neosjetljivost na udaljenost regija koje se ne preklapaju (Rezatofighi et al., 2019), neosjetljivost na granice objekata (Dolz et al., 2018) te osjetljivost na male difuzne lezije (Guizard et al., 2015; Harmouche et al., 2014; Styner et al., 2008; Wong et al.,

2018; Lucas et al., 2018). Formula Dice funkcije dana je Jednadžbom 7 i 8 gdje  $A$  predstavlja istinitu masku, a  $B$  predviđenu masku. Presjek maski se računa kao umnožak po elementima (engl. *element-wise*) te se potom sumira rezultirajuća matrica.

$$Dice = 2 \cdot \frac{|A \cap B|}{|A| + |B|}$$

*Jednadžba 7. Dice funkcija (Izvor: Izrada autora prema Wang et al., 2020)*

$$DiceLoss = 1 - Dice$$

*Jednadžba 8. Dice funkcija gubitka (Izvor: Izrada autora prema Wang et al., 2020)*

### 7.3.3. Jaccard/Intersection over union (IoU)

Jaccard indeks, odnosno IoU gubitak daje rezultat u rasponu  $[0, 1]$  za presjek između istinite i predviđene maske preko njihove unije s pretpostavkom da je  $0/0$  jednako 1 (Berman et al., 2018). IoU rezultat želimo maksimizirati dok IoU gubitak želimo minimizirati tijekom treniranja. Nedostatak funkcije su loše performanse na malim objektima segmentacije. U tim slučajevima nema varijacije indeksa i ti objekti ostaju nezapaženi tijekom treninga (Kamal et al., 2019). Formula za Jaccard/IoU funkciju gubitka dana je Jednadžbom 8. Oznake  $A$  i  $B$  imaju isto značenje kao u formuli 7.

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}$$

*Jednadžba 9. Jaccard/Intersection over union funkcija (Izvor: Izrada autora prema Zhou et al., 2019)*

### 7.3.4. Focal

*Focal* je varijacija *Cross entropy* funkcije. Autori su predložili dodavanje faktora  $(1-p_t)^\gamma$  standardnoj funkciji gubitka *Cross entropy* te se postavljanjem  $\gamma > 0$  može smanjiti gubitak na ispravno predviđenom skupu primjera (Lin et al., 2017). Funkcija dijeli primjere u lagane i teške te pomoću faktora povećava vrijednost gubitka za teške,

odnosno smanjuje ga za lagane primjere. Funkcija gubitka klasificira svaki piksel pojedinačno što rezultira time da pogrešna anotacija značajno utječe na performanse mreže (Wang et al., 2020). Navedeni problem nije značajan kada se koriste podaci generirani iz simulatora. Formula *Focal* funkcije dana je Jednadžbom 10 gdje je  $p_t$  predviđena vjerojatnost ispravne klase,  $\gamma$  parametar koji kontrolira stupanj za koji će gubitak biti smanjen za ispravno klasificirane primjere a  $\alpha$  težinski faktor za svaki primjer.

$$FL(p_t) = -\alpha(1 - p_t)^\gamma \log(p_t)$$

*Jednadžba 10. Focal funkcija (Izvor: Jadon, 2020)*

### 7.3.5. Tversky

Tversky proširuje Dice funkciju uvođenjem dvaju koeficijenata koji kontroliraju osjetljivost gubitka na lažno pozitivne i lažno negativne rezultate. Koeficijenti  $\alpha$  i  $\beta$  omogućuju određivanje kompromisa između preciznosti i odziva u izračunu gubitka. To rezultira fleksibilnijom funkcijom gubitka u usporedbi s gubitkom Dice funkcije budući da se može bolje nositi s neuravnoteženim skupovima podataka i smanjiti utjecaj klasa s velikim brojem piksela (Gabor, 2022). Tversky funkcija gubitka je dana Jednadžbom 11 gdje TP označava istinito pozitivne piksele, FP lažno pozitivne, FN lažno negativne a  $\alpha$  i  $\beta$  su ranije spomenuti koeficijenti.

$$T_{\alpha,\beta}(y, \hat{y}) = \frac{TP}{TP + \alpha FP + \beta FN}$$

*Jednadžba 11. Tversky funkcija (Izvor: Izrada autora prema Salehi et al., 2017)*

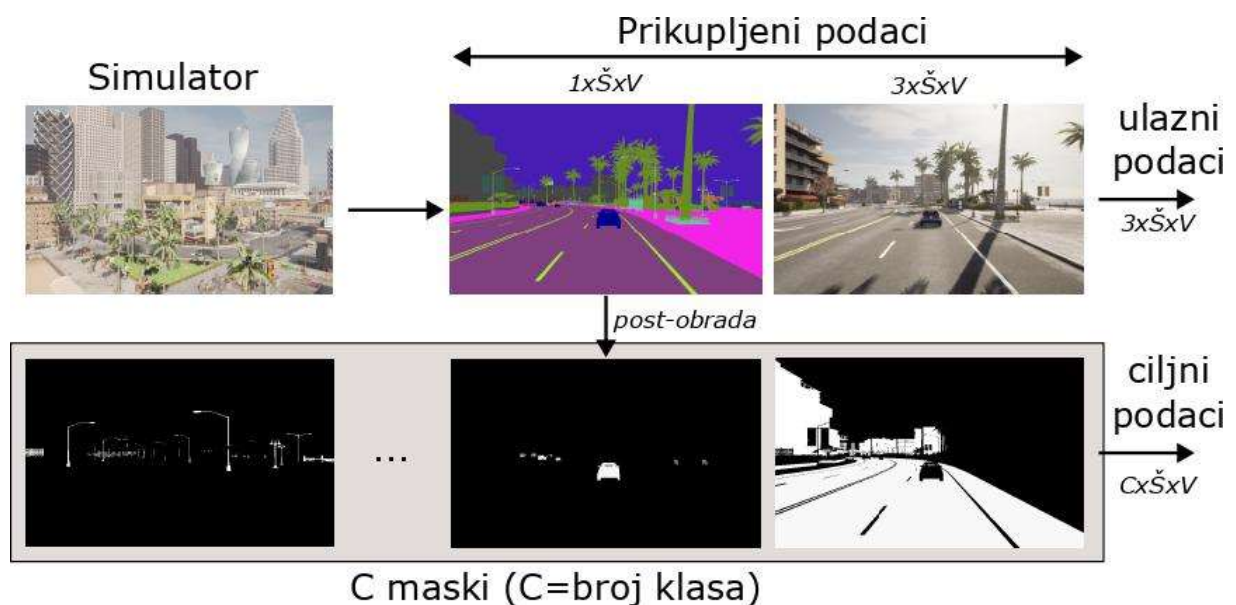
## 7.4. Eksperiment i rezultati analize

Da bi dali sustavan prikaz utjecaja funkcija gubitka na semantičku segmentaciju prometa, u istraživanje su uključene tri arhitekture najčešće korištene u literaturi što ukupno daje 15 različitih modela koje smo vrednovali koristeći četiri mjere uspješnosti

predikcije: IoU, F mjera, preciznost i odziv. Detaljan opis eksperimenta zajedno s rezultatima je dan u nastavku.

#### 7.4.1. Skup podataka iz simulatora CARLA

Skup podataka je prikupljen koristeći CARLA simulator autonomne vožnje. Za generiranje podataka kreirana su dva različita senzora, jedan za generiranje RGB slike, a drugi za generiranje semantičke segmentacije. Trenutačna verzija CARLA-e sadržava 23 različite klase ( $C=23$ ). Senzor za semantičku segmentaciju označava klase svakog piksela oznakama od 1 do 23 u crvenom (R) kanalu slike, dok su ostala dva kanala (G i B) prazna, odnosno sadržavaju vrijednost 0. Koristeći funkciju *cityScapesPalette* ta slika se može transformirati iz slike s vrijednosti samo u crvenom kanalu u sliku s vrijednostima u svim kanalima te se grafički lijepo prikazati. Za generiranje prometa koristila se skripta *generate\_traffic.py*, a za postavljanje Ego vozila (vozila s kamerom), pokretanje senzora te autopilot, odvojena, vlastita skripta.



Slika 3. Generiranje skupa podataka

Slika 3 prikazuje dohvaćanje podataka iz simulatora. Podaci sa senzora se dohvaćaju u paru te sadržavaju RGB sliku i masku u jednom kanalu (koja se iscrtava kao RGB sa spomenutom *cityScapesPalette* funkcijom). Slika je veličine  $3 \times \text{Širina}(\text{Š}) \times \text{Visina}(V)$ , a maska  $1 \times \text{Š} \times V$ . U post-obradi tj. pripremi ciljnih podataka, maska se tehnikom *one-hot encoding* dijeli na C zasebnih kanala u kojima se nalaze odvojene maske svake klase. Na Slici 3 su odvojene maske vidljive kao crno-bijele slike gdje



bijeli pikseli prikazuju prisutnost pojedine klase. Modelu se tijekom treniranja kao ulazni podatak daje slika i maska veličine  $C \times \check{S} \times V$ . Model predviđa masku za svaku pojedinu klasu na zasebnom kanalu. Da bi se nakon predikcije došlo do inicijalne dimenzije maske ( $1 \times \check{S} \times V$ ) izlaz se agregira koristeći *argmax* funkciju po prvoj dimenziji. Prikupljeno je 11000 slika, koje su raspodijeljene u omjeru 70/20/10 u trening, validacijski i testni skup respektivno.

#### 7.4.2. Modeli

Ovo istraživanje analizira pet različitih funkcija gubitaka koje su korištene za trening u kombinaciji s tri različite arhitekture mreže. Kao osnovna mreža (engl. *backbone*) odabran je Resnext50, a korištene težine su pred-trenirane na Imagenet skupu podataka. Za arhitekture smo koristili: FPN (*Feature Pyramid Network*) (Lin et al., 2017), DeepLabV3+ (Chen et al., 2018) te U-net (Ronneberger et al., 2015.). FPN arhitektura se koristi za izdvajanje značajki slike (engl. *feature extractor*). Kao ulaz prima sliku proizvoljne veličine i generira mape značajki proporcionalne veličine na više razina. DeepLabV3Plus koristi modul *Atrous spatial pyramid pooling* (ASPP) za izdvajanje kontekstualnih informacija na više razina (engl. *multi-scale*) iz mapa značajki konvolucijske neuronske mreže. U-net je enkoder-dekoder tip arhitekture koji prvo uz pomoć enkodera izdvaja značajke te potom dekoderom rekonstruira segmentacijsku mapu.

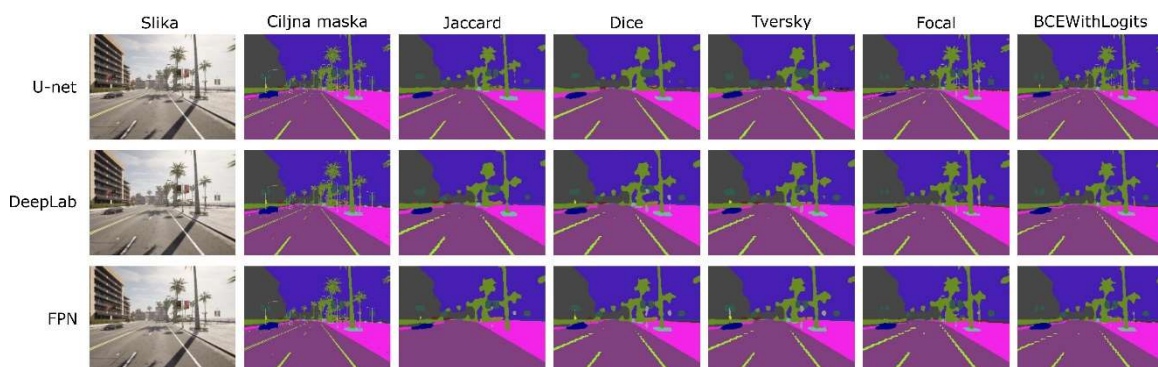
#### 7.4.3. Evaluacijske mjere

Evaluacijske mjere se koriste za procjenu uspješnosti modela, a mi ih koristimo za usporedbu prikladnosti odabranih funkcija gubitka. Mnoge mjere imaju polazište u matrici konfuzije (engl. *confusion matrix*). U matrici postoje 4 moguća ishoda predikcije: istinito pozitivna, istinito negativna, lažno pozitivna te lažno negativna. Različite mjere predstavljaju različite odnose tih ishoda. U nastavku su opisane korištene evaluacijske mjere u istraživanju. Preciznost je udio točno klasificiranih piksela u skupu pozitivno klasificiranih piksela. Odziv je udio točno klasificiranih piksela u skupu svih pozitivnih piksela. F-mjera je harmonijska sredina preciznosti i odziva. IoU mjera je udio ispravno

klasificiranih piksela klase u skupu svih piksela koji toj klasi pripadaju ili su svrstani u nju.

#### 7.4.4. Rezultati

Trening modela je zaustavljan kada je vrijednost funkcije gubitka prestala padati na validacijskom skupu podataka. Konačni rezultati uspješnosti modela na testnom skupu podataka, iskazani ranije spomenutim mjerama, su vidljivi u Tablici 2. Na Slici 3 je dan usporedni prikaz predikcije modela za isti ulazni podatak. Analizirane su performanse pojedinih modela kroz funkcije gubitka, a potom i korištenu arhitekturu.



Slika 4. Prikaz predikcija svih modela na istoj slici

Iz Tablice 2 je vidljivo da je *Binary Cross entropy* funkcija ostvarila najbolje rezultate mjereno F mjerom, IoU mjerom i preciznosti predikcije. U Poglavlju 7.3. je istaknuto da je to najčešće korištena funkcija te da ostvaruje dobre rezultate na semantičkoj segmentaciji što ovo istraživanje potvrđuje. Štoviše, iz Slike 4 je vidljivo da su i manji objekti prepoznati, što se često navodi kao glavni nedostatak te funkcije. Rezultati pokazuju da je *Binary Cross entropy* robusna funkcija te da se može koristiti s bilo kojom arhitekturom. *Focal* funkcija gubitka ima vrlo slične rezultate kao i *Binary Cross entropy*, što dokazuje da se radi o njezinoj varijanti koja dodaje jedan termin. Obzirom da je *Focal* funkcija parametrizirana s  $\gamma$ , kojim se određuje koliko će biti smanjen utjecaj laganih primjera na vrijednost funkcije, njegovim mijenjanjem se može utjecati na konačan ishod. U eksperimentu smo zadržali zadanu vrijednost  $\gamma=2$  koju autori navode kao najbolji odabir. *Focal* funkcija je konzistentno kroz sve arhitekture imala nešto bolju preciznost i manji odziv od *Cross entropy* funkcije. Ako usporedimo dvije funkcije na Slici 4, vidimo da udaljene linije između prometnih traka (manji objekti) bolje prepoznaje *Focal* funkcija, dok automobil bolje prepoznaje *Cross entropy*. Dice funkcija

i njezino proširenje Tversky ostvaruju slične rezultate, koji su lošiji od ranije spomenutih tehnika. Ukoliko se koristi U-net ili FPN tada Dice postiže bolje rezultate, dok s DeepLabV3Plus je obrnuto. Najveća razlika je vidljiva na U-net mreži gdje Dice ostvaruje značajno bolji rezultat. Za razliku od prošle dvije tehnike, ove obje imaju veći odziv nego preciznost, bez obzira na korištenu mrežu. Vizualnom analizom predikcija sa Slike 4 je vidljivo da obje funkcije postižu lošije rezultate od *Cross entropy* i *Focal* funkcije, iako na nekim elementima kao što je stup rasvjete s lijeve strane ceste (žuta boja na maski) imaju bolju detekciju. Jaccard funkcija postiže dobre rezultate ako se ne koristi s FPN mrežom u kojem slučaj rezultira s najlošijim modelom. Iznimno je dobra u kombinaciji s U-net arhitekturom jer postiže rezultate tek nešto lošije od *Focal* funkcije. Međutim, za razliku od nje ima bolje vrijednosti odziva. Na slici je vidljivo da Jaccard funkcija najlošije prepoznaje male i udaljene objekte kao što su znakovi, stupovi i linije prometnih traka.

| Arhitektura mreža | Funkcija gubitka | Mjeratest     |               |               |               |
|-------------------|------------------|---------------|---------------|---------------|---------------|
|                   |                  | F mjera       | IOU           | Preciznost    | Odziv         |
| U-net             | Jaccard          | 0.8836        | 0.7443        | 0.8352        | <b>0.9385</b> |
|                   | Dice             | 0.8805        | 0.7889        | 0.8387        | 0.9272        |
|                   | Tversky          | 0.849         | 0.74          | 0.7781        | 0.935         |
|                   | Focal            | 0.9078        | 0.8351        | <b>0.9552</b> | 0.8653        |
|                   | BCEWithLogits    | <b>0.9179</b> | <b>0.8524</b> | 0.9386        | 0.8983        |
| DeepLabV3Plus     | Jaccard          | 0.8205        | 0.6974        | 0.7192        | 0.9562        |
|                   | Dice             | 0.7558        | 0.6086        | 0.6224        | <b>0.9637</b> |
|                   | Tversky          | 0.762         | 0.6172        | 0.6319        | 0.9619        |
|                   | Focal            | 0.8914        | 0.8076        | <b>0.9591</b> | 0.8334        |
|                   | BCEWithLogits    | <b>0.908</b>  | <b>0.835</b>  | 0.9371        | 0.8809        |
| FPN               | Jaccard          | 0.5535        | 0.3833        | 0.3907        | 0.9504        |
|                   | Dice             | 0.6142        | 0.4437        | 0.4479        | <b>0.9784</b> |
|                   | Tversky          | 0.6021        | 0.4313        | 0.4352        | <b>0.9784</b> |
|                   | Focal            | 0.8903        | 0.8056        | <b>0.9617</b> | 0.8296        |
|                   | BCEWithLogits    | <b>0.9081</b> | <b>0.8351</b> | 0.9389        | 0.8795        |

Tablica 2. Rezultati usporedbe funkcija gubitka

Analizom rezultata iz Tablice 2 jasno se vidi da je vrlo bitno voditi brigu koju funkciju gubitka koristiti s kojom arhitekturom jer rezultati mogu biti značajno različiti. U-net

mreža je ostvarila najbolje rezultate. Sve funkcije u kombinaciji s njom postižu najbolje rezultate F i IoU mjere. Slike koje prikazuju kombinaciju U-net i *Cross entropy* ili *Focal* funkcije su iznimno dobre te se na njima vide detalji kao što su grane i lišće na drvećima, što nije prisutno na slikama ostalih modela. S druge strane FPN nije dobro kombinirati s Jaccard, Dice i Tversky funkcijama jer generirani modeli imaju značajno lošije performanse.

## 7.5. Zaključak

Cilj ovog istraživanja je bio usporediti najpopularnije funkcije gubitka za semantičku segmentaciju na skupu slika iz prometa. U tu svrhu smo generirali slike i njihove anotacije pomoću CARLA simulatora. U eksperiment smo uključili tri različite arhitekture mreža koje su najzastupljenije u dostupnoj literaturi. Nakon provedenog istraživanja možemo zaključiti da je *Binary Cross entropy* funkcija postigla najbolje rezultate bez obzira na odabranu arhitekturu. Dovoljno je robusna da bi ju svakako trebalo razmotriti u inicijalnom koraku stvaranja modela semantičke segmentacije. Najbolja se pokazala njezina kombinacija s U-net arhitekturom koja je ostvarila vrlo uvjerljive rezultate i impresivne slike maski. Osim *Binary Cross entropy* funkcije i *Focal* funkcija je dobar odabir jer otvara prostor da pomoću validacije  $\gamma$  parametra dođemo do još boljih rezultata. Ukoliko je odziv bitna mjera tada je preporučena kombinacija U-net mreže i Jaccard funkcije.

## 8. Usporedba tehnika za augmentaciju podataka za semantičku segmentaciju objekata u prometu

Autonomna vožnja sve je važnija i prisutnija u životima pojedinaca te je zbog toga sve veća važnost treniranja modela dubokog učenja koji omogućuju precizno semantičko segmentiranje ceste, vozila, prometnih znakova i ostalih objekata u prometu. Treniranje dubokog učenja, osim adekvatnog modela, zahtijeva i uporabu dovoljno velikog skupa podataka koji će obuhvaćati što je više moguće scenarija koji se mogu dogoditi u svakodnevnim situacijama u prometu. Ovim istraživanjem se uspoređuje više različitih tehnika augmentacije podataka kako bi se proširio skup podataka. Istraživanje je provedeno na podacima koji su prikupljeni iz CARLA simulatora prometa u urbanom okruženju te obuhvaća sljedeće tehnike augmentacije podataka: nasumična rotacija, nasumična inverzija boja, nasumično vertikalno zaokretanje, nasumično horizontalno zaokretanje, nasumično brisanje te nasumično pretvaranje slike u sive nijanse. Nabrojane tehnike upotrebljavaju se za povećavanje skupa podataka na kojem se treniraju tri različite arhitekture mreža, a one su: U-net, DeepLabV3Plus te FPN. Istraživanje nastoji donijeti preporuke i prikazati ostvarene rezultate korištenja različitih augmentacijskih tehnika.

### 8.1. Uvod

Proizvođači automobila danas se sve više okreću autonomnoj vožnji te korištenju umjetne inteligencije u svakodnevnom prijevozu ljudi. Vozilo koje koristi umjetnu inteligenciju, odnosno autonomni sustav za upravljanje vozilom u prometu, mora znati ne samo kako upravljati vozilom, već treba moći i percipirati buduće događaje u prometu. Mora očekivati i biti spremno na nepredvidive događaje koji uključuju druge sudionike u prometu te iz tih razloga vozilo, odnosno sustav, mora biti spreman u vrlo kratkom roku donijeti optimalnu odluku za promjenu smjera, brzine ili neku drugu akciju vezanu uz kontrolu vozila. Rješavanje takvog problema moguće je uporabom modela dubokog učenja koji svojim treniranjem *uči* raspoznavati različite objekte, sudionike te moguće događaje u prometu, pod uvjetom da je taj model treniran na raznolikom skupu podataka. Skup podataka mora obuhvaćati jako puno objekata i različitih scenarija, zbog čega takav skup podataka zahtijeva veliku količinu uloženog vremena, truda i

financijskih sredstava. Iz tog razloga koriste se simulatori gradske vožnje poput CARLA simulatora. On nudi kontrolu nad različitim varijabilnim stavkama simulatora kao naprimjer: vozila, pješaci, građevine, znakovi u prometu, vremenski uvjeti, doba dana, brzina i razne druge (Yuan i Xu, 2022). Prikupljanje skupa podataka zahtijeva i njihovu anotaciju te pogotovo taj proces dosta utječe na dugotrajnost i skupoću (Wang et al., 2020). Ovo istraživanje je fokusirano na uporabu različitih tehnika za augmentaciju podataka kako bi se iz ograničenog skupa podataka stvorio raznovrsniji i robusniji skup.

## 8.2. Pregled literature

Do danas je objavljeno puno istraživanje koji se bave problemom augmentacije podataka, ali ne postoje istraživanja koji su usmjereni isključivo na semantičku segmentaciju u prometu. U nastavku se prikazuje analiza postojeće literature, provedena u svrhu upoznavanja s trenutnim stanjem predmetnog područja. A Zhao et al. su prikazali korištenje metode za augmentaciju podataka u svrhu stvaranja novih, raznolikih i realističnih medicinskih slika (Zhao et al., 2019). Spomenuta metoda trenirana je kako bi prepoznala različite transformacije na slici. G Ghiasi et al. su prikazali uporabu jednostavne *Copy-Paste* metode koja je nasumično zaljepljivala objekte na sliku neovisno o vizualnom kontekstu. Ostvarili su povećanje od 0.6 prosječne preciznosti na maski za *instance* segmentaciju te 1.5 prosječne preciznosti za *box* u odnosu na tadašnji *state of the art* (Ghiasi et al., 2019). V Olsson et al. su predložili novu augmentacijsku tehniku *ClassMix* koja miješanjem neoznačenih uzoraka generira augmentacije te su prikazali da je ostvarila *state of the art* rezultate (Olsson et al., 2021). P Chen et al. su također predložili novu tehniku *GridMask* koja briše regije na slici te tako stvara augmentiranu sliku. Oni prikazuju rezultate ove tehnike koja je ostvarila bolje performanse od ostalih tehnika koje izostavljaju informacije (npr. *AutoAugment*) (Chen et al., 2020). S Liu et al. su prikazali kako su uz korištenje *Generative Adversial Networks* (GAN) generirali realistične slike, povećali točnost onih klasa s niskom točnošću te povećali sveukupnu točnost segmentacije s 1.3% na 2.1% (Liu et al., 2019). S Yang et al. su istražili različite tehnike augmentacije podataka i dali preporuke za semantičku i instance segmentaciju te detekciju objekata (Yang et al., 2022). J Nalepa et al. su istražili različite augmentacijske tehnike za

segmentaciju tumora na mozgu te su naveli kako se kombiniranjem različitih tehnika iz različitih algoritamskih grupa mogu ostvariti bolji rezultati, no također su naveli kako je potrebno dodatno istražiti posljedice uključivanja anatomske netočnosti tumora mozga u trening skup podataka (Nalepa et al., 2019). L Caselles et al. su predložili metodu baziranu na realističnim transformacijama koja kombinira pouzdani registracijski algoritam te strategiju miješanja (Caselles et al., 2022). A Anaya-Asaza i L Mera-Jiménez su usporedili različite augmentacijske tehnike (rotacija, zaokretanje, distorzija, ...) na slikama raka te su koristili *transfer-learning* na ImageNet skupu podataka i ostvarili F1 rezultat od 92.34% (Anaya i Mera-Jiménez, 2022). N Araslanov i S Roth su predložili uporabu standardnih augmentacijskih tehnika kao što su zaokretanje, fotometrijska buka i skaliranje za zadatak semantičke segmentacije. Ostvarili su značajna poboljšanja *state of the art* točnosti nakon uporabe tih tehnika augmentacije podataka (Araslanov i Roth, 2021). S Yukun et al. navode kako su konvencionalne tehnike koje se koriste za augmentaciju podataka geometrijske transformacije, nasumično obrezivanje te izmjenjivanje boja (Su et al., 2021). Casado-García et al. su naveli tri vrste augmentacijskih tehnika za detekciju objekata, a to su: geometrijske transformacije (translacije, rotacije i skaliranje), transformacije boja i filteri za koje kao primjer navode Gaussian filtere (Casado-García et al., 2019). S Illarionova et al. su opisali značenje augmentacije kao primjenu transformacija koje mogu biti: zaokretanje, rotacija, skaliranje, promjena svjetline i kontrasta (Illarionova et al., 2021). S Illarionova et al. su u svome istraživanju primijenili kombinaciju augmentacijskih tehnika rotacije (90 stupnjeva), vertikalno i horizontalno zaokretanje te približavanje i udaljavanje do 20% (Illarionova et al., 2020). A Körez et al. su primijenili augmentacijske tehnike zamagljenja, vertikalnog i horizontalnog zaokretanja, *gamma* pretvorbe i nasumične svjetline slike (Körez et al., 2020). X Yu et al. su u svome istraživanju naveli korištenje tri augmentacijske tehnike: zaokretanje, translacija i rotacija (Yu et al., 2017). Z Zong et al. su koristili biblioteku za augmentaciju koja uključuje kombinaciju različitih tehnika od kojih su neke: nasumično izrezivanje, rotacija malog kuta, zamagljivanje i zrcaljenje (Zong et al., 2019). W Li et al. su iskoristili augmentacijske tehnike skaliranja, izrezivanja i rotacije (Li et al., 2019). A Tavera et al. su koristili kombinaciju nasumično izabranih geometrijskih tehnika (horizontalno i vertikalno zaokretanje te nasumična rotacija) i nasumične fotometrijske augmentacije (transformacije boja) (Tavera et al., 2022). P Chen et al. su koristili tehnike nasumičnog brisanja, *Cutout* i *hide-and-see*k (Chen et al., 2020). Ovim

istraživanjem donosimo usporedbu različitih standardnih augmentacijskih tehnika za semantičku segmentaciju na tri najpopularnije arhitekture modela.

### 8.3. Augmentacija podataka i tehnike

Nastavak sadržava kratki opis svake augmentacijske tehnike koja je obuhvaćena ovim istraživanjem. Opisane tehnike su izabrane zbog njihove uporabe u prethodno navedenoj literaturi. Svaka tehnika sadržava argumente čijim se korištenjem upravlja samom tehnikom. Tehnike i argumenti su dio Pytorch paketa Torchvision. U Tablici 3 su nabrojane korištene augmentacijske tehnike te njihove skraćene oznake kako bi podaci u tablici bili pregledniji.

| Naziv augmentacijske tehnike               | Skraćena oznaka |
|--|-----------------|
| Nasumična rotacija                         | NR              |
| Nasumična inverzija boja                   | NIB             |
| Nasumično vertikalno zaokretanje           | NVZ             |
| Nasumično horizontalno zaokretanje         | NHZ             |
| Nasumično brisanje                         | NB              |
| Nasumično pretvaranje slike u sive nijanse | NPSUSN          |

Tablica 3. Popis svih augmentacijskih tehnika i njihovih skraćenih oznaka

#### 8.3.1. Nasumična rotacija

Nasumičnom rotacijom se rotira dana slika za neki stupanj, a stupanj rotacije daje se kao parametar minimalne i maksimalne vrijednosti između kojih se nasumično izabire stupanj rotacije. Parametri koje ova tehnika sadržava su:

- **stupnjevi** (engl. *degrees*) - predstavlja raspon stupnjeva između kojih se nasumično izabire stupanj rotacije
- **interpolacija** (engl. *interpolation*) - vrsta interpolacije, odnosno kreiranja novih podataka na temelju postojećih, zadana vrsta je *nearest*



- **proširi** (engl. *expand*) - *boolean* vrijednost koja, ukoliko je istinita, postavlja da je slika dovoljno velika da prikaže cijelu rotiranu sliku, a ukoliko nije, prikazuje se izlazna slika iste veličine kao ulazna
- **centar** (engl. *center*) - postavljanje x i y točke na kojima će se rotirati slika, zadana vrijednost je centar
- **ispuna** (engl. *fill*) - boja kojom će se obojati pikseli izvan slike, zadana je nula

### 8.3.2. Nasumična inverzija boja

Nasumična inverzija boja nasumično izabire koje će se boje na danoj slici izmijeniti, a jedini parametar koji ova tehnika sadržava je:

- **vjerojatnost** - vjerojatnost da će slika imati invertirane boje, zadana vrijednost je 0.5 (50%)

### 8.3.3. Nasumično vertikalno zaokretanje

Nasumično vertikalno zaokretanje vertikalno zaokreće sliku te sadržava samo jedan parametar. Parametar koji ova tehnika sadržava je:

- **vjerojatnost** - označava vjerojatnost da će se slika nasumično vertikalno zaokrenuti, zadana je vrijednost 0.5 (50%)

### 8.3.4. Nasumično horizontalno zaokretanje

Nasumično horizontalno zaokretanje isto je kao vertikalno zaokretanje, samo što se zaokretanje odvija horizontalno. Jedini parametar koji sadržava je isti kao i kod vertikalnog zaokretanja. Parametar koji ova tehnika sadržava je:

- **vjerojatnost** - označava vjerojatnost da će se slika nasumično horizontalno zaokrenuti, zadana je vrijednost 0.5 (50%)

### 8.3.5. Nasumično brisanje dijelova slike

Nasumično brisanje dijelova slike nasumično izabire pravokutnik na danoj slici te briše piksele unutar tog pravokutnika. Parametri kojima se može upravljati ovom tehnikom su:

- **vjerojatnost** - vjerojatnost da će se ova operacija izvršiti na slici
- **skala** (engl. *scale*) - proporcija pravokutnika na slici kojem će se izbrisati pikseli
- **omjer** (engl. *ratio*) - *aspect ratio* izbrisanog dijela slike
- **vrijednost** (engl. *value*) - vrijednost koja određuje što će se brisati, zadano je nula što znači da će se sve izbrisati, ukoliko je *tuple* veličine 3 onda se brišu R, G i B kanali, a *string* s vrijednošću *random* nasumično briše piksele
- **na mjestu** (engl. *inplace*) - *boolean* vrijednost koja postavlja da je ta augmentacija *inplace* ovisno o istinitosti dodane vrijednosti

### 8.3.6. Nasumično pretvaranje slike u sive nijanse

Nasumično pretvaranje boja u sive nijanse je slično nasumičnoj inverziji boja, samo što su boje koje se ovom tehnikom dobivaju iz palete sivih boja. Jedini parametar koji ova tehnika sadržava također je:

- **vjerojatnost** - vjerojatnost da će slika biti pretvorena u sliku s bojama iz palete sivih boja

## 8.4. Eksperiment i rezultati analize

Radi prikazivanja utjecaja različitih augmentacijskih tehnika na treniranje modela semantičke segmentacije u prometu, prikazan je utjecaj svake od navedenih augmentacijskih tehnika na tri arhitekture koje su najčešće korištene u literaturi. Pregled i praćenje uspješnosti modela je prikazano koristeći sljedeće evaluacijske mjere: F mjera (engl. *F score*), IoU (engl. *Intersection over union*), preciznost (engl.

*Precision*) i odziv (engl. *Recall*). Ostvareni rezultati su prikazani u nastavku. Analiza rezultata može se podijeliti u nekoliko kategorija iz razloga što je to česti oblik koji se koristi u literaturi, a te kategorije su: trenirani modeli bez upotrebe augmentacijskih tehnika, zasebno korištene augmentacijske tehnike, kombinacija dvije augmentacijske tehnike, kombinacija tri augmentacijske tehnike, kombinacija pet augmentacijskih tehnika i kombinacija svih spomenutih augmentacijskih tehnika. Augmentacijskim tehnikama nije bio prosljeđen niti jedan argument prilikom treninga, već su korištene zadane vrijednosti. Time smo osigurali minimalno utjecanje na trening kako bi objektivno prikazali rezultate i zaključak istraživanja. Također su vrijednosti zadanih argumenata često preporučene vrijednosti autora ili stručnjaka.

#### **8.4.1. Skup podataka iz simulatora CARLA**

Skup podataka je stvoren koristeći CARLA simulator autonomne vožnje. Radi dobivanja slika i istinitih maski korištena su dva zasebna senzora. Jedan od dva senzora korišten je za prikupljanje maski. Taj senzor sadržava prazne vrijednosti u zelenom i plavom kanalu, a u crvenom kanalu sadržava broj klase (od 0 do 22 jer sadržava sveukupno 23 klase). Korištenjem senzora koje nudi CARLA generiraju se dvije slike kao jedan par, jedna je RGB slika, a druga je maska. Slika je širine 3 x širina x visina, dok je maska dimenzija 1 x širina x visina. U procesu pripreme ciljnih podataka maska se dijeli na način da se dobije C (engl. *Classes*) zasebnih kanala od kojih svaki sadržava samo one piksele koji pripadaju toj klasi. Predikcija generira sliku s C zasebnih kanala te se sažima korištenjem *argmax* funkcije. Skup podataka sadržava 11000 slika od kojih je 70% trening, 20% validacijskih i 10% test slika.

#### **8.4.2. Modeli**

Ovo istraživanje je nastavak na prijašnje istraživanje *Usporedba funkcija gubitka za semantičku segmentaciju objekata u prometu* gdje su detaljnom analizom dobiveni rezultati da je za svaku od tri različite arhitekture mreža najbolja funkcija gubitka zapravo *Binary Cross entropy*. Korištene arhitekture su: FPN (engl. *Feature Pyramid*

*Network*), DeepLabV3Plus te U-net. O tim arhitekturama mreža se može detaljnije pročitati u navedenom istraživanju.

### 8.4.3. Evaluacijske mjere

Evaluacijske mjere korištene za mjerenje uspješnosti modela u ovom istraživanju su F mjera, preciznost i odziv koje imaju polazište u matrici zabune (engl. *Confusion matrix*) te IoU. Matrica zabune sadržava 4 moguća rezultata predikcije: istinito pozitivna predikcija, istinito negativna predikcija, lažno pozitivna predikcija te lažno negativna predikcija. F mjera je harmonijska sredina presjeka preciznosti i odziva. Preciznost je udio ispravno klasificiranih slučajeva u odnosu na skup svih pozitivno klasificiranih, odziv je udio ispravno klasificiranih slučajeva u odnosu na skup svih pozitivnih slučajeva, a IoU mjeri preklapanje između predviđene i istinite maske.

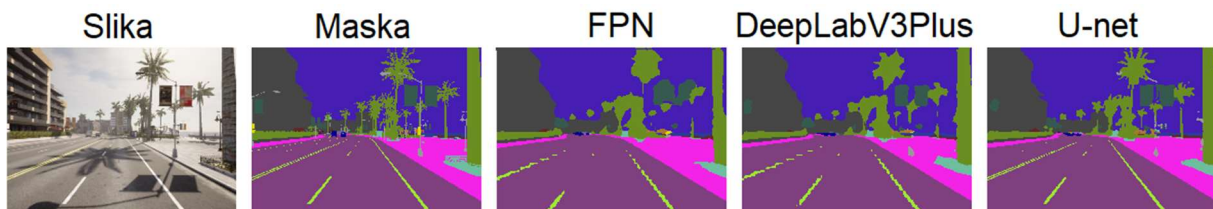
### 8.4.4. Rezultati

Trening modela je trajao do trenutka kada je rezultat funkcije gubitka prestao značajno padati, a to je omogućeno korištenjem tehnike ranog zaustavljanja (engl. *early stopping*) koja je zaustavila trening ukoliko se rezultat funkcije gubitka nije u zadnjih 5 epoha spustio za 0.001. Vodeći se spomenutim evaluacijskim mjerama, u Tablici 5 prikazani su konačni rezultati svake arhitekture po pojedinoj augmentacijskoj tehnici. Performanse svake pojedine augmentacijske tehnike su analizirane te je analizirana i korištena arhitektura. U Tablici 4 su prikazani rezultati treniranja navedenih arhitektura mreža sa spomenutom funkcijom gubitka bez upotrebe augmentacijskih tehnika. Najbolje rezultate, vodeći se evaluacijskim mjerama F mjera, IoU te odziv, ostvarila je arhitektura U-net, dok je po evaluacijskoj mjeri preciznost najbolji rezultat ostvarila arhitektura FPN. Postignuti rezultati su prikazani na Slici 5 gdje je vidljivo da je poprilično dobru predikciju ostvarila arhitektura U-net, dok su ostale arhitekture ostvarile predikciju s manjom uspješnošću u usporedbi s U-net arhitekturom. Graf rezultata funkcije gubitka na Slici 7 prikazuje kako je najveći pad rezultata funkcije

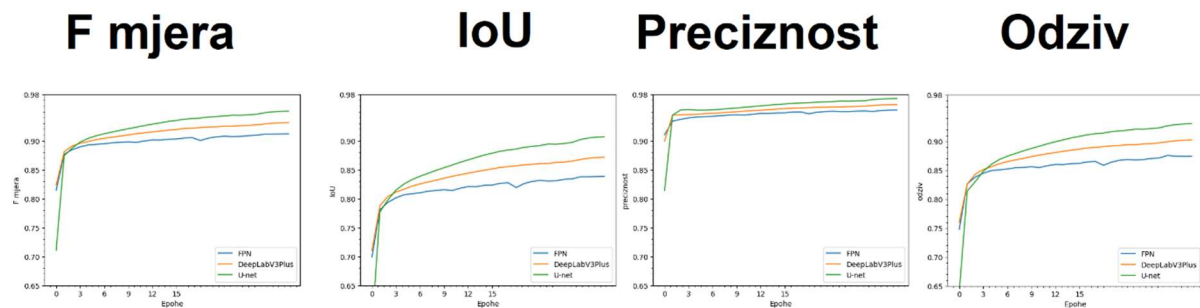
gubitka ostvario model s arhitekturom DeepLabV3Plus, dok je najmanji rezultat funkcije gubitka ostvario model s arhitekturom U-net.

| Arhitektura mreža | Funkcija gubitka     | Mjera <sub>test</sub> |               |               |               |
|-------------------|----------------------|-----------------------|---------------|---------------|---------------|
|                   |                      | F mjera               | IoU           | Preciznost    | Odziv         |
| FPN               | Binary Cross entropy | 0.9064                | 0.8309        | <b>0.9477</b> | 0.8689        |
| DeepLabV3Plus     | Binary Cross entropy | 0.9079                | 0.8335        | 0.9373        | 0.8804        |
| U-net             | Binary Cross entropy | <b>0.9188</b>         | <b>0.8524</b> | 0.94          | <b>0.8986</b> |

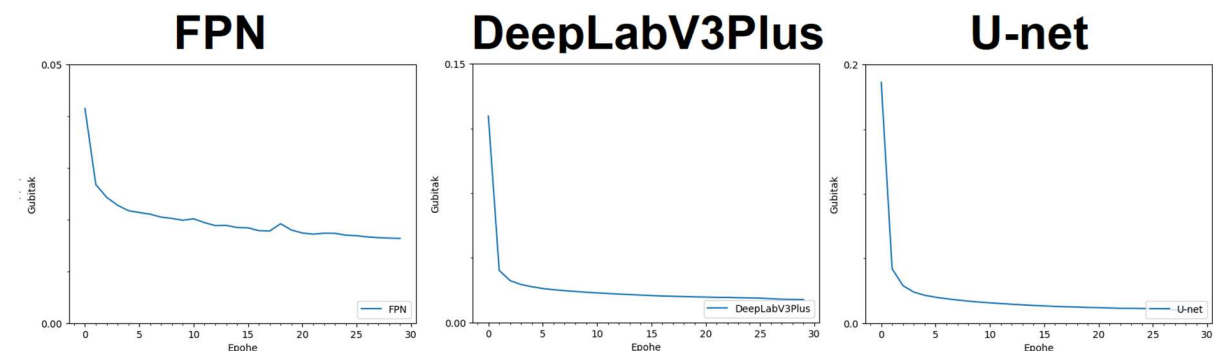
Tablica 4. Rezultati treniranja bez uporabe augmentacijske tehnike



Slika 5. Prikaz predikcija bez korištenja augmentacijskih tehnika



Slika 6. Prikaz rezultata funkcija gubitaka po arhitekturi mreže



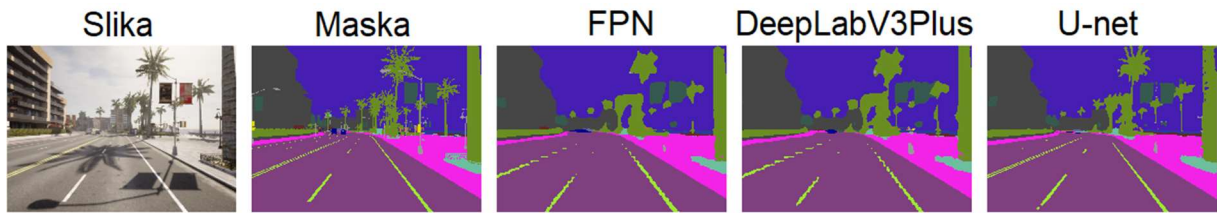
Slika 7. Prikaz treninga po evaluacijskim mjerama

#### 8.4.4.1. Zasebno korištene augmentacijske tehnike

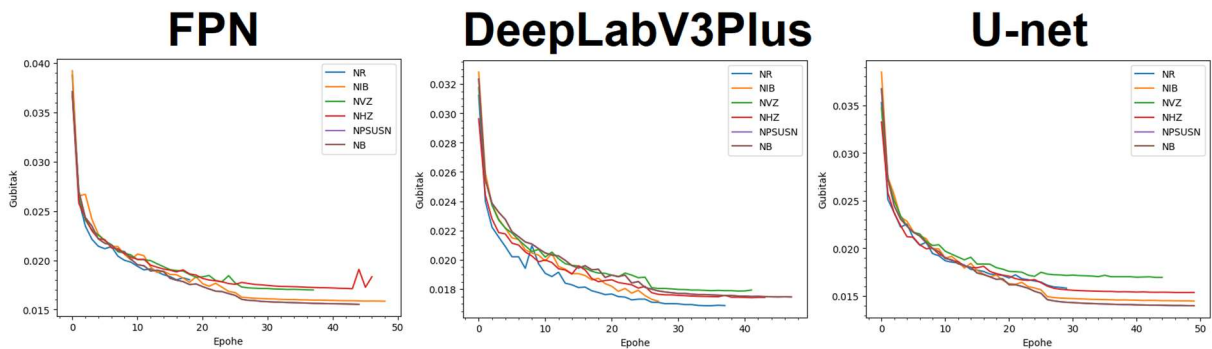
Tablica 5 prikazuje rezultate ostvarene treniranjem modela s jednom augmentacijskom tehnikom. Analizirajući tablicu može se iščitati kako je augmentacijska tehnika nasumične rotacije ostvarila najbolje rezultate po F mjeri s arhitekturama FPN i U-net, dok je augmentacijska tehnika nasumične inverzije boja ostvarila najbolje rezultate s arhitekturom DeepLabV3Plus, također gledajući po F mjeri. Predikcije su prikazane na Slici 8, a pad rezultata funkcije gubitka i porast rezultata po svakoj evaluacijskoj mjeri su prikazani na Slici 9 i Slici 10.

| Arhitektura mreža | Augmentacijska tehnika | Mjera <sub>test</sub> |               |               |               |
|-------------------|------------------------|-----------------------|---------------|---------------|---------------|
|                   |                        | F mjera               | IoU           | Preciznost    | Odziv         |
| FPN               | NR                     | <b>0.9146</b>         | <b>0.8439</b> | <b>0.9505</b> | 0.8815        |
|                   | NIB                    | 0.9073                | 0.8323        | 0.9471        | 0.871         |
|                   | NVZ                    | 0.9056                | 0.8295        | 0.9488        | 0.8666        |
|                   | NHZ                    | 0.9056                | 0.8294        | 0.949         | <b>0.9056</b> |
|                   | NB                     | 0.9089                | 0.8351        | 0.9489        | 0.8726        |
|                   | NPSUSN                 | 0.9061                | 0.8305        | 0.9446        | 0.871         |
| DeepLabV3Plus     | NR                     | 0.9037                | 0.8267        | 0.9432        | 0.8679        |
|                   | NIB                    | <b>0.9082</b>         | 0.832         | <b>0.9512</b> | 0.869         |
|                   | NVZ                    | 0.9041                | 0.8273        | 0.948         | 0.8653        |
|                   | NHZ                    | 0.9059                | 0.8298        | 0.9476        | 0.8683        |
|                   | NB                     | 0.9081                | <b>0.8338</b> | 0.9492        | <b>0.871</b>  |
|                   | NPSUSN                 | 0.9041                | 0.8272        | 0.9454        | 0.8669        |
| U-net             | NR                     | <b>0.9251</b>         | <b>0.861</b>  | <b>0.9567</b> | <b>0.8956</b> |
|                   | NIB                    | 0.9178                | 0.8502        | 0.9507        | 0.8874        |
|                   | NVZ                    | 0.9172                | 0.8493        | 0.952         | 0.8853        |
|                   | NHZ                    | 0.918                 | 0.8508        | 0.9525        | 0.8864        |
|                   | NB                     | 0.9208                | 0.8556        | 0.9515        | 0.8924        |
|                   | NPSUSN                 | 0.917                 | 0.8491        | 0.9477        | 0.8885        |

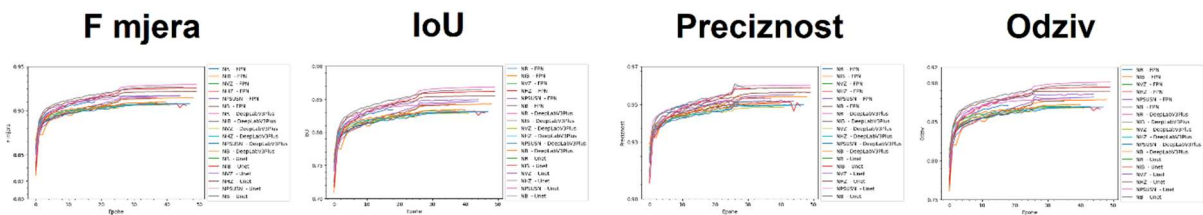
Tablica 5. Rezultati korištenja jedne augmentacijske tehnike



Slika 8. Prikaz predikcija koristeći jednu augmentacijsku tehniku



Slika 9. Prikaz rezultata funkcija gubitaka po arhitekturi mreže



Slika 10. Prikaz treninga po evaluacijskim mjerama

Analizom rezultata F mjere iz Tablice 5 vidljivo je da korištena arhitektura utječe na rezultate pa je tako model s arhitekturom U-net ostvario najbolje rezultate po svim tehnikama u usporedbi s ostalim arhitekturama. Približne rezultate su ostvarile arhitekture FPN te DeepLabV3Plus.

FPN je po svim mjerama, izuzev odziva, najbolje rezultate ostvario s tehnikom nasumične rotacije. DeepLabV3Plus je za F mjeru i preciznost najbolje rezultate ostvario s tehnikom nasumične inverzije boja, a za IoU i odziv s tehnikom nasumičnog brisanja. U-net je najbolje rezultate neovisno o mjeri ostvario s tehnikom nasumične rotacije. Uspoređujući ove rezultate s rezultatima ostvarenima s modelima bez upotrebe augmentacijskih tehnika vidljivo je da je svaki model ostvario bolje rezultate. Najbolji rezultat po F mjeri FPN arhitekture je porastao s 0.9065 na 0.9146, za DeepLabV3Plus arhitekturu s 0.9079 na 0.9082, a za U-net s 0.9188 na 0.9251. Slika

8 sadržava slike predikcija po arhitekturi s kombinacijom koja je ostvarila najbolje rezultate po F mjeri. Modeli koji su ostvarili bolje rezultate su također ostvarili i bolje predikcije koje su prikazane na Slici 8. Na toj slici je vidljivo da je model s arhitekturom FPN u kombinaciji s tehnikom nasumične rotacije uspio prepoznati dno stupa, ali nije uspio prepoznati prometni znak. Također je detaljnije označio stabla, ogradu i bolje prepoznao ograđeni dio oko stabla. DeepLabV3Plus u kombinaciji s tehnikom nasumične inverzije boja je također detaljnije označio stabla, stupove i kolnik. U-net, u kombinaciji s tehnikom nasumične rotacije, je bolje prepoznao oznake na prometnoj traci i ogradu te bolje označio stabla u daljini. Na Slici 9 je prikazan pad rezultata funkcije gubitka dok je na Slici 10 vidljivo kretanje rezultata po evaluacijskim mjerama.

#### 8.4.4.2. Kombinacija dvije augmentacijske tehnike

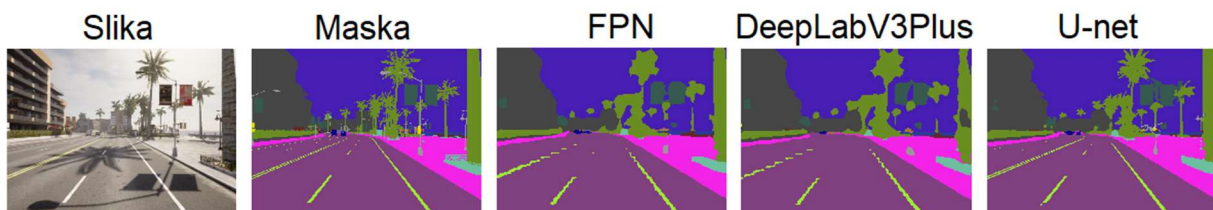
S obzirom na to da su W Li et al. koristili kombinaciju skaliranja, izrezivanja i rotacije (Li et al., 2019), u ovom istraživanju se koristi kombinacija tehnika nasumične rotacije i brisanja umjesto izrezivanja. Z Zong et al. su koristili kombinaciju raznih tehnika (Zong et al., 2019) te se iz tog razloga koristi i kombinacija tehnika nasumične rotacije i horizontalnog zaokretanja. A Tavera et al. između ostalog su koristili i tehnike zaokretanja te nasumične fotometrijske tehnike (Tavera et al., 2022). Zbog toga su u ovo istraživanje uključene kombinacije tehnika nasumičnog vertikalnog zaokretanja i nasumičnog pretvaranja slike u sive nijanse te kombinacije nasumičnog vertikalnog zaokretanja i nasumične inverzije boja. Augmentacijska tehnika nasumičnog vertikalnog zaokretanja je korištena zbog ostvarenih boljih rezultata kod treniranja s jednom augmentacijskom tehnikom, u odnosu na tehniku nasumičnog horizontalnog zaokretanja. X Yu et al. su koristili zaokretanje, translaciju te rotaciju (Yu et al., 2017), iz tog razloga se u ovom istraživanju koristi kombinacija tehnika nasumičnog vertikalnog zaokretanja i nasumične rotacije.

| Arhitektura mreža | Augmentacijska tehnika | Mjera <sub>test</sub> |               |               |               |
|-------------------|------------------------|-----------------------|---------------|---------------|---------------|
|                   |                        | F mjera               | IoU           | Preciznost    | Odziv         |
| FPN               | NR i NB                | 0.8943                | 0.8107        | 0.9434        | 0.8506        |
|                   | NR i NHZ               | <b>0.9162</b>         | <b>0.8467</b> | <b>0.9546</b> | <b>0.8811</b> |
|                   | NVZ i NPSUSN           | 0.9044                | 0.8275        | 0.9456        | 0.867         |

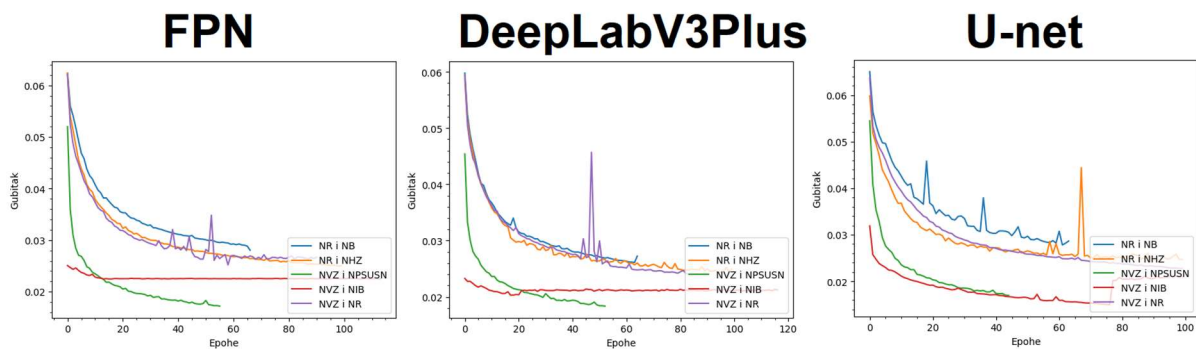


|               |              |               |               |               |               |
|---------------|--------------|---------------|---------------|---------------|---------------|
|               | NVZ i NIB    | 0.9054        | 0.8293        | 0.9438        | 0.8703        |
|               | NVZ i NR     | 0.9156        | 0.8457        | 0.9533        | 0.8811        |
| DeepLabV3Plus | NR i NB      | 0.8961        | 0.8139        | 0.9382        | 0.8584        |
|               | NR i NHZ     | <b>0.9157</b> | 0.846         | 0.9537        | 0.8813        |
|               | NVZ i NPSUSN | 0.9025        | 0.8243        | 0.9443        | 0.8647        |
|               | NVZ i NIB    | 0.904         | 0.8268        | 0.9395        | 0.8715        |
|               | NVZ i NR     | 0.9165        | <b>0.8472</b> | <b>0.9512</b> | <b>0.8848</b> |
| U-net         | NR i NB      | 0.9092        | 0.8359        | 0.9435        | 0.8776        |
|               | NR i NHZ     | <b>0.9273</b> | <b>0.866</b>  | <b>0.9573</b> | 0.8993        |
|               | NVZ i NPSUSN | 0.915         | 0.8458        | 0.9469        | 0.8855        |
|               | NVZ i NIB    | 0.9151        | 0.8458        | 0.949         | 0.884         |
|               | NVZ i NR     | 0.9266        | 0.8649        | 0.9528        | <b>0.902</b>  |

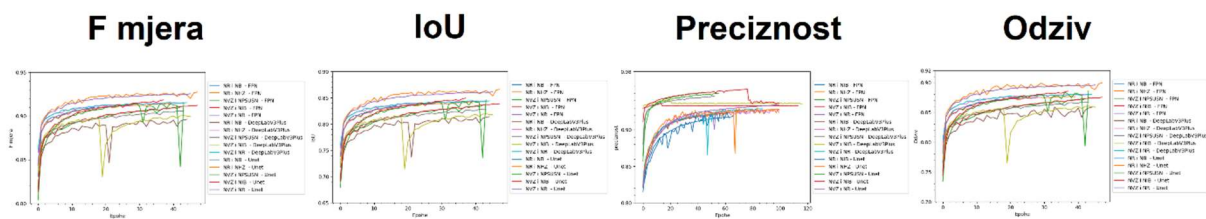
Tablica 6. Rezultati korištenja kombinacije dvije augmentacijske tehnike



Slika 11. Prikaz predikcija koristeći dvije augmentacijske tehnike



Slika 12. Prikaz rezultata funkcija gubitaka po arhitekturi mreže



Slika 13. Prikaz treninga po evaluacijskim mjerama

Tablica 6 prikazuje rezultate korištenja kombinacije dvije različite tehnike za augmentaciju podataka. Analizom rezultata vidljivo je kako su ostvareni rezultati, neovisno o korištenoj evaluacijskoj mjeri, postigli napredak u usporedbi s prijašnje ostvarenim rezultatima. Vidljivo je kako je korištenje kombinacije u kojoj je jedna od tehnika nasumična rotacija, a druga nasumično horizontalno zaokretanje omogućilo ostvarivanje najboljih rezultata za FPN arhitekturu neovisno o evaluacijskoj mjeri. Za U-net je ta kombinacija omogućila napredak po svakoj evaluacijskoj mjeri osim odziva. Odziv je pak ostvario malo bolje rezultate s kombinacijom nasumičnog vertikalnog zaokretanja i nasumične rotacije. Ponovno je najbolje rezultate ostvarila arhitektura U-net, zatim FPN te DeepLabV3Plus. Na Slici 11 je vidljivo da su modeli s arhitekturama FPN i DeepLabV3Plus ostvarili slabije predikcije u usporedbi s U-net arhitekturom. Uspoređujući rezultate s modelima koji su koristili samo jednu tehniku vidljivo je kako je FPN bolje prepoznao vozila u daljini i stabla te je uspio prepoznati udaljenu ogradu i detaljnije označiti male objekte na slici (naprimjer stup sa strane). DeepLabV3Plus je također uspio prepoznati udaljenu ogradu te je bolje naučio prepoznavati stabla (naprimjer stablo s lijeve strane), dok je U-net puno bolje uspio prepoznati vozila u daljini, male objekte i prometni znak s desne strane označen žutom bojom. Na Slici 12 je prikazan pad rezultata funkcija gubitka, dok je na Slici 13 vidljivo kretanje rezultata po evaluacijskim mjerama.

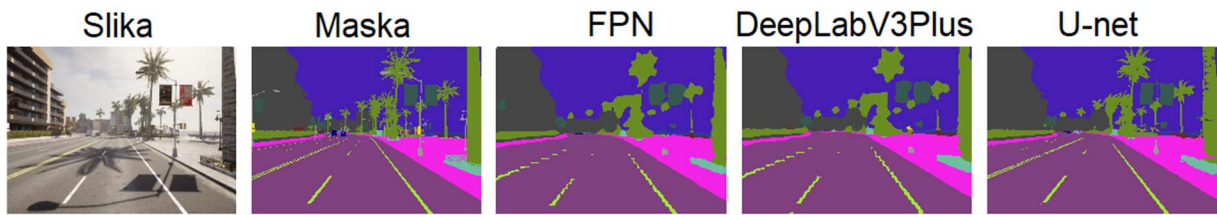
#### 8.4.4.3. Kombinacija tri augmentacijske tehnike

S Illarionova et al. su koristili rotaciju, vertikalno i horizontalno zaokretanje te približavanje i udaljavanje (Illarionova et al., 2020) pa se iz tog razloga kao prva kombinacija koristila kombinacija tehnika nasumične rotacije te nasumičnog horizontalnog i vertikalnog zaokretanja. S Illarionova et al. u drugom istraživanju su koristili kombinaciju raznih tehnika od kojih su neke zaokretanje, fotometrijske tehnike

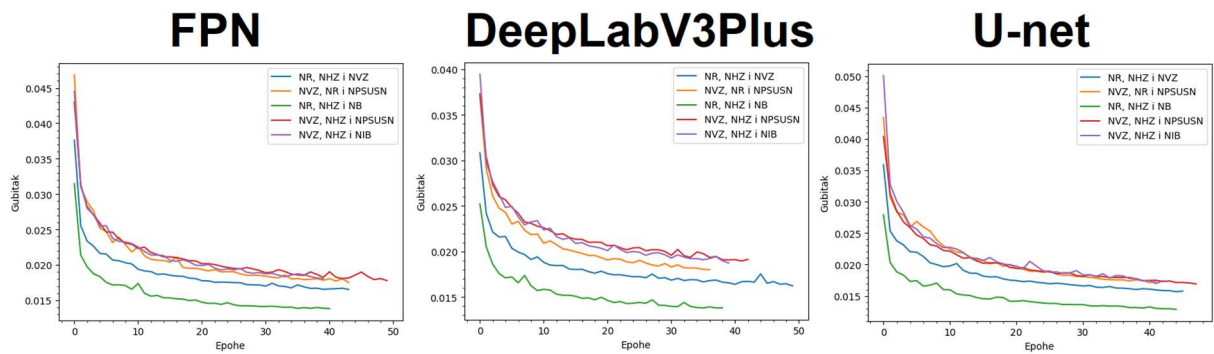
i rotacija (Illarionova et al., 2021) zbog čega se koristila kombinacija tehnika nasumičnog vertikalnog zaokretanja, rotacije i nasumičnog pretvaranja slika u sive nijanse. Z Zong et al. su koristili tehnike nasumične rotacije i horizontalnog zaokretanja te nasumičnog brisanja (Zong et al., 2019). Spomenuta kombinacija se koristila i u ovom istraživanju. A Körez et al. su koristili tehnike zaokretanja i transformacija boja (Körez et al., 2020). Zbog toga su u ovom istraživanju korištene sljedeće dvije kombinacije: kombinacija tehnika nasumičnog vertikalnog i horizontalnog zaokretanja te nasumičnog pretvaranja slike u sive nijanse i kombinacija nasumičnog vertikalnog i horizontalnog zaokretanja i inverzije boja.

| Arhitektura mreža | Augmentacijska tehnika | Mjera <sub>test</sub> |               |               |               |
|-------------------|------------------------|-----------------------|---------------|---------------|---------------|
|                   |                        | F mjera               | IoU           | Preciznost    | Odziv         |
| FPN               | NR, NHZ i NVZ          | <b>0.9163</b>         | <b>0.8469</b> | <b>0.9548</b> | <b>0.8812</b> |
|                   | NVZ, NR i NPSUSN       | 0.912                 | 0.8396        | 0.9517        | 0.8758        |
|                   | NR, NHZ i NB           | 0.895                 | 0.812         | 0.9428        | 0.8523        |
|                   | NVZ, NHZ i NPSUSN      | 0.9041                | 0.8269        | 0.9443        | 0.8676        |
|                   | NVZ, NHZ i NB          | 0.9054                | 0.8291        | 0.9443        | 0.8701        |
| DeepLabV3Plus     | NR, NHZ i NVZ          | <b>0.9174</b>         | <b>0.8487</b> | <b>0.9515</b> | <b>0.886</b>  |
|                   | NVZ, NR i NPSUSN       | 0.9116                | 0.839         | 0.9476        | 0.8787        |
|                   | NR, NHZ i NB           | 0.8932                | 0.8096        | 0.9421        | 0.8501        |
|                   | NVZ, NHZ i NPSUSN      | 0.9021                | 0.8238        | 0.942         | 0.866         |
|                   | NVZ, NHZ i NB          | 0.9045                | 0.8274        | 0.9408        | 0.8712        |
| U-net             | NR, NHZ i NVZ          | <b>0.9266</b>         | <b>0.8647</b> | <b>0.9544</b> | <b>0.9006</b> |
|                   | NVZ, NR i NPSUSN       | 0.9204                | 0.8541        | 0.9531        | 0.8901        |
|                   | NR, NHZ i NB           | 0.91                  | 0.8371        | 0.9436        | 0.879         |
|                   | NVZ, NHZ i NPSUSN      | 0.9129                | 0.8422        | 0.9467        | 0.8818        |
|                   | NVZ, NHZ i NB          | 0.9135                | 0.843         | 0.949         | 0.8809        |

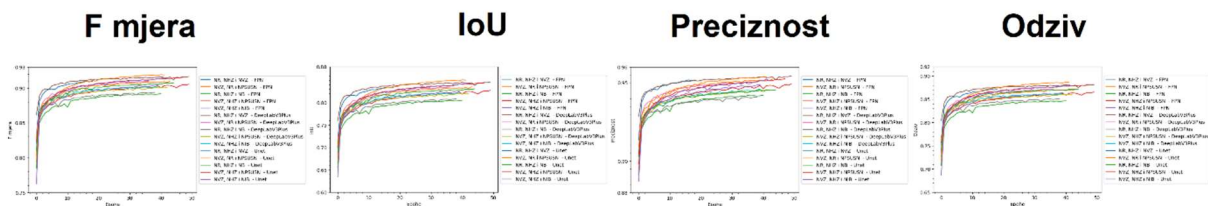
Tablica 7. Rezultati korištenja kombinacije tri augmentacijske tehnike



Slika 14. Prikaz predikcija koristeći tri augmentacijske tehnike



Slika 15. Prikaz rezultata funkcija gubitaka po arhitekturi mreže



Slika 16. Prikaz treniranja po evaluacijskim mjerama

U Tablici 7 je prikazano kako je korištenje kombinacije od tri različite tehnike za augmentaciju podataka rezultiralo ostvarivanjem približnih ili boljih performansi modela od kombinacije dviju tehnika. Najbolje rezultate za sve arhitekture mreža je ostvarila kombinacija nasumične rotacije te nasumičnog vertikalnog i horizontalnog zaokretanja. FPN arhitektura je ostvarila poboljšanje po F mjeri s 0.9162 na 0.9163, a DeepLabV3Plus s 0.9157 na 0.9174. Spomenuta kombinacija tehnika s arhitekturom U-net nije ostvarila bolje rezultate od prijašnje najbolje kombinacije dviju tehnika: tehnike nasumične rotacije i nasumičnog horizontalnog zaokretanja; rezultati su gotovo jednaki. Iako korištene kombinacije nisu ostvarile puno bolje rezultate, napredak se može vidjeti na predikcijama na Slici 14. Neovisno o arhitekturi, ovim kombinacijama poboljšalo se prepoznavanje udaljene ograde te je DeepLabV3Plus uspio prepoznati prometni znak s desne strane. Rezultati funkcije gubitka i rezultati treniranja po evaluacijskim mjerama su prikazani na Slici 15 i 16.

#### 8.4.4.4. Kombinacija pet augmentacijskih tehnika

Kako su A Tavera et al. koristili razne tehnike za augmentaciju podataka (Tavera et al., 2022) tako se i u ovom istraživanju koriste kombinacije po pet spomenutih tehnika, od kojih se za dvije kombinacije jedino razlikuju tehnika nasumične rotacije i brisanja. Kao preostale tri kombinacije koriste se ostale tehnike od kojih od dvije kombinacije jedna sadržava tehniku nasumičnog pretvaranja slike u sive nijanse, a druga tehniku nasumične inverzije boja. Preostala kombinacija sadržava sve tehnike izuzev tehnike nasumičnog vertikalnog zaokretanja jer je ostvarila lošije rezultate od tehnike nasumičnog horizontalnog zaokretanja.

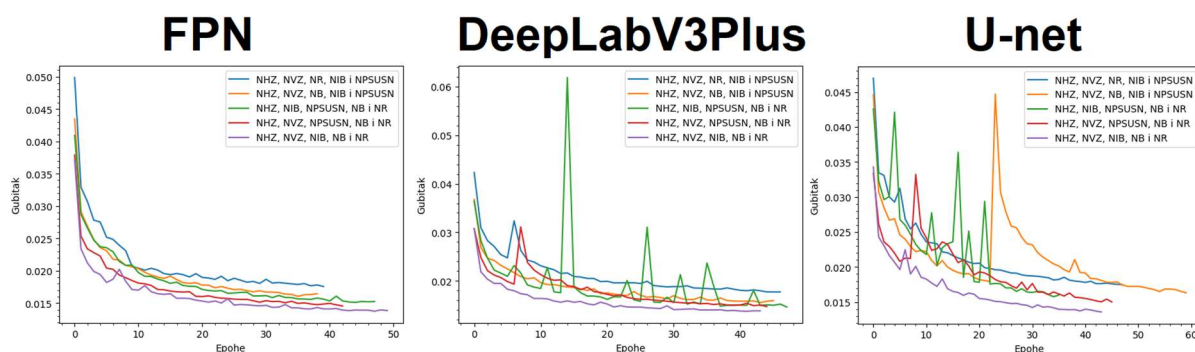
| Arhitektura mreža | Augmentacijska tehnika     | Mjera <sub>test</sub> |               |               |               |
|-------------------|----------------------------|-----------------------|---------------|---------------|---------------|
|                   |                            | F mjera               | IoU           | Preciznost    | Odziv         |
| FPN               | NHZ, NVZ, NR, NIB i NPSUSN | 0.8899                | 0.8037        | 0.9385        | 0.8467        |
|                   | NHZ, NVZ, NB, NIB i NPSUSN | <b>0.9019</b>         | <b>0.8216</b> | <b>0.9477</b> | <b>0.8604</b> |
|                   | NHZ, NIB, NPSUSN, NB i NR  | 0.8853                | 0.7962        | 0.9407        | 0.8369        |
|                   | NHZ, NVZ, NPSUSN, NB i NR  | 0.8898                | 0.8034        | 0.9386        | 0.8463        |
|                   | NHZ, NVZ, NIB, NB i NR     | 0.895                 | 0.8118        | 0.9443        | 0.8512        |
| DeepLabV3Plus     | NHZ, NVZ, NR, NIB i NPSUSN | 0.8905                | 0.8046        | 0.9354        | 0.8503        |
|                   | NHZ, NVZ, NB, NIB i NPSUSN | <b>0.9008</b>         | <b>0.8214</b> | <b>0.9408</b> | <b>0.8644</b> |
|                   | NHZ, NIB, NPSUSN, NB i NR  | 0.888                 | 0.8008        | 0.9364        | 0.8452        |
|                   | NHZ, NVZ, NPSUSN, NB i NR  | 0.8903                | 0.8043        | 0.9383        | 0.8474        |
|                   | NHZ, NVZ, NIB, NB i NR     | 0.8952                | 0.8125        | 0.9383        | 0.8568        |
| U-net             | NHZ, NVZ, NR, NIB i NPSUSN | 0.8992                | 0.8189        | 0.9414        | 0.861         |
|                   | NHZ, NVZ, NB, NIB i NPSUSN | 0.9026                | 0.8249        | <b>0.9436</b> | 0.8654        |

|  |                                 |              |               |        |               |
|--|---------------------------------|--------------|---------------|--------|---------------|
|  | NHZ, NIB,<br>NPSUSN, NB i<br>NR | 0.8919       | 0.8072        | 0.9365 | 0.8519        |
|  | NHZ, NVZ,<br>NPSUSN, NB i<br>NR | 0.8948       | 0.8122        | 0.9389 | 0.8554        |
|  | NHZ, NVZ, NIB,<br>NB i NR       | <b>0.907</b> | <b>0.8319</b> | 0.9389 | <b>0.8774</b> |

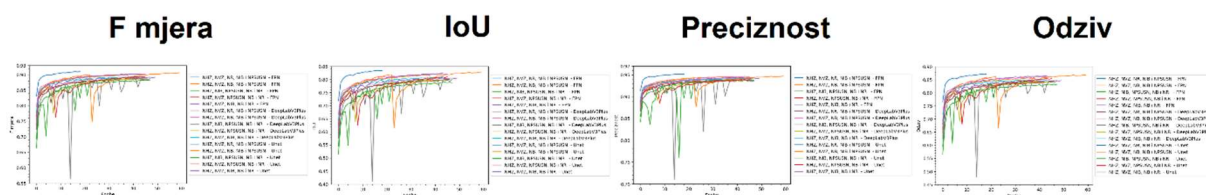
Tablica 8. Rezultati korištenja kombinacije pet augmentacijskih tehnika



Slika 17. Prikaz predikcija koristeći pet augmentacijskih tehnika



Slika 18. Prikaz rezultata funkcija gubitaka po arhitekturi mreže



Slika 19. Prikaz treninga po evaluacijskim mjerama

Tablica 8 prikazuje kako je korištenje kombinacije od pet tehnika rezultiralo time da performanse modela ostanu približno iste. Kombinacije s pet tehnika nisu uspjele ostvariti bolje rezultate od kombinacija s tri tehnike za augmentaciju podataka. Arhitekture FPN i DeepLabV3Plus su najbolje rezultate ostvarile koristeći kombinaciju nasumičnog horizontalnog i vertikalnog zaokretanja, nasumičnog brisanja, inverzije boja te nasumičnog pretvaranja slike u sive nijanse. Arhitektura U-net, koja je ostvarila



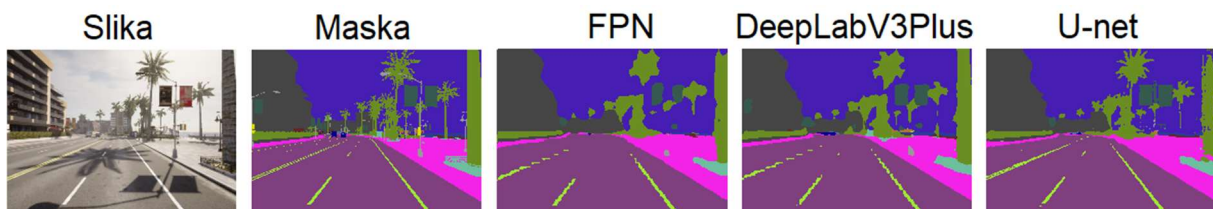
najbolje rezultate u usporedbi s ostalim arhitekturama, imala je najbolje performanse s modelom koji je koristio tehnike nasumičnog horizontalnog i vertikalnog zaokretanja, inverzije boja, brisanja i nasumične rotacije. Na Slici 17 su prikazane predikcije, a gubitak i rezultati treniranja po evaluacijskim mjerama su prikazani na Slici 18 i Slici 19. Iz grafa rezultata funkcije gubitka na Slici 18 je vidljivo kako je FPN jedini imao stabilno smanjenje rezultata funkcije gubitka, dok su druge dvije arhitekture imale veliku varijaciju. Analizom grafa na Slici 19 vidljivo je kako je svaka od mjera imala veliku varijaciju rezultata.

#### 8.4.4.5. Kombinacija svih augmentacijskih tehnika

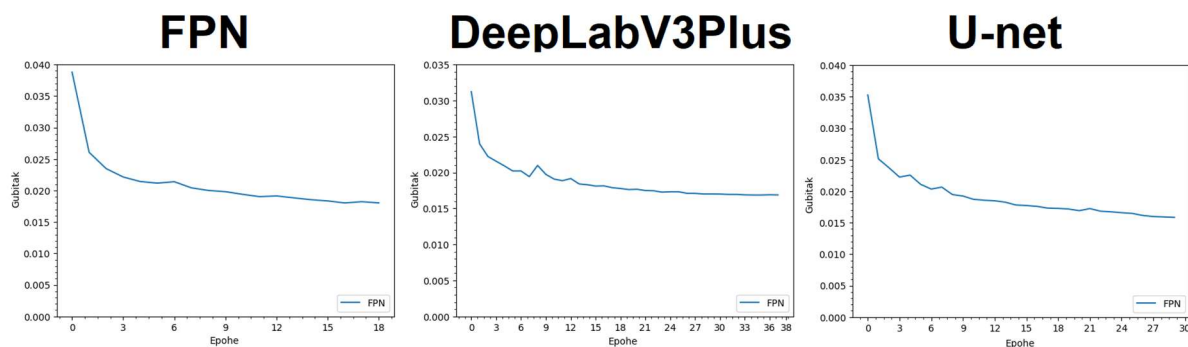
Sve spomenute augmentacijske tehnike su korištene tijekom treniranja u ovom dijelu istraživanja.

| Arhitektura mreža | Augmentacijska tehnika         | Mjera <sub>test</sub> |               |               |               |
|-------------------|--------------------------------|-----------------------|---------------|---------------|---------------|
|                   |                                | F mjera               | IoU           | Preciznost    | Odziv         |
| FPN               | NVZ, NHZ, NIB, NPSUSN, NB i NR | 0.8824                | 0.7918        | 0.938         | 0.8335        |
| DeepLabV3Plus     | NVZ, NHZ, NIB, NPSUSN, NB i NR | 0.882                 | 0.7913        | 0.934         | 0.8366        |
| U-net             | NVZ, NHZ, NIB, NPSUSN, NB i NR | <b>0.8931</b>         | <b>0.8091</b> | <b>0.9404</b> | <b>0.8509</b> |

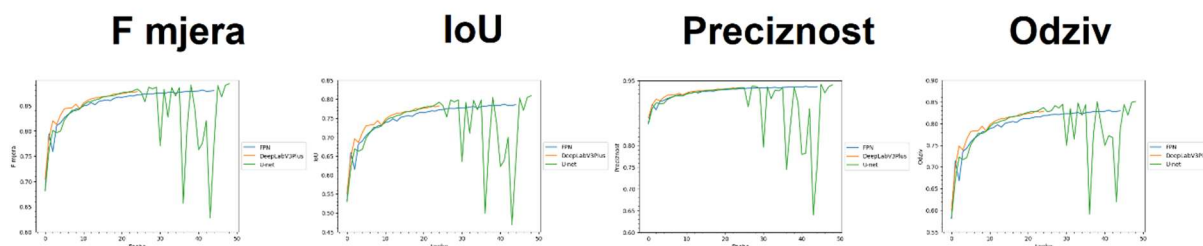
Tablica 9. Rezultati korištenja kombinacije svih augmentacijskih tehnika



Slika 20. Prikaz predikcija koristeći sve augmentacijske tehnike



Slika 21. Prikaz rezultata funkcija gubitaka po arhitekturi mreže



Slika 22. Prikaz treniranja po evaluacijskim mjerama

Korištenje svih navedenih augmentacijskih tehnika je rezultiralo približnim rezultatima kao kombinacija s pet tehnika. Na Slici 20 su prikazane predikcije, a rezultat funkcije gubitka i rezultati treniranja po evaluacijskim mjerama su prikazani na Slici 21 i Slici 22.

## 8.5. Zaključak

Cilj ovog istraživanja je bio napraviti usporedbu različitih augmentacijskih tehnika i njihovih kombinacija kod treniranja modela na tri različite arhitekture mreža u svrhu semantičke segmentacije na skupu slika iz prometa. Generirani skup podataka je dodatno augmentiran s navedenim tehnikama te su rezultati detaljno opisani u istraživanju. Nakon provedbe ove analize jasno je vidljivo da je upotreba arhitekture i augmentacijske tehnike vrlo bitna stavka u treniranju modela te da je njihov izbor ovisan o željenim ishodima modela. Usporedbom korištenja različitih kombinacija navedenih tehnika pokazalo se da je korištenje tehnika s preporučenim hiperparametrima najbolje u kombinaciji od tri tehnike (nasumična rotacija, nasumično



horizontalno zaokretanje te nasumično vertikalno zaokretanje) za arhitekture FPN i DeepLabV3Plus. Arhitektura U-net najbolje rezultate je ostvarila u kombinaciji od dvije tehnike (nasumična rotacija i nasumično horizontalno zaokretanje). Od navedenih arhitektura i njihovih kombinacija s kojima su ostvarile najbolje rezultate, najbolje predikcije je ostvarila kombinacija arhitekture U-net i tehnika nasumičnog horizontalnog zaokretanja te nasumične rotacije. Neovisno o objektu koji se promatra i klasificira, model s U-net arhitekturom i tehnikama nasumičnog horizontalnog zaokretanja te nasumične rotacije najbolje je označio objekte.

Iz navedenog može se zaključiti da dodavanje više augmentacijskih tehnika ne znači ujedno i bolje rezultate. Prilikom izbora tehnika za augmentaciju podataka važno je usmjeriti pažnju i na ostale elemente treninga i njihovu kompatibilnost u rješavanju zadatka segmentacije.

## 9. Zaključak diplomskog rada

Autonomna vožnja danas se sve više upotrebljava te je važno da se osigura sigurnost prijevoza svakog pojedinca koji sudjeluje kao putnik ili sudionik u prometu. Stvaranje sustava koji će moći čim bolje i sigurnije upravljati kroz promet jedna je od najvažnijih stavki kod kreiranja sustava za autonomnu vožnju. Semantička segmentacija i duboko učenje daju rješenje takvom problemu. U ovom radu sam istražio primjenu semantičke segmentacije na autonomnu vožnju.

Diplomski rad sadržava dva znanstvena rada. U prvom znanstvenom radu, koji je usmjeren na istraživanje najbolje funkcije gubitka za zadatak semantičke segmentacije, donijet je zaključak da je od pet korištenih funkcija na tri različite arhitekture mreže najbolja funkcija gubitka *Binary Cross entropy*. Pogotovo u kombinaciji s U-net arhitekturom pokazala je vrlo uvjerljive rezultate i impresivne predikcije. Drugi znanstveni rad nastavak je prvog rada te se na tri spomenute arhitekture mreža u kombinaciji s *Binary Cross entropy* funkcijom gubitka, dodaju šest različitih tehnika za augmentaciju podataka. U radu se prikazuju rezultati treniranja bez, s jednom tehnikom za augmentaciju podataka te kombinacija više različitih tehnika. Najbolje rezultate s arhitekturom mreže FPN i DeepLabV3Plus ostvarila je kombinacija tehnika nasumične rotacije, nasumičnog horizontalnog zaokretanja te nasumičnog vertikalnog zaokretanja. Za U-net arhitekturu, koja je ujedno ostvarila i najbolje rezultate u odnosu na ostale dvije arhitekture mreža, najbolje rezultate je ostvarila u kombinaciji s nasumičnom rotacijom i nasumičnim horizontalnim zaokretanjem.

Iako je ovaj rad donio dobre rezultate i zaključke, on je ograničen na određeni skup podataka i određenu domenu. Budući rad mogao bi se osloniti na daljnje istraživanje ovog područja te uključiti druge skupove podataka i provjeriti performanse modela na drugim domenama osim prometa.

## Popis literature

### Članci

1. Anaya-Isaza, A. i Mera-Jiménez, L., 2022. Data augmentation and transfer learning for brain tumor detection in magnetic resonance imaging. *IEEE Access*, 10, str. 23217-23233.
2. Araslanov, N. i Roth, S., 2021. Self-supervised augmentation consistency for adapting semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (str. 15384-15394).
3. Beauxis-Aussalet, E. i Hardman, L., 2014. Visualization of confusion matrix for non-expert users. In *IEEE Conference on Visual Analytics Science and Technology (VAST)-Poster Proceedings* (str. 1-2).
4. Beiker, S.A., 2012. Legal aspects of autonomous driving. *Santa Clara L. Rev.*, 52, str. 1145.
5. Berman, M., Triki, A.R. i Blaschko, M.B., 2018. The lovász-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (str. 4413-4421).
6. Boudissa, M., Kawanaka, H. i Wakabayashi, T., 2022, August. semantic segmentation of traffic landmarks using classical computer vision and U-Net model. In *Journal of Physics: Conference Series* (izdanje 2319, br. 1, str. 012031). IOP Publishing.
7. Caesar, H., Uijlings, J. i Ferrari, V., 2016. Region-based semantic segmentation with end-to-end training. In *Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14* (str. 381-397). Springer International Publishing.
8. Casado-García, Á., Domínguez, C., García-Domínguez, M., Heras, J., Inés, A., Mata, E. i Pascual, V., 2019. CLoDSA: a tool for augmentation in classification, localization, detection, semantic segmentation and instance segmentation tasks. *BMC bioinformatics*, 20, str. 1-14.
9. Caselles, L., Jailin, C. i Muller, S., 2022. Data augmentation for breast cancer mass segmentation. In *Proceedings of 2021 International Conference on Medical Imaging*

and Computer-Aided Diagnosis (MICAD 2021) Medical Imaging and Computer-Aided Diagnosis (str. 228-237). Springer Singapore.

10. Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F. i Adam, H., 2018. Encoder-decoder with atrous separable convolution for semantic image segmentation. In Proceedings of the European conference on computer vision (ECCV) (str. 801-818).

11. Chen, P., Liu, S., Zhao, H. i Jia, J., 2020. Gridmask data augmentation. arXiv preprint arXiv:2001.04086.

12. Cheng, H.X., Han, X.F. i Xiao, G.Q., 2022, July. Cenet: Toward concise and efficient lidar semantic segmentation for autonomous driving. In 2022 IEEE International Conference on Multimedia and Expo (ICME) (str. 01-06). IEEE.

13. Dai, J., He, K. i Sun, J., 2015. Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In Proceedings of the IEEE international conference on computer vision (str. 1635-1643).

14. Derczynski, L., 2016, May. Complementarity, F-score, i NLP Evaluation. In Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16) (str. 261-266).

15. Dolz, J., Desrosiers, C. i Ayed, I.B., 2018. 3D fully convolutional networks for subcortical segmentation in MRI: A large-scale study. *NeuroImage*, 170, str. 456-470.

16. Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A. i Koltun, V., 2017, October. CARLA: An open urban driving simulator. In Conference on robot learning (str. 1-16). PMLR.

17. Gábor, H., 2022. The Tversky loss function and its modifications for medical image segmentation.

18. Ghandorh, H., Boulila, W., Masood, S., Koubaa, A., Ahmed, F. i Ahmad, J., 2022. Semantic segmentation and edge detection—Approach to road detection in very high resolution satellite images. *Remote Sensing*, 14(3), str. 613.

19. Ghiasi, G., Cui, Y., Srinivas, A., Qian, R., Lin, T.Y., Cubuk, E.D., Le, Q.V. i Zoph, B., 2021. Simple copy-paste is a strong data augmentation method for instance segmentation. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (str. 2918-2928).

20. Guizard, N., Coupé, P., Fonov, V.S., Manjón, J.V., Arnold, D.L. i Collins, D.L., 2015. Rotation-invariant multi-contrast non-local means for MS lesion segmentation. *NeuroImage: Clinical*, 8, str. 376-389.

21. Guo, Y., Liu, Y., Georgiou, T. i Lew, M.S., 2018. A review of semantic segmentation using deep neural networks. *International journal of multimedia information retrieval*, 7, str. 87-93.
22. Harmouche, R., Subbanna, N.K., Collins, D.L., Arnold, D.L. i Arbel, T., 2014. Probabilistic multiple sclerosis lesion classification based on modeling regional intensity variability and local neighborhood information. *IEEE transactions on Biomedical Engineering*, 62(5), str. 1281-1292.
23. Huang, Z., Wang, X., Wang, J., Liu, W. i Wang, J., 2018. Weakly-supervised semantic segmentation network with deep seeded region growing. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (str. 7014-7023).
24. Illarionova, S., Nesteruk, S., Shadrin, D., Ignatiev, V., Pukalchik, M. i Oseledets, I., 2021. Object-based augmentation for building semantic segmentation: Ventura and santa rosa case study. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (str. 1659-1668).
25. Illarionova, S., Trekin, A., Ignatiev, V. i Oseledets, I., 2020. Neural-based hierarchical approach for detailed dominant forest species classification by multispectral satellite imagery. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14, str. 1810-1820.
26. Jadon, S., 2020, October. A survey of loss functions for semantic segmentation. In *2020 IEEE conference on computational intelligence in bioinformatics and computational biology (CIBCB)* (str. 1-7). IEEE.
27. Kamal, U., Tonmoy, T.I., Das, S. i Hasan, M.K., 2019. Automatic traffic sign detection and recognition using SegU-Net and a modified Tversky loss function with L1-constraint. *IEEE Transactions on Intelligent Transportation Systems*, 21(4), str. 1467-1479.
28. Kervadec, H., Bouchtiba, J., Desrosiers, C., Granger, E., Dolz, J. i Ayed, I.B., 2019, May. Boundary loss for highly unbalanced segmentation. In *International conference on medical imaging with deep learning* (str. 285-296). PMLR.
29. Khoreva, A., Benenson, R., Hosang, J., Hein, M. i Schiele, B., 2017. Simple does it: Weakly supervised instance and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (str. 876-885).
30. Körez, A., Barışçı, N., Çetin, A. i Ergün, U., 2020. Weighted ensemble object detection with optimized coefficients for remote sensing images. *ISPRS International Journal of Geo-Information*, 9(6), str. 370.

31. LeCun, Y., Chopra, S., Hadsell, R., Ranzato, M. i Huang, F., 2006. A tutorial on energy-based learning. *Predicting structured data*, 1(0), str. 10.
32. Li, L., Doroslovački, M. i Loew, M.H., 2020. Approximating the gradient of cross-entropy loss function. *IEEE access*, 8, str. 111626-111635.
33. Li, W., He, C., Fang, J., Zheng, J., Fu, H. i Yu, L., 2019. Semantic segmentation-based building footprint extraction using very high-resolution satellite images and multi-source GIS data. *Remote Sensing*, 11(4), str. 403.
34. Li, X., Sun, X., Meng, Y., Liang, J., Wu, F. i Li, J., 2019. Dice loss for data-imbalanced NLP tasks. *arXiv preprint arXiv:1911.02855*.
35. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B. i Belongie, S., 2017. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (str. 2117-2125).
36. Lin, T.Y., Goyal, P., Girshick, R., He, K. i Dollár, P., 2017. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision* (str. 2980-2988).
37. Liu, S., Zhang, J., Chen, Y., Liu, Y., Qin, Z. i Wan, T., 2019, May. Pixel level data augmentation for semantic image segmentation using generative adversarial networks. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (str. 1902-1906). IEEE.
38. Long, J., Shelhamer, E. i Darrell, T., 2015. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (str. 3431-3440).
39. Lucas, C., Kemmling, A., Mamlouk, A.M. i Heinrich, M.P., 2018, April. Multi-scale neural network for automatic segmentation of ischemic strokes on acute perfusion images. In *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)* (str. 1118-1121). IEEE.
40. Ma, J., 2020. Segmentation loss odyssey. *arXiv preprint arXiv:2005.13449*.
41. Ma, J., Chen, J., Ng, M., Huang, R., Li, Y., Li, C., Yang, X. i Martel, A.L., 2021. Loss odyssey in medical image segmentation. *Medical Image Analysis*, 71, str. 102035.
42. Malec, S., 2021. Semantic segmentation with carla simulator.
43. Nalepa, J., Marcinkiewicz, M. i Kawulok, M., 2019. Data augmentation for brain-tumor segmentation: a review. *Frontiers in computational neuroscience*, 13, str. 83.

44. Natan, O. i Miura, J., 2021, November. Semantic segmentation and depth estimation with RGB and DVS sensor fusion for multi-view driving perception. In Asian Conference on Pattern Recognition (str. 352-365). Cham: Springer International Publishing.
45. Nesti, F., Rossolini, G., Nair, S., Biondi, A. i Buttazzo, G., 2022. Evaluating the robustness of semantic segmentation for autonomous driving against real-world adversarial patch attacks. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (str. 2280-2289).
46. Olsson, V., Tranheden, W., Pinto, J. i Svensson, L., 2021. Classmix: Segmentation-based data augmentation for semi-supervised learning. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (str. 1369-1378).
47. Oreški, G. i Aničić, S., 2023. Usporedba funkcija gubitka za semantičku segmentaciju objekata u prometu. In Proceedings of the 46th International Convention on Information and Communication Technology, Electronics and Microelectronics, Opatija, Croatia (str. 1241-1248).
48. Papandreou, G., Chen, L.C., Murphy, K.P. i Yuille, A.L., 2015. Weakly-and semi-supervised learning of a deep convolutional network for semantic image segmentation. In Proceedings of the IEEE international conference on computer vision (str. 1742-1750).
49. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L. i Lerer, A., 2017. Automatic differentiation in pytorch.
50. Powers, D.M., 2020. Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. arXiv preprint arXiv:2010.16061.
51. Qiu, X., Sun, T., Xu, Y., Shao, Y., Dai, N. i Huang, X., 2020. Pre-trained models for natural language processing: A survey. Science China Technological Sciences, 63(10), str. 1872-1897.
52. Ren, S., He, K., Girshick, R. i Sun, J., 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems, str. 28.
53. Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I. i Savarese, S., 2019. Generalized intersection over union: A metric and a loss for bounding box regression. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (str. 658-666).

54. Ronneberger, O., Fischer, P. i Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation. In Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18 (str. 234-241). Springer International Publishing.
55. Salehi, S.S.M., Erdogmus, D. and Gholipour, A., 2017, September. Tversky loss function for image segmentation using 3D fully convolutional deep networks. In International workshop on machine learning in medical imaging (str. 379-387). Cham: Springer International Publishing.
56. Seferbekov, S., Iglovikov, V., Buslaev, A. i Shvets, A., 2018. Feature pyramid network for multi-class land segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (str. 272-275).
57. Shorten, C. i Khoshgoftaar, T.M., 2019. A survey on image data augmentation for deep learning. Journal of big data, 6(1), str. 1-48.
58. Siddique, N., Sidike, P., Elkin, C. i Devabhaktuni, V., 2020. U-Net and its variants for medical image segmentation: theory and applications. arXiv preprint arXiv:2011.01118.
59. Singh, S., 2015. Critical reasons for crashes investigated in the national motor vehicle crash causation survey (No. DOT HS 812 115).
60. Sokolova, M., Japkowicz, N. i Szpakowicz, S., 2006, December. Beyond accuracy, F-score and ROC: a family of discriminant measures for performance evaluation. In Australasian joint conference on artificial intelligence (str. 1015-1021). Berlin, Heidelberg: Springer Berlin Heidelberg..
61. Styner, M., Lee, J., Chin, B., Chin, M., Commowick, O., Tran, H., Markovic-Plese, S., Jewells, V. i Warfield, S., 2008. 3D segmentation in the clinic: A grand challenge II: MS lesion segmentation. MIDAS journal, 2008, str. 1-6.
62. Su, J., Anderson, S. i Mihaylova, L.S., 2022, October. A Deep Learning Method with Cross Dropout Focal Loss Function for Imbalanced Semantic Segmentation. In 2022 Sensor Data Fusion: Trends, Solutions, Applications (SDF) (str. 1-6). IEEE.
63. Su, Y., Sun, R., Lin, G. i Wu, Q., 2021. Context decoupling augmentation for weakly supervised semantic segmentation. In Proceedings of the IEEE/CVF international conference on computer vision (str. 7004-7014).
64. Syed, S., Anderssen, K. E., Stormo, S. K. i Kranz, M., 2023. Weakly supervised semantic segmentation for MRI: exploring the advantages and disadvantages of class



activation maps for biological image segmentation with soft boundaries. *Scientific Reports*, 13(1), str. 2574.

65. Tavera, A., Arnaudo, E., Masone, C. i Caputo, B., 2022. Augmentation invariance and adaptive sampling in semantic segmentation of agricultural aerial images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (str. 1656-1665).

66. Uijlings, J.R., Van De Sande, K.E., Gevers, T. i Smeulders, A.W., 2013. Selective search for object recognition. *International journal of computer vision*, 104, str. 154-171.

67. Wang, L., Wang, C., Sun, Z. i Chen, S., 2020. An improved dice loss for pneumothorax segmentation by mining the information of negative areas. *IEEE Access*, 8, str. 167939-167949.

68. Wang, P., Chen, P., Yuan, Y., Liu, D., Huang, Z., Hou, X. i Cottrell, G., 2018, March. Understanding convolution for semantic segmentation. In *2018 IEEE winter conference on applications of computer vision (WACV)* (str. 1451-1460). Ieee.

69. Wang, Q., Ma, Y., Zhao, K. i Tian, Y., 2020. A comprehensive survey of loss functions in machine learning. *Annals of Data Science*, str. 1-26.

70. Wang, Z., Zhao, Y., Tian, Y., Zhang, Y. i Gao, L., 2022, July. The Improved Deeplabv3plus Based Fast Lane Detection Method. In *Actuators* (izdanje 11, br. 7, str. 197). MDPI.

71. Wong, K.C., Moradi, M., Tang, H. i Syeda-Mahmood, T., 2018. 3D segmentation with exponential logarithmic loss for highly unbalanced object sizes. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2018: 21st International Conference, Granada, Spain, September 16-20, 2018, Proceedings, Part III 11* (str. 612-619). Springer International Publishing.

72. Yang, S., Xiao, W., Zhang, M., Guo, S., Zhao, J. i Shen, F., 2022. Image data augmentation for deep learning: A survey. *arXiv preprint arXiv:2204.08610*.

73. Yeung, M., Sala, E., Schönlieb, C.B. i Rundo, L., 2022. Unified focal loss: Generalising dice and cross entropy-based losses to handle class imbalanced medical image segmentation. *Computerized Medical Imaging and Graphics*, 95, str. 102026.

74. Yu, J., Xu, J., Chen, Y., Li, W., Wang, Q., Yoo, B. i Han, J.J., 2021, July. Learning generalized intersection over union for dense pixelwise prediction. In *International Conference on Machine Learning* (str. 12198-12207). PMLR.

75. Yu, X., Wu, X., Luo, C. i Ren, P., 2017. Deep learning in remote sensing scene classification: a data augmentation enhanced convolutional neural network framework. *GIScience & Remote Sensing*, 54(5), str. 741-758.
76. Yuan, W. i Xu, W., 2022. GapLoss: A Loss Function for Semantic Segmentation of Roads in Remote Sensing Images. *Remote Sensing*, 14(10), str. 2422.
77. Zhang, Y., Liu, S., Li, C. i Wang, J., 2021. Rethinking the dice loss for deep learning lesion segmentation in medical images. *Journal of Shanghai Jiaotong University (Science)*, 26, str. 93-102.
78. Zhao, A., Balakrishnan, G., Durand, F., Guttag, J.V. i Dalca, A.V., 2019. Data augmentation using learned transformations for one-shot medical image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (str. 8543-8553).
79. Zhou, D., Fang, J., Song, X., Guan, C., Yin, J., Dai, Y. and Yang, R., 2019, September. Iou loss for 2d/3d object detection. In *2019 international conference on 3D vision (3DV)* (str. 85-94). IEEE.
80. Zong, Z., Chen, C., Mi, X., Sun, W., Song, Y., Li, J., Dong, Z., Huang, R. i Yang, B., 2019. A deep learning approach for urban underground objects detection from vehicle-borne ground penetrating radar data in real-time. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42, str. 293-299.

## **Internetski izvori**

1. PYTORCH. (2017.) *Transforming and augmenting images*. [Online] Dostupan na: <https://pytorch.org/vision/stable/transforms.html>. [Pristupljeno: 16. travnja 2023.]

## Popis slika

|  |    |
|--|----|
| Slika 1. Slika i maska semantičke segmentacije.....                      | 3  |
| Slika 2: Kod za generiranje skupa podataka.....                          | 16 |
| Slika 3. Generiranje skupa podataka.....                                 | 24 |
| Slika 4. Prikaz predikcija svih modela na istoj slici .....              | 26 |
| Slika 5. Prikaz predikcija bez korištenja augmentacijskih tehnika.....   | 37 |
| Slika 6. Prikaz rezultata funkcija gubitaka po arhitekturi mreže.....    | 37 |
| Slika 7. Prikaz treninga po evaluacijskim mjerama.....                   | 37 |
| Slika 8. Prikaz predikcija koristeći jednu augmentacijsku tehniku.....   | 39 |
| Slika 9. Prikaz rezultata funkcija gubitaka po arhitekturi mreže.....    | 39 |
| Slika 10. Prikaz treninga po evaluacijskim mjerama.....                  | 39 |
| Slika 11. Prikaz predikcija koristeći dvije augmentacijske tehnike ..... | 41 |
| Slika 12. Prikaz rezultata funkcija gubitaka po arhitekturi mreže.....   | 41 |
| Slika 13. Prikaz treninga po evaluacijskim mjerama.....                  | 42 |
| Slika 14. Prikaz predikcija koristeći tri augmentacijske tehnike .....   | 44 |
| Slika 15. Prikaz rezultata funkcija gubitaka po arhitekturi mreže.....   | 44 |
| Slika 16. Prikaz treninga po evaluacijskim mjerama.....                  | 44 |
| Slika 17. Prikaz predikcija koristeći pet augmentacijskih tehnika.....   | 46 |
| Slika 18. Prikaz rezultata funkcija gubitaka po arhitekturi mreže.....   | 46 |
| Slika 19. Prikaz treninga po evaluacijskim mjerama.....                  | 46 |
| Slika 20. Prikaz predikcija koristeći sve augmentacijske tehnike.....    | 47 |
| Slika 21. Prikaz rezultata funkcija gubitaka po arhitekturi mreže.....   | 48 |
| Slika 22. Prikaz treninga po evaluacijskim mjerama.....                  | 48 |

## Popis tablica

|  |    |
|--|----|
| Tablica 1. Konfuzijska matrica (Izvor: Goutte i Gaussier, 2005).....           | 11 |
| Tablica 2. Rezultati usporedbe funkcija gubitka.....                           | 27 |
| Tablica 3. Popis svih augmentacijskih tehnika i njihovih skraćenih oznaka..... | 32 |
| Tablica 4. Rezultati treniranja bez uporabe augmentacijske tehnike .....       | 37 |
| Tablica 5. Rezultati korištenja jedne augmentacijske tehnike.....              | 38 |
| Tablica 6. Rezultati korištenja kombinacije dvije augmentacijske tehnike ..... | 41 |

|  |    |
|--|----|
| Tablica 7. Rezultati korištenja kombinacije tri augmentacijske tehnike.....    | 43 |
| Tablica 8. Rezultati korištenja kombinacije pet augmentacijskih tehnika.....   | 46 |
| Tablica 9. Rezultati korištenja kombinacije svih augmentacijskih tehnika ..... | 47 |

## Popis jednadžbi

|   |    |
|---|----|
| Jednadžba 1. Preciznost (Izvor: Goutte i Gaussier, 2005) .....  | 12 |
| Jednadžba 2. Odziv (Izvor: Goutte i Gaussier, 2005).....  | 12 |
| Jednadžba 3. F1 mjera (Izvor: Goutte i Gaussier, 2005).....   | 13 |
| Jednadžba 4. IoU mjera (Izvor: Yu et al., 2021) .....   | 13 |
| Jednadžba 5. Cross entropy funkcija (Izvor: izrada autora prema Jadon, 2020).....                             | 21 |
| Jednadžba 6. Binary Cross entropy funkcija (Izvor: izrada autora prema Jadon, 2020)<br>.....                  | 21 |
| Jednadžba 7. Dice funkcija (Izvor: Izrada autora prema Wang et al., 2020) .....                               | 22 |
| Jednadžba 8. Dice funkcija gubitka (Izvor: Izrada autora prema Wang et al., 2020)                             | 22 |
| Jednadžba 9. Jaccard/Intersection over union funkcija (Izvor: Izrada autora prema<br>Zhou et al., 2019) ..... | 22 |
| Jednadžba 10. Focal funkcija (Izvor: Jadon, 2020) .....   | 23 |
| Jednadžba 11. Tversky funkcija (Izvor: Izrada autora prema Salehi et al., 2017).....                          | 23 |

## **Sažetak na hrvatskom jeziku**

Ovaj rad se bavi proučavanjem, usporedbom i analizom često korištenih funkcija gubitaka i jednostavnih tehnika za augmentaciju podataka koje se mogu primijeniti na rješavanje problema semantičke segmentacije u autonomnoj vožnji. Rad najprije daje strukturirani pregled teorije na kojoj su bazirani znanstveni radovi, a kako bi se detaljnije razumio proces treniranja modela te objasnili korišteni pojmovi i različite tehnike. Drugi dio rada odnosi se na znanstvene radove u kojima se detaljno prikazuje korištenje različitih funkcija gubitaka te tehnika za augmentaciju podataka i njihovih ostvarenih rezultata na različitim modelima. Rezultati ovog rada doprinose svima koji žele naučiti više o spomenutim temama te su u radu dane i preporuke za korištenje funkcija gubitaka i tehnika za augmentaciju podataka u kombinaciji s predtreniranim modelima, ovisno o zadatku koji se nastoji ostvariti njihovim korištenjem.

**Ključne riječi:** semantička segmentacija, duboko učenje, autonomna vozila, funkcija gubitka, tehnike augmentacije podataka

## **Abstract**

This thesis deals with the study, comparison and analysis of commonly used loss functions and simple data augmentation techniques that can be applied to solving semantic segmentation problems in autonomous driving. The thesis first gives a structured overview of the theory on which the scientific papers are based, in order to understand the model training process in more detail and to explain the concepts and different techniques used. The second part of the thesis refers to scientific papers in which the use of different loss functions and techniques for data augmentation and their achieved results on different pre-trained models are presented in detail. The results of this work contribute to everyone who wants to learn more about the mentioned topics, and recommendations for the use of loss functions and techniques for data augmentation in combination with pre-trained models are given within the thesis, depending on the task that is sought to be achieved by using them.

**Keywords:** semantic segmentation, deep learning, autonomous vehicles, loss functions, data augmentation techniques