

# Izrada alata za automatsko označavanje objekata na slikama dobivenim iz simulatora autonomne vožnje

---

**Benčević, Zvonimir**

**Master's thesis / Diplomski rad**

**2022**

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

*Permanent link / Trajna poveznica:* <https://um.nsk.hr/um:nbn:hr:200:718217>

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

*Download date / Datum preuzimanja:* **2025-03-10**

*Repository / Repozitorij:*

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



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

**Sveučilišni studij**

**IZRADA ALATA ZA AUTOMATSKO OZNAČAVANJE  
OBJEKATA NA SLIKAMA DOBIVENIM IZ  
SIMULATORA AUTONOMNE VOŽNJE**

**Diplomski rad**

**Zvonimir Benčević**

**Osijek, 2022.**

**FERIT**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA  
I INFORMACIJSKIH TEHNOLOGIJA **OSIJEK****Obrazac D1: Obrazac za imenovanje Povjerenstva za diplomski ispit**

Osijek, 15.09.2022.

Odboru za završne i diplomske ispite

**Imenovanje Povjerenstva za diplomski ispit**

<b>Ime i prezime Pristupnika:</b>	Zvonimir Benčević
<b>Studij, smjer:</b>	Diplomski sveučilišni studij Automobilsko računarstvo i komunikacije
<b>Mat. br. Pristupnika, godina upisa:</b>	D-44ARK, 11.10.2020.
<b>OIB studenta:</b>	42785970865
<b>Mentor:</b>	Izv.prof.dr.sc. Ratko Grbić
<b>Sumentor:</b>	,
<b>Sumentor iz tvrtke:</b>	Borna Jelić
<b>Predsjednik Povjerenstva:</b>	Izv. prof. dr. sc. Mario Vranješ
<b>Član Povjerenstva 1:</b>	Izv.prof.dr.sc. Ratko Grbić
<b>Član Povjerenstva 2:</b>	Prof. dr. sc. Marijan Herceg
<b>Naslov diplomskog rada:</b>	Izrada alata za automatsko označavanje objekata na slikama dobivenim iz simulatora autonomne vožnje
<b>Znanstvena grana diplomskog rada:</b>	<b>Umjetna inteligencija (zn. polje računarstvo)</b>
<b>Zadatak diplomskog rada:</b>	<p>Velike količine označenih slika najčešće su potrebne za razvoj modernih modela temeljenih na dubokom učenju. Kako bi se izbjeglo ručno označavanje slika, u okviru diplomskog rada potrebno je napraviti alat koji će automatski označavati objekte od interesa pomoću graničnih pravokutnika u scenama dobivenim korištenjem simulatora. Korištenjem simulatora smatra se postavljanje okoliša (grad, otvorena cesta, vremenski uvjeti) te kretanje pomoću Ego vozila kroz simulaciju. Kada se na slici pojave objekti od interesa koji zadovoljavaju predefrirane uvjete (npr. veličina i položaj), alat treba &amp;quot;uslikati&amp;quot; scenu, označiti objekte te spremi sliku i datoteku s oznakama u prikladnom obliku za treniranje detektora objekata. Alat treba raditi paralelno sa simulatorom te se slike trebaju označavati i spremati u stvarnom vremenu. Tema rezervirana za: Zvonimir Benčević Sumentor iz tvrtke: Borna, Jelić (Institut RT-RK Osijek d.o.o.)</p>
<b>Prijedlog ocjene pismenog dijela ispita (diplomskog rada):</b>	Izvrstan (5)
<b>Kratko obrazloženje ocjene prema Kriterijima za ocjenjivanje završnih i diplomskih radova:</b>	Primjena znanja stečenih na fakultetu: 3 bod/boda Postignuti rezultati u odnosu na složenost zadatka: 3 bod/boda Jasnoća pismenog izražavanja: 2 bod/boda Razina samostalnosti: 3 razina
<b>Datum prijedloga ocjene od strane mentora:</b>	15.09.2022.
Potvrda mentora o predaji konačne verzije rada:	<i>Mentor elektronički potpisao predaju konačne verzije.</i>
	Datum: 27.9.2022.

**FERIT**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA  
I INFORMACIJSKIH TEHNOLOGIJA OSIJEK**IZJAVA O ORIGINALNOSTI RADA**

Osijek, 27.09.2022.

<b>Ime i prezime studenta:</b>	Zvonimir Benčević
<b>Studij:</b>	Diplomski sveučilišni studij Automobilsko računarstvo i komunikacije
<b>Mat. br. studenta, godina upisa:</b>	D-44ARK, 11.10.2020.
<b>Turnitin podudaranje [%]:</b>	3

Ovom izjavom izjavljujem da je rad pod nazivom: **Izrada alata za automatsko označavanje objekata na slikama dobivenim iz simulatora autonomne vožnje**

izrađen pod vodstvom mentora Izv.prof.dr.sc. Ratko Grbić

i sumentora ,

moj vlastiti rad i prema mom najboljem znanju ne sadrži prethodno objavljene ili neobjavljene pisane materijale drugih osoba, osim onih koji su izričito priznati navođenjem literature i drugih izvora informacija. Izjavljujem da je intelektualni sadržaj navedenog rada proizvod mog vlastitog rada, osim u onom dijelu za koji mi je bila potrebna pomoć mentora, sumentora i drugih osoba, a što je izričito navedeno u radu.

Potpis studenta:

# SADRŽAJ

<b>1. UVOD .....</b>	<b>1</b>
<b>2. GENERIRANJE PODATKOVNOG SKUPA POMOĆU SIMULATORA AUTONOMNE VOŽNJE .....</b>	<b>3</b>
2.1. Pregled postojećih skupova podataka s podacima iz stvarnoga svijeta .....	5
2.2. Pregled postojećih skupova sa sintetički generiranim podacima .....	6
2.3. Pregled postojećih alata za generiranje sintetičkih skupova podataka .....	9
<b>3. ALAT ZA AUTOMATSKO OZNAČAVANJE OBJEKATA NA SLIKAMA DOBIVENIM IZ SIMULATORA AUTONOMNE VOŽNJE .....</b>	<b>10</b>
3.1. Zahtjevi koje treba ispunjavati alat za automatsko označavanje objekata.....	10
3.2. Opis korištenih alata za izradu zadatka.....	11
3.3. Proces izrade alata za automatsko označavanje objekata na slikama .....	11
3.3.1. Pohrana slika .....	12
3.3.2. Označavanje objekata 3D graničnim okvirom.....	12
3.3.3. Transformacija 3D graničnog okvira na 2D ravninu kamere .....	13
3.3.4. Zapisivanje podataka o graničnom okviru.....	14
3.3.5. Brisanje nepotrebnih podataka .....	15
3.3.6. Dodavanje argumenata za korištenje alata.....	15
3.3.7. Zapisivanje parametara podatkovnog skupa.....	15
3.3.8. Dohvaćanje slika iz NumPy niza .....	15
3.3.9. Pregledavanje podatkovnog skupa.....	16
3.4. Upute za korištenje alata za automatsko označavanje objekata na slikama .....	16
<b>4. TESTIRANJE ALATA ZA AUTOMATSKO OZNAČAVANJE OBJEKATA NA SLIKAMA DOBIVENIM IZ SIMULATORA AUTONOMNE VOŽNJE.....</b>	<b>19</b>
<b>5. ZAKLJUČAK.....</b>	<b>25</b>
<b>LITERATURA .....</b>	<b>26</b>
<b>SAŽETAK.....</b>	<b>28</b>
<b>ABSTRACT .....</b>	<b>29</b>
<b>ŽIVOTOPIS.....</b>	<b>30</b>
<b>PRILOZI.....</b>	<b>31</b>

## 1. UVOD

Razvoj umjetne inteligencije i strojnog učenja omogućio je stvaranje umjetnih neuronskih mreža za rješavanje zadataka kao što su klasifikacija digitalnih slika i detekcija objekata na slikama. Umjetne neuronske mreže često se koriste u vozilima za realizaciju naprednih sustava za pomoć vozaču u vožnji (eng. *Advanced Driver Assistance System* - ADAS) u svrhu detekcije prometnih znakova, pješaka, drugih vozila i drugih objekata od interesa na temelju signala s kamere montirane na prednjoj strani vozila. Strojno učenje temelji se na podatkovnim primjerima odnosno učenju pomoću njih. Skup označenih podataka koji se koristi za učenje neuronske mreže naziva se podatkovni skup (eng. *Dataset*). U slučaju izrade detektora objekata, kao oznaka se podrazumijeva oznaka objekta na slici pomoću graničnog okvira (eng. *bounding box*) i klasa kojoj označeni objekt pripada. Za zadatke klasifikacije digitalnih slika i detekciju objekata na slikama najčešće se koriste konvolucijske neuronske mreže (eng. *Convolutional Neural Networks* - CNN). Za efikasnije učenje konvolucijskih neuronskih mreža potrebna je značajna količina označenih podataka što je ponekad praktičan problem, kao primjerice u području autonomne vožnje.

Kada se radi o autonomnoj vožnji, sustavi za detekciju objekata u prometu predstavljaju dio sigurnosno-kritičnih sustava automobila zato što ljudski životi mogu ovisiti o njihovim performansama. Od izrazite važnosti je postići što je moguće veću preciznost detekcije objekata u prometnoj okolini. Prikupljanje podataka iz stvarnog svijeta potrebnih za učenje neuronske mreže i označavanje istih može biti skup i dugotrajan posao jer se radi o višesatnom snimanju prometa pomoću automobila te ručnom označavanju svakog pojedinog objekta na snimljenim scenama. Kao jeftinije i brže rješenje pojavljuju se računalni simulatori stvarnoga svijeta pomoću kojih je moguće simulirati prometnu okolinu i vremenske uvjete. Budući da se radi o računalno generiranoj simulaciji, nije potrebno ručno označavati objekte jer su poznati svi podatci o objektima u simuliranoj okolini pa tako i njihova točna pozicija na slici. Stoga, korištenjem računalno generiranih simulatora stvarnoga svijeta moguće je generirati velike količine raznovrsnih podataka pomoću kojih se uče umjetne neuronske mreže.

U okviru ovog rada izrađen je alat koji automatski označava objekte od interesa pomoću graničnog okvira u scenama dobivenim korištenjem simulatora autonomne vožnje te se tako stvara sintetički podatkovni skup. Za potrebe izrade alata korišten je CARLA simulator autonomne vožnje koji nudi krajnjem korisniku modifikaciju parametara u simulaciji kao što su vremenski uvjeti i gustoća prometa. Korištenjem simulatora smatra se postavljanje okoliša (grad, otvorena cesta, vremenski uvjeti) te kretanje pomoću Ego vozila kroz simulaciju. Izraz Ego vozilo odnosi se na vozilo koje

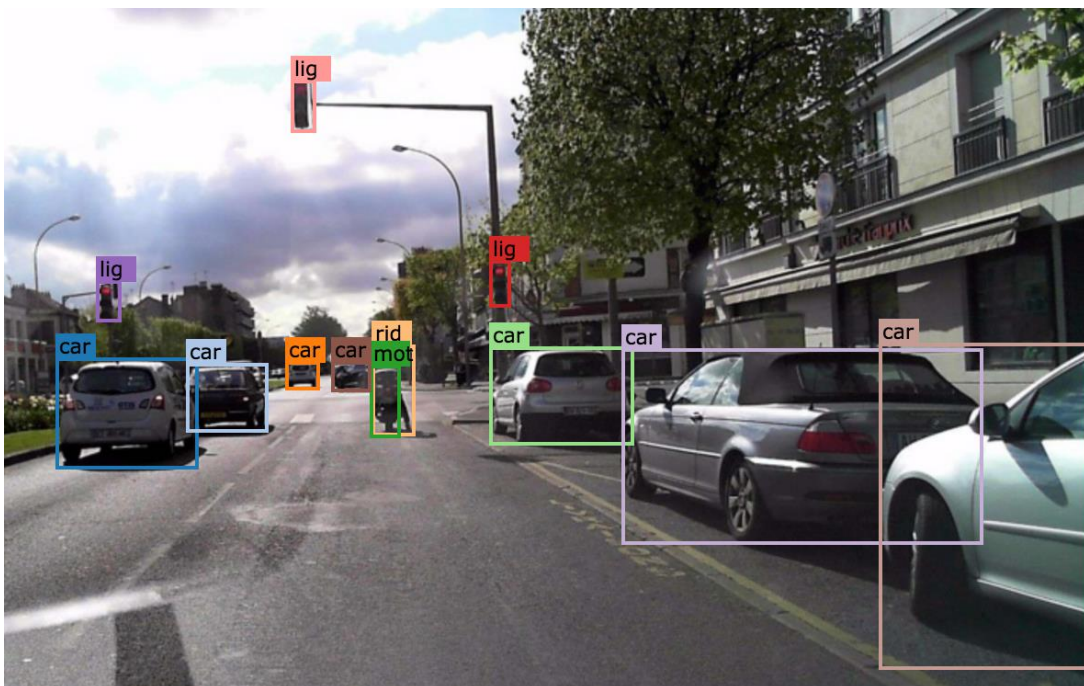
sadrži senzore koji percipiraju okolinu oko vozila. Kada se na slici pojave objekti od interesa koji zadovoljavaju predefiniране uvjete (npr. veličina i položaj), alat treba pohraniti sliku scene, označiti objekte od interesa te pohraniti sliku i datoteku s oznakama u prikladnom obliku za učenje detektora objekata. Alat radi paralelno sa simulatorom te se slike označavaju i spremaju u stvarnom vremenu. Alat omogućuje generiranje velike količine označenih podataka u relativno kratkom vremenu.

U drugom poglavlju opisan je problem izrade podatkovnog skupa koji sadrži označene digitalne slike dobivene iz stvarnoga svijeta te kako je generiranje sintetičkih podatkovnih skupova pomoću simulatora autonomne vožnje zapravo lakše i učinkovitije. Navedeni su postojeći skupovi podataka sa sintetički generiranim podacima odnosno s podacima generiranim pomoću simulatora autonomne vožnje. Treće poglavlje sadrži opis predloženog rješenja za automatsko označavanje objekata na slikama dobivenih iz simulatora autonomne vožnje. U četvrtom poglavlju predstavljeni su rezultati testiranja alata za automatsko označavanje objekata. Peto poglavlje sadrži zaključak.

## 2. GENERIRANJE PODATKOVNOG SKUPA POMOĆU SIMULATORA AUTONOMNE VOŽNJE

Duboke konvolucijske neuronske mreže zahtijevaju učenje velikog broja parametara nadziranim učenjem odnosno na temelju označenog podatkovnog skupa (tzv. trening skupa). Pri tome pod označeni skup podrazumijeva se da je za svaku sliku na raspolaganju podatak o tome gdje se točno nalazi objekt na slici odnosno temeljna istinita pozicija objekta na slici (eng. *ground truth*) i kojoj klasi pripada objekt. Te oznake se dobivaju ručnim označavanjem, odnosno osobe pregledavaju slike te označavaju objekte od interesa (vozila, pješake i sl.) graničnim okvirima. Očito da je takav posao dugotrajan i složen kada se radi o velikom skupu slika, a i podložan ljudskoj pogrešci (pogrešno označavanje, nedovoljno precizno označen granični okvir i sl.). Ručno označavanje podataka posebno je izazovno za semantičku segmentaciju budući da su potrebne oznake čija je preciznost na razini piksela.

U današnje vrijeme u modernijim vozilima nalazi se sve više ADAS sustava koji čine vožnju sigurnijom i udobnijom. Zajednica koja se bavi računalnim vidom, između ostalog, pridonosi razvoju ADAS-a i autonomne vožnje zbog brzog povećanja performansi alata koji se temelje na računalnom vidu kao što su detekcija objekata u prometu, prepoznavanje prometnih znakova, segmentacija cesta, itd. Primjer graničnog okvira i pripadajuće klase na slici dobivenoj iz prometne scene u stvarnom svijetu nalazi se na slici 2.1.



Slika 2.1. Primjer slike s oznakama [1]



U praksi, čak ni najbolji vizualni deskriptori, modeli klasa, metode kodiranja značajki i diskriminativne tehnike strojnog učenja nisu dovoljni za proizvodnju pouzdanih klasifikatora ako nisu dostupni ispravno označeni skupovi podataka s dovoljnom raznolikošću. Doista, ovo nije mali problem budući da je označavanje podataka zapravo naporan ljudski rad sklon pogreškama, čak je i angažiranje mnoštva ljudi za označavanje velikih količina slika netrivialan zadatak [2].

Kako bi se ublažio ovaj problem, postoje paradigme kao što su nenadzirano učenje kod kojega nije potrebno imati oznake, polu-nadzirano učenje kod kojega je potrebno samo nekoliko označenih podataka i aktivno učenje kod kojega je fokus na označavanje samo informativnih podataka. Navedene paradigme vrijede pod pretpostavkom da je posjedovanje označenih podataka (npr. slike) problematično, ali prikupljanje istih je jeftino. Za ADAS, prikupljanje takvih podataka je skupa aktivnost budući da se mora prijeći puno kilometara da bi se dobila dovoljna raznolikost podataka. Štoviše, dobro je poznato da nadzirano učenje s unaprijed određenim oznakama daje najtočnije modele s visokom efikasnošću.

U današnje vrijeme duboke konvolucijske neuronske mreže postale su nekakav standard prilikom rješavanja zadataka vezanih za računalni vid. Duboke konvolucijske neuronske mreže su dovele do značajnog povećanja performansi za razne vizualne zadatke [2,3]. Duboke konvolucijske neuronske mreže se temelje na vrlo nelinearnom treningu koji se sastoji od označenih sirovih podataka dobivenih s kamere kao i podataka s oznakama klasa što podrazumijeva učenje nekoliko milijuna parametara i sukladno tome zahtijevaju relativno veću količinu označenih podataka nego metode temeljene na ručno izrađenim vizualnim deskriptorima.

Postoje razni podaci koji se mogu smatrati sintetičkim u kontekstu strojnog učenja. Na primjer, uobičajena tehnika korištenja augmentacije za stvaranje varijacija u podacima tijekom treniranja je umjetno povećavanje trening skupa na temelju postojećih podatkovnih primjera. S druge strane postoje pristupi za stvaranje označenih podataka potpuno umjetnim putem. Sintetički skupovi podataka koji se koriste za zadatke računalnog vida općenito koriste računalnu grafiku kako bi stvorili slike koje se mogu koristiti za trening i validaciju sustava strojnog učenja. Sintetičke metode generiranja podataka općenito ističu nisku cijenu proizvodnje podataka kao glavnu prednost, ali fokus na tu prednost prikriva neke od možda bitnijih činjenica vezanih za sintetičke podatke. Sposobnost proizvodnje proizvoljnih količina podataka iz izvora s proizvoljnom vjerojatnosti pojavljivanja objekata od interesa i s proizvoljno detaljnim oznakama može biti vrlo korisno. Vjernost podataka generiranih pomoću simulatora može varirati u ovisnosti o izvedbi samoga simulatora. Dobiveni podatci iz simulatora autonomne vožnje trebaju biti što sličniji

stvarnim podacima jer se u konačnici modeli implementiraju i koriste u stvarnosti. U tom smislu, nisu svi pristupi generiranju sintetičkih podataka jednaki.

## **2.1. Pregled postojećih skupova podataka s podacima iz stvarnoga svijeta**

Čimbenici koji trenutno usporavaju razvoj novih podatkovnih skupova velikih razmjera poput ImageNeta [5] su vrijeme i novac. Unatoč tim čimbenicima, zajednica je uložila velike napore u stvaranje skupova podataka kao što su NYU-Depth V2 [6] (više od 1449 temeljito označenih slika), PASCAL-Context Dataset [7] (10103 temeljito označenih slika s preko 540 kategorija objekata), i MS COCO [8] (više od 300000 slika s oznakama za 80 kategorija objekata). Ovi skupovi podataka definitivno su pridonijeli jačanju istraživanja semantičke segmentacije unutarnjih scena kao i uobičajenih objekata, ali nisu prikladni za specifičnije zadatke kao što su oni uključeni u scenarije autonomne vožnje.

Kada se semantička segmentacija promatra u kontekstu autonomnih vozila, otkriveno je da je količina i raznolikost označenih slika urbanih scenarija mnogo niža u smislu ukupnog broja označenih elemenata slike i broja klasa. Dobar primjer je skup podataka CamVid [9] koji se sastoji od niza monokularnih slika snimljenih u Cambridgeu u Velikoj Britaniji. Međutim, samo 701 slika sadrži oznake na razini piksela u ukupno 32 kategorije objekata (kombinirajući objekte i arhitektonske scene), iako se obično koristi samo 11 najvećih kategorija. Slično tome, skup podataka Daimler Urban Segmentation [10] sadrži 500 potpuno označenih jednobojnih slika za 5 kategorija objekata. Noviji KITTI referentni skup podataka [11] pružio je veliku količinu slika urbanih scena iz grada Karlsruhe u Njemačkoj s temeljnim istinitim podacima za nekoliko zadataka. Međutim, sadrži samo 430 označenih slika za semantičku segmentaciju.

Ograničenje prijašnje spomenutih skupova podataka je to što nisu dovoljno raznoliki jer su slike prikupljane u jednom određenom gradu. Projekt LabelMe [12] ispravlja navedeni problem nudeći oko 1000 potpuno označenih slika urbanih sredina diljem svijeta i više od 3000 slika s nepreciznim oznakama. Veći skup podataka je CBCL StreetScenes [13], koji sadrži 3547 slika ulica Chicaga u 9 klasa s nepreciznim oznakama na razini piksela. Ovaj skup podataka nedavno je poboljšan u [14], poboljšavajući kvalitetu oznaka i dodajući dodatne klase. Jedan od većih skupova podataka za semantičku segmentaciju je i CityScapes [15], koji se sastoji od skupa slika prikupljenih u 50 gradova diljem Njemačke, Švicarske i Francuske u različitim godišnjim dobima i ima 5000 slika s preciznim oznakama i 20000 s manje preciznim oznakama u ukupno 30 klasa.

Kako se nalazimo u vremenu kada se generira jako velika količina informacija tako se generiraju i sve veći skupovi podataka. Jedan od trenutno najvećih skupova podataka je Mapillary Vistas

Dataset [16] skup podataka sa slikama visoke rezolucije na razini ulice koji sadrži 25000 slika visoke razlučivosti označenih u 66 kategorija objekata s dodatnim oznakama specifičnim za instance 37 klasa. Označava se u stilu korištenja poligona za ocrtavanje pojedinačnih objekata. Slike dolaze s različitih uređaja za snimanje (mobilni telefoni, tableti, akcijske kamere, profesionalna oprema za snimanje) i fotografa s različitim iskustvom. Na takav je način skup podataka dizajniran i sastavljen kako bi pokrio raznolikost, bogatstvo detalja i zemljopisni opseg. Kao zadane referentne zadatke, definira semantičku segmentaciju slike i segmentaciju slike specifične za instancu, s ciljem značajnog unapređenja razvoja najsuvremenijih metoda za vizualno razumijevanje prizora na cesti.

Trenutno najveći skup podataka s prometnim scenama je NEXET [17] podatkovni skup koji sadrži 50000 označenih slika za trening i testiranje te 41190 slika za validaciju. Slike u skupu podataka prikupljene su u 77 različitih zemalja diljem svijeta. Slike su snimane po noći i po danu te u različitim vremenskim uvjetima. Trošak skaliranja ove vrste projekta zahtijeva previsoka ekonomska ulaganja kako bi se snimile slike iz većeg broja zemalja, u različitim godišnjim dobima i različitim prometnim uvjetima. Iz tih razloga, obećavajuća alternativa predložena u ovom radu je korištenje alata za generiranje sintetičkih slika na kojima su simulirane stvarne urbane scene u velikom broju različitih uvjeta te generiranje odgovarajućih oznaka pomoću njega.

## **2.2. Pregled postojećih skupova sa sintetički generiranim podacima**

Upotreba sintetičkih podataka uvelike se povećala posljednjih godina zbog značajnog razvoja 3D računalne grafike koja vjerno reproducira stvarno okruženje. Sintetički podaci pomažu u rješavanju problema kao što je procjena izvedbe značajki slike pod određenim vrstama promjena. To znači da u je virtualnom okruženju moguće simulirati bilo koju promjenu u sceni koju bi bilo teško izvesti u pravom svijetu. U [18], autori su koristili virtualni svijet kako bi provjerili koliko su deskriptori otporni na promjenu scene, promjenu u osvjetljenju i promjenu perspektive gledišta. U području detekcije objekata slične su pristupe predložile različite skupine koristeći CAD modele, virtualne svjetove i proučavajući teme poput utjecaja realističnog svijeta na konačnu točnost detektora te važnost prilagodbe domene. Sintetički podaci također su korišteni za procjenu položaja [19] kako bi se nadoknadio nedostatak preciznih oznaka položaja objekata. Rješavanje problema semantičke segmentacije također je počelo imati koristi od ovog trenda sa segmentacijom virtualnih scena unutarnjeg okruženja kao što su prostorije u kućanstvu ili industrijski pogoni [20].

Skupovi sa sintetičkim podacima koji sadrže scene iz vožnje prikupljaju se pomoću grafičkih okruženja i simulatora u kojima su vjerno simulirani zakoni fizike. SYNTHIA [21] skup podataka sadrži 9400 fotorealističnih slika koje su generirane pomoću virtualnog grada napravljenog u simulatoru. Dolazi s oznakama na razini piksela u 13 različitih klasa. Svaka slika u SYNTHIA skupu podataka je veličine 1280x960 piksela. Synscapes [22] je sintetički skup podataka za raščlanjivanje uličnih scena stvoren korištenjem tehnika fotorealističnog prikazivanja i prikazuje najsvremenije rezultate za trening i testiranje, kao i nove vrste analiza. Skup podataka GTA5 (*eng. Grand Theft Auto 5*) [23] sadrži 24966 sintetičkih slika sa semantičkim oznakama s preciznošću na razini piksela. Slike su generirane pomoću videoigre Grand Theft Auto 5 i sve su iz perspektive automobila na ulicama virtualnih gradova u američkom stilu. Postoji 19 semantičkih klasa koje su kompatibilne s onima iz skupa podataka Cityscapes. Virtualni KITTI [24] je fotorealistični skup sintetičkih video podataka dizajniran za trening i testiranje modela za nekoliko zadataka: otkrivanje objekata i praćenje više objekata, semantička segmentacija na razini scene i na razini instance, optički tok i procjena dubine. Virtualni KITTI sadrži 50 monokularnih videozapisa visoke rezolucije (21260 slika) generiranih iz pet različitih virtualnih svjetova u urbanim okruženjima pod različitim slikovnim i vremenskim uvjetima. Ovi su svjetovi stvoreni korištenjem Unity game engine-a i nove metode kloniranja objekata iz stvarnog okruženja u virtualno okruženje. Ovi foto-realistični sintetički videozapisi automatski su, točno i potpuno označeni za 2D i 3D praćenje više objekata i s preciznošću na razini piksela s oznakama kategorije, instance, protoka i dubine. SHIFT [25] je sintetički skup podataka koji bilježi kontinuirano razvijajuću prirodu stvarnog svijeta kroz realistične diskretne i kontinuirane promjene u smjerovima okoliša kritičnim za sigurnost: doba dana, naoblaka, kiša, gustoća magle i gustoća vozila i pješaka. Generiran koristeći CARLA simulator, SHIFT uključuje opsežan paket senzora i pokriva najvažnije zadatke percepcije. Brojeći više od 4800 sekvenci snimljenih iz skupa senzora s više pogleda na 8 različitih lokacija. Podržava 13 zadataka percepcije za višezadačne upravljačke sustave: semantičku/instancnu segmentaciju, monokularnu/stereo dubinsku regresiju, 2D/3D otkrivanje objekata, 2D/3D višestruko praćenje objekata, optička procjena protoka, registracija oblaka točaka, vizualna odometrija, predviđanje putanje i procjena ljudskog položaja. To je jedan od najvećih sintetičkih skupova podataka za autonomnu vožnju i pruža detaljan skup bilješki i uvjeta za 2,5 milijuna slika. U tablici 2.1. nalaze se svi skupovi podataka spomenuti u ovome radu te njihove specifikacije.

**Tablica 2.1.** Prikaz podatkovnih skupova i njihovih parametara prikazanih u ovome radu

Skup podataka	Vrsta	Količina train slika	Količina validacijskih slika	Količina test slika	Tip oznaka	Broj klasa	Rezolucija slika
ImageNet	Stvarni podaci	1281167	50000	100000	Granični okvir	1000	prosječno 469x387
NYU-Depth	Stvarni podaci	1449 označenih slika i 407024 neoznačenih slika			Semantička segmentacija	nepoznat	640x480
PASCAL-Context	Stvarni podaci	10103	10103	9637	Semantička segmentacija	više od 540, najčešće 59	nepoznata
MS COCO	Stvarni podaci	83000	41000	41000	Semantička segmentacija	80	640x480
CamVid	Stvarni podaci	701			Semantička segmentacija	32	960x720
Daimler Urban Segmentation	Stvarni podaci	500			Semantička segmentacija	5	1024x440
KITTI	Stvarni podaci	12919			Semantička segmentacija i Granični okvir	11	1382x512
LabelMe	Stvarni podaci	više od 4000			Semantička segmentacija	2888	nepoznata
CBCL StreetScenes	Stvarni podaci	3547			Semantička segmentacija	9	1280x960
CityScapes	Stvarni podaci	5000 slika s preciznim oznakama i 20000 s manje preciznim oznakama			Semantička segmentacija	30	1024x2048
Mapillary Vistas	Stvarni podaci	25000			Semantička segmentacija	37	nepoznata(visoka rezolucija)
NEXET	Stvarni podaci	50000	5000	41190	Granični okvir	5	nepoznata
SYNTHIA	Sintetički podaci	9400			Semantička segmentacija	13	1280x960
Synscapes	Sintetički podaci	25000			Semantička segmentacija	19	1440x720
GTA5	Sintetički podaci	24966			Semantička segmentacija	19	1914x1052
Virtualni KITTI	Sintetički podaci	21260			Semantička segmentacija i Granični okvir	nepoznato	nepoznata(visoka rezolucija)
SHIFT	Sintetički podaci	2500000			Semantička segmentacija i Granični okvir	23	1280x800

### **2.3. Pregled postojećih alata za generiranje sintetičkih skupova podataka**

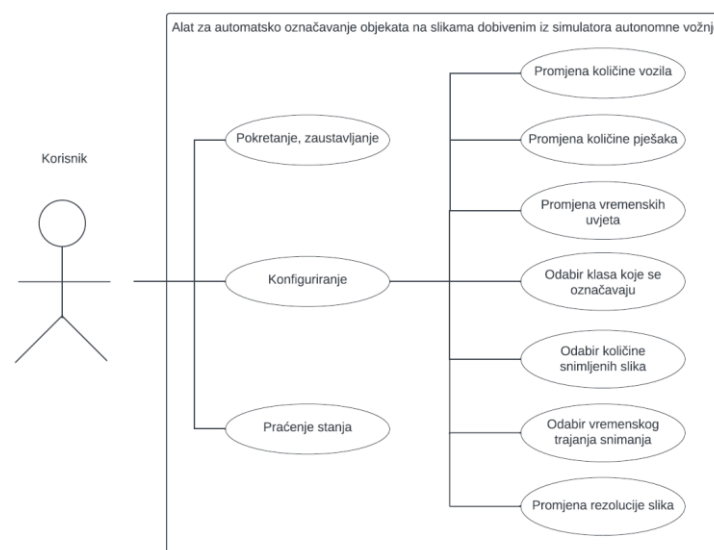
U današnje vrijeme postoje alati koji omogućavaju korisniku generiranje sintetičkog skupa podataka gdje korisnik u određenoj mjeri utječe na ishod generiranja odnosno na izgled podatkovnog skupa. Personaliziranje skupova podataka je glavni cilj takvih alata i može biti jako korisno u slučajevima kada postojeći skupovi podataka nisu zadovoljavajući za rješavanje postavljenog problema jer ne sadrže određene klase objekata odnosno određeni objekti nisu označeni. Pravi sintetički podaci čine veliku razliku u naprednoj percepciji i strojnom učenju. Mogu popuniti praznine u podacima ili nadopuniti snimke iz stvarnog svijeta u raznim industrijama. Cognata [26], izraelska start-up tvrtka, razvila je komercijalnu platformu za simulaciju koja iskorištava podatke mapa stvarnog svijeta za stvaranje točnog i realističnog virtualnog okruženja. Ideja tvrtke je ubrzati testiranje i validaciju umjetnih neuronskih mreža izgradnjom što detaljnijih 3D modela gradova. Prema [27] platforma radi na tri glavne razine, prvo je generiran sloj koji sadrži parametre svakog statičkog elementa, kao što su zgrade, ceste, prometni znakovi ili čak drveće. Zatim je generiran drugi sloj koji sadrži sve dinamičke informacije i modele kao što su ponašanje vozila i pješaka u prometu i vremenski uvjeti prethodno zabilježeni u stvarnom svijetu. Konačno, treći sloj posvećen je simulaciji svih senzora i interakciji s okolinom. Cognata simulacijska platforma omogućuje testiranje ADAS-a temeljeno na scenarijima koje razvojnim programerima nudi kompletna rješenja, uključujući strukturirane testne planove za provjeru performansi njihovih ADAS sustava. Svaki test može pružiti do stotine permutacija odjednom i omogućuje korisnicima prikupljanje različitih scenarija kako bi se njihove generirane simulacije mogle pokrenuti i analizirati kao dio jedne cjeline. Anyverse [28] je fleksibilna i točna platforma za generiranje sintetičkih podataka koja nudi mogućnost izrade podataka koji su potrebni za korisnički sustav u nekoliko minuta. Nudi mogućnost dizajniranja scenarija za bilo koji slučaj upotrebe s raznim varijacijama parametara te generiranje skupova podataka u oblaku. Koristeći Anyverse platformu moguće je simulirati bilo koji scenarij i pokriti cijeli niz krajnjih slučajeva s točnošću koja posljedično može poboljšati sustave temeljene na umjetnoj inteligenciji.

### 3. ALAT ZA AUTOMATSKO OZNAČAVANJE OBJEKATA NA SLIKAMA DOBIVENIM IZ SIMULATORA AUTONOMNE VOŽNJE

U ovome poglavlju opisani su zahtjevi koje alat za automatsko označavanje objekata na slikama treba ispunjavati nakon što bude izrađen. Opisana je programska podrška pomoću koje je izrađen alat za automatsko označavanje objekata na slikama dobivenih iz simulatora autonomne vožnje. Objasnjen je cijeli proces izrade alata korak po korak te su dane upute za korištenje alata.

#### 3.1. Zahtjevi koje treba ispunjavati alat za automatsko označavanje objekata

Alat za automatsko označavanje objekata treba sadržavati funkcije konfiguracije koje omogućuju korisniku generiranje označenog podatkovnog skupa s prometnim scenama i s određenim karakteristikama. Korisnik bi trebao moći kontrolirati sve uvjete u prometu koji utječu na izgled podatkovnog skupa kao što su gustoća prometa (količina pješaka i vozila) i vremenski uvjeti. Kako bi korisnik utjecao na izgled podatkovnog skupa, potrebno je omogućiti odabiranje klasa koje će biti označene u podatkovnom skupu (semafori, prometni znakovi, pješaci, vozila). Korisnik bi trebao moći kontrolirati veličinu podatkovnog skupa kojeg snima odnosno trebao bi moći utjecati na količinu snimljenih slika ili vrijeme snimanja podatkovnog skupa. Korisnik bi također trebao moći utjecati na rezoluciju slika u podatkovnom skupu. Kako bi se kontrolirao rad alata za automatsko označavanje objekata, potrebno je korisniku omogućiti praćenje stanja simulacije kao i parametara koje je korisnik odredio. Na slici 3.1. prikazan je dijagram slučaja upotrebe alata za automatsko označavanje objekata.



Slika 3.1. Dijagram slučaja upotrebe alata za automatsko označavanje objekata

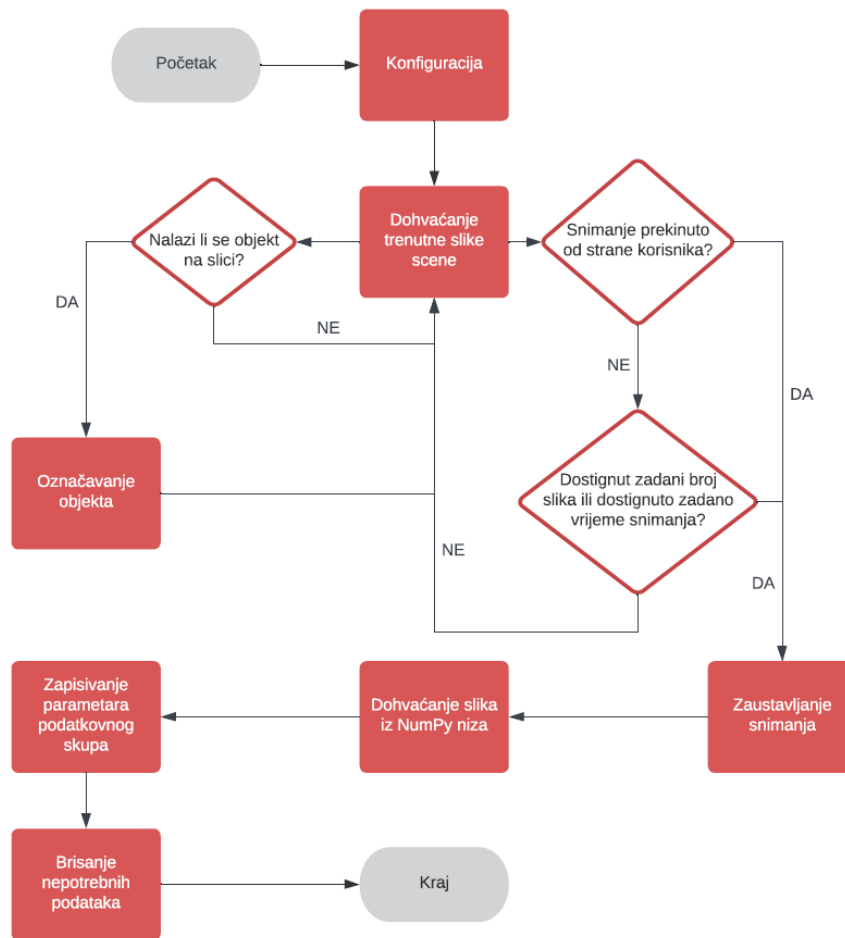
### **3.2. Opis korištenih alata za izradu zadatka**

Za potrebe izrade alata za automatsko označavanje objekata na slikama korišten je CARLA simulator [29]. CARLA simulator je razvijen kako bi podržao razvoj, trening i validaciju sustava za autonomnu vožnju. Uz otvoren kod i protokole, CARLA nudi otvorene digitalne resurse (urbane planove, zgrade, vozila) koji su stvoreni u tu svrhu i mogu se slobodno koristiti. Platforma za simulaciju podržava fleksibilnu specifikaciju skupova senzora, uvjete okoline, potpunu kontrolu svih statičkih i dinamičkih aktera (npr. semafori, pješaci, vozila), generiranje mapa i još mnogo toga. U ovome radu CARLA simulator je korišten kako bi se napravio alat koji će biti u mogućnosti generirati skup sintetičkih podataka gdje korisnik može personalizirati podatke odnosno može mijenjati razne parametre i uvjete izvođenja simulacije. Za pokretanje CARLA skripti potreban je Python. Python [30] je interpretirani, objektno orijentirani programski jezik visoke razine s dinamičkom semantikom. Njegove visoke razine ugrađenih podatkovnih struktura, u kombinaciji s dinamičkim određivanjem tipa varijable i dinamičkim vezanjem čine ga vrlo atraktivnim za brzi razvoj aplikacija te u svrhu korištenja kao skriptnog jezika za međusobno povezivanje postojećih komponenti. Jednostavna sintaksa Python programskog jezika koju je lako naučiti naglašava čitljivost te tako smanjuje troškove održavanja programa. Python podržava module i pakete, što potiče modularnost programa i ponovnu upotrebu koda. Python interpreter i opsežna standardna biblioteka dostupni su u izvornom ili binarnom obliku bez naknade za sve glavne platforme i mogu se besplatno distribuirati.

### **3.3. Proces izrade alata za automatsko označavanje objekata na slikama**

Nakon instaliranja CARLA-e i inicijalnog postavljanja radnog okruženja pregledane su sve skripte koje su došle uz simulator te je odabrana skripta `manual_control.py` koja je služila kao temelj za izradu alata za automatsko označavanje objekata. U toj skripti nalazile su se neke od funkcionalnosti koje bi trebale biti dio alata za automatsko označavanje objekata, a to su: ručna vožnja automobila, autopilot, promjena vremenskih uvjeta, snimanje slika u vožnji i promjena pozicije kamere. Funkcionalni blok dijagram alata za automatsko označavanje objekata na slikama prikazan je na slici 3.2.





**Slika 3.2.** Funkcionalni blok dijagram

### 3.3.1. Pohrana slika

Pohranjivanje slika u odabranoj skripti bilo je implementirano od strane razvojnih programera CARLA simulatora, a riješeno je tako što su se fotografije spremale na tvrdi disk. Međutim, to rješenje nije bilo prikladno za alat za automatsko označavanje objekata na slikama jer je bilo sporo. Drugo rješenje je bilo zapisivati slike u radnu memoriju te nakon određenog broja spremljenih slika, zapisati ih na tvrdi disk. To je riješeno pomoću NumPy niza gdje se slike spremaju u niz podataka i kada se spremi stotinu slika u jedan niz onda se cijeli niz sprema na tvrdi disk te se slike nastavljaju zapisivati u novi niz. Ovakav način zapisivanja slika poboljšava performanse alata u smislu brzine i efikasnosti.

### 3.3.2. Označavanje objekata 3D graničnim okvirom

Objekti koje bi trebalo biti moguće označiti na dohvaćenim trenutnim scenama iz simulatora pomoću alata za automatsko označavanje objekata su: vozila, pješaci, prometni znakovi i semafori.

Označavanje objekata na slikama s 3D graničnim okvirom započelo je prvo s označavanjem semafora. Pomoću metode koja dohvaća sve semafore i prometne znakove na cesti od trenutne lokacije Ego vozila do navedene udaljenosti, uz odgovarajući argument predan metodi, dohvaćeni su svi semafori do udaljenosti od 50 metara od Ego vozila. Zatim pomoću metode koja dohvaća sve dostupne 3D granične okvire za sve objekte, korištenjem odgovarajućeg argumenta, dohvaćeni su granični okviri za sve semafore na trenutnoj mapi. Pošto se na cesti nalaze semafori za sve sudionike u prometu, bilo je potrebno izolirati one semafore koji svojim radom utječu na kretanje Ego vozila. Semafori su izolirani tako što je određena rotacija semafora i rotacija stupa na kojem stoji semafor. Semafori koji utječu na Ego vozilo, zakrenuti su za 90 stupnjeva u odnosu na stup na kojem semafor stoji. Prometni znakovi označeni su na sličan način kao i semafori. Također su pomoću metode koja dohvaća sve orijentacijske objekte na cesti, s odgovarajućim argumentom, dohvaćeni prometni znakovi do udaljenosti od 50 metara od Ego vozila. Pomoću metode koja dohvaća sve granične okvire, korištenjem odgovarajućeg argumenta, dohvaćeni su granični okviri za sve prometne znakove na trenutnoj mapi. Određena je rotacija prometnog znaka te je određena rotacija Ego vozila. Prometni znakovi koji utječu na Ego vozilo zakrenuti su za 90 stupnjeva u odnosu na Ego vozilo. Ovaj način dohvaćanja prometnih znakova bio je potreban u pet mapa od osam dostupnih mapa. Na ostale tri mape nije bilo potrebno računati rotaciju prometnog znaka i automobila jer su znakovi postavljeni na ravnim dionicama ceste te na dionicama ceste gdje ne mogu utjecati na druge sudionike u prometu osim na Ego vozilo i ostala vozila koja se kreću u istome pravcu. Označavanje vozila graničnim okvirima podijeljeno je na označavanje statičnih vozila koja se generiraju zajedno s mapom te na označavanje dinamičnih vozila koja korisnik generira na mapu pri pokretanju skripte. Granični okviri dohvaćeni su pomoću metode koja dohvaća granične okvire od svih vozila na mapi, a zatim su posebno dohvaćena statična vozila i dinamična vozila pomoću određenih metoda. Granični okviri vozila su također dohvaćeni samo za vozila koja se nalaze unutar 50 metara od Ego vozila.

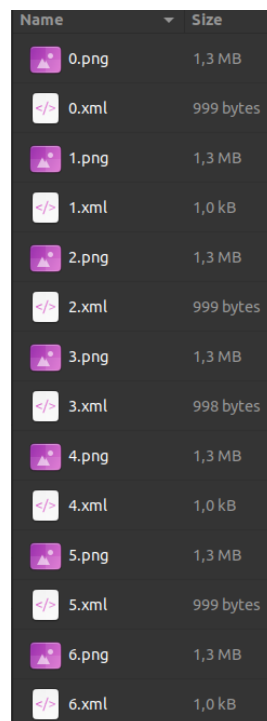
### **3.3.3. Transformacija 3D graničnog okvira na 2D ravninu kamere**

Kako bi se na snimljenim trenutnim scenama simulacije mogao označiti objekt s graničnim okvirom koji je dvodimenzionalan, potrebno je dohvatiti krajnje točke 3D graničnog okvira iz simulacije i projicirati ih na 2D ravninu kamere. Prvo je potrebno konstruirati matricu projekcije kamere zatim koristiti projekcijsku matricu kamere za projiciranje 3D točaka u 2D točke. Prvi korak je transformirati 3D koordinate simulacijskog svijeta u koordinate kamere, koristeći inverznu transformaciju kamere koja se može dohvatiti pomoću određene metode. Nakon toga potrebno je koristiti matricu projekcije kamere za projiciranje 3D točaka u koordinatama kamere

u ravninu 2D kamere. Nakon što se naprave funkcije za projiciranje 3D točaka u 2D točke potrebno je dohvatiti parametre kamere. Svi CARLA objekti imaju pridruženi granični okvir. Vrhovi graničnog okvira mogu se dohvatiti pomoću jedne od zadanih funkcija. Kako bi se nacrtali granični okvir na ravninu kamere, potrebno je spojiti vrhove odgovarajućim redoslijedom kako bi se stvorile stranice pravokutnika odnosno graničnog okvira.

### 3.3.4. Zapisivanje podataka o graničnom okviru

Kako bi koristili granične okvire u svrhu treniranja neuronske mreže, potrebno je zapisati njihove podatke. Postoji niz različitih formata koji se koriste za autonomnu vožnju i detekciju objekata, kao što su KITTI, PASCAL VOC ili MicroSoft COCO. U ovome zadatku korišten je PASCAL VOC zapis podataka. Za zapisivanje podataka o graničnim okvirima korištena je biblioteka pod nazivom `pascal-voc-writer` [31]. Ova se biblioteka može koristiti za stvaranje XML datoteka s parametrima slika u formatu PASCAL VOC. Za svaku spremljenu sliku simulacije generira se popratna XML datoteka koja sadrži pojedinosti o graničnim okvirima na slici. Pomoću navedene biblioteke zapisani su podaci o graničnim okvirima te su označeni objekti klasificirani u ovisnosti o tipu objekta koji se nalazi unutar graničnog okvira. Na slici 3.3. prikazan je izgled podatkovnog skupa nakon što se spremi na tvrdi disk.



Name	Size
0.png	1,3 MB
0.xml	999 bytes
1.png	1,3 MB
1.xml	1,0 kB
2.png	1,3 MB
2.xml	999 bytes
3.png	1,3 MB
3.xml	998 bytes
4.png	1,3 MB
4.xml	1,0 kB
5.png	1,3 MB
5.xml	999 bytes
6.png	1,3 MB
6.xml	1,0 kB

**Slika 3.3.** Prikaz dijela zapisa podatkovnog skupa na tvrdom disku

### 3.3.5. Brisanje nepotrebnih podataka

Kako bi se izbjeglo da u podatkovnom skupu postoje slike na kojima nema označenih objekata napravljen je algoritam koji nakon snimanja podatkovnog skupa pregleda sve datoteke u kojima se nalaze zapisi o graničnim okvirima te u slučaju da u nekoj od datoteka ne postoji zapis o graničnom okviru tada se briše ta datoteka i pripadajuća slika.

### 3.3.6. Dodavanje argumenata za korištenje alata

Alat za automatsko označavanje objekata služi kako bi korisnik imao kontrolu nad parametrima podatkovnog skupa odnosno kako bi se mogao generirati podatkovni skup koji odgovara želji korisnika. Dodavanjem argumenata nudi se korisniku mogućnost izbora parametara koji utječu na izgled podatkovnog skupa. Moguće je odabrati željene klase objekata za koje će se spremati podaci o graničnim okvirima. Parametri koje je moguće mijenjati kako bi se utjecalo na gustoću prometa odnose se na količinu pješaka i vozila na cesti. Tako je moguće direktno utjecati na količinu označenih objekata u podatkovnom skupu. Prije nego se pokrene snimanje podatkovnog skupa moguće je odrediti broj slika koji će biti snimljen i željeno vrijeme snimanja podatkovnog skupa. Moguće je mijenjati rezoluciju slika koje će se spremati.

### 3.3.7. Zapisivanje parametara podatkovnog skupa

Nakon što se podatkovni skup snimi, svi bitni parametri su spremljeni u tekstualnu datoteku kako bi, pri pregledavanju podatkovnog skupa, bilo moguće provjeriti pod kojim uvjetima je sniman podatkovni skup. Na slici 3.4. prikazan je izgled datoteke u koju su zapisani parametri podatkovnog skupa.

```
Rezolucija slika je: 1280x720
Objekti koji su označeni: traffic_light, traffic_sign, pedestrian, vehicle
Vremenski uvjeti: Clear Noon
Vrijeme snimanja dataseta je: 112 sekundi
Broj snimljenih slika: 1000
```

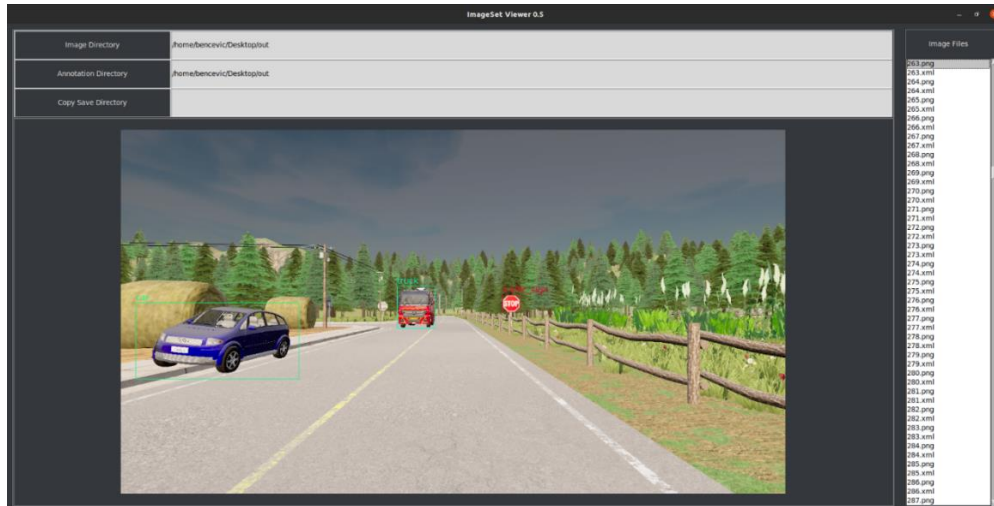
**Slika 3.4.** *Parametri podatkovnog skupa*

### 3.3.8. Dohvaćanje slika iz NumPy niza

Nakon što se slike zapišu na tvrdi disk u obliku NumPy nizova, tada je potrebno dohvatiti slike iz NumPy nizova kako bi se one mogle pregledavati i koristiti. Dohvaćanje slika je riješeno tako da se nakon snimanja slika i zapisivanja NumPy nizova na tvrdi disk pozove skripta koja čita podatke iz svakog pojedinačnog niza i zapisuje ih na tvrdi disk u obliku slika.

### 3.3.9. Pregledavanje podatkovnog skupa

Za pregledavanje i provjeravanje valjanosti podatkovnog skupa korišten je alat ImageSetViewer [32]. ImageSet Viewer je program s grafičkim korisničkim sučeljem koji vizualizira označene okvire i kategorije za Pascal VOC format zapisa detekcije objekata. Na slici 3.5. prikazan je izgled vizualizacije graničnih okvira na objektima od interesa pomoću alata ImageSet Viewer.



Slika 3.5. Prikaz podatkovnog skupa pomoću alata ImageSet Viewer

## 3.4. Upute za korištenje alata za automatsko označavanje objekata na slikama

Potrebno je instalirati Python prema uputama na web stranici [33], također je potrebno instalirati CARLA simulator prema uputama na web stranici [34]. Nakon instaliranja i pripremanja radnog okruženja potrebno je preuzeti datoteku odnosno alat za automatsko označavanje objekata na slikama, poveznica se nalazi u prilogu P.3.1 i datoteku za dohvaćanje podataka iz NumPy nizova, poveznica se nalazi u prilogu P.3.2. U datoteci u kojoj je napisan kod za alat za automatsko označavanje objekata potrebno je izmijeniti putanju na kojoj se nalazi mapa u koju se spremaju slike i pripadajući XML zapisi, putanju gdje će se spremati tekstualna datoteka s parametrima podatkovnog skupa i putanju gdje se nalazi datoteka za dohvaćanje podataka iz NumPy nizova. Nakon inicijalnog postavljanja putanja potrebno je pokrenuti CARLA simulator. Pokretanje CARLA simulatora i odgovarajućih skripti izvodi se pomoću terminala. Datoteka za pokretanje CARLA simulatora nalazi se u mapi `carla-simulator` i pokreće se naredbom `./CarlaUE4.sh`. Nakon pokretanja server strane CARLA simulatora, moguće je promijeniti mapu koja je generirana na serveru. Kako bi se mapa promijenila potrebno je pozicionirati se u

mapu `/carla-simulator/PythonAPI/util/` i pokrenuti konfiguracijsku datoteku naredbom `python3 config.py -m ime_mape`. Argument `-m` u naredbi označava mapu te nakon argumenta potrebno je upisati ime željene mape. Argumentom `--list` moguće je provjeriti sve dostupne mape koje je moguće pokrenuti u CARLA simulatoru. Nakon što je odabrana mapa koja će se izvoditi u simulaciji potrebno je pozicionirati se u mapu gdje se nalazi datoteka koja sadrži alat za automatsko označavanje objekata. Pomoću naredbe `/usr/bin/python3 alat_za_automatsko_označavanje_objekata.py` te u nastavku željeni argument, moguće je pokrenuti alat i započeti korištenje. U tablici 3.1. prikazani su argumenti pomoću kojih je moguće konfigurirati alat za automatsko označavanje objekata.

**Tablica 3.1.** Argumenti za upravljanje alatom za automatsko označavanje objekata na slikama

Naziv	Argument	Opis
Spremanje željenih klasa	<code>-c</code> ili <code>--classes</code>	Klase se odabiru tako da se upiše željena ključna riječ koja odgovara određenoj klasi, argument je tip podataka string te je bitno da su riječi točno napisane, a mogu biti odvojene razmakom ili zarezom. Za semafore ključna riječ je <code>traffic_light</code> , za znakove je <code>traffic_sign</code> , za vozila <code>vehicle</code> i za pješake <code>pedestrian</code> .
Količina vozila	<code>-n</code> ili <code>--number-of-vehicles</code>	Definira broj vozila koji se stvori na mapi, zadana vrijednost je 10 vozila. Obratiti pažnju na to da se broj sličica u sekundi smanjuje što je više vozila s autopilotom na mapi.
Količina pješaka	<code>-w</code> ili <code>--number-of-walkers</code>	Definira broj pješaka koji se stvori na mapi, zadana vrijednost je 10 pješaka.
Rezolucija slika	<code>--res</code>	Rezolucija slika upisuje se kao „širinaxvisina“, zadana vrijednost je 1280x720.
Trajanje snimanja	<code>--duration</code>	Definira trajanje snimanja podatkovnog skupa u sekundama, zadana vrijednost je 18000 sekundi ili 5 sati. Zadano vrijeme je dugačko kako se snimanje podatkovnog skupa ne bi iznenada prekinulo u slučaju da korisnik ne zada željeno vrijeme snimanja podatkovnog skupa.

Količina snimljenih slika	--number-of-images	Definira broj snimljenih slika za vrijeme trajanja snimanja podatkovnog skupa, zadani broj slika je jedan milijun. Zadani broj slika je velik kako se snimanje podatkovnog skupa ne bi iznenada prekinulo u slučaju da korisnik ne zada željeni broj snimljenih slika. Treba napomenuti kako broj snimljenih slika i završni broj snimljenih slika nije isti jer se brišu slike koje na sebi nemaju nikakav označeni objekt.
---------------------------	--------------------	--

U tablici 3.2. nalaze se funkcionalnosti s kojima je moguće upravljati pomoću tipkovnice nakon pokretanja alata za automatsko označavanje objekata.

**Tablica 3.2.** *Funkcionalnosti kojima se upravlja pomoću tipkovnice*

Tipka	Opis
W	Gas
S	Kočnica
A/D	Skretanje lijevo i desno
Razmak	Ručna kočnica
Q	Vožnja unatrag
P	Autopilot
TAB	Promjena pozicije kamere
C	Promjena vremenskih uvjeta (Shift+C, promjena unatrag)
R	Pokretanje i zaustavljanje snimanja

Jedan od primjera kako bi se pokrenuo alat u slučaju da korisnik zahtjeva da na mapi bude 20 vozila i 15 pješaka i da samo oni budu označeni uz to da se podatkovni skup snima 5 minuta:

```
/usr/bin/python3 alat_za_automatsko_oznacavanje_objekata.py -n 20
-w 15 -c vehicle,pedestrian --duration 300
```

#### 4. TESTIRANJE ALATA ZA AUTOMATSKO OZNAČAVANJE OBJEKATA NA SLIKAMA DOBIVENIM IZ SIMULATORA AUTONOMNE VOŽNJE

Konfiguracija računala na kojem je testiran alat za automatsko označavanje objekata nalazi se u tablici 4.1.

**Tablica 4.1.** Konfiguracija testnog računala

Operacijski sustav	Ubuntu 20.04.3 LTS
Procesor	Intel Core i7-6700 @ 3.40GHz x 8
Grafička kartica	NVIDIA GeForce GTX 1070 Ti
Radna memorija	16 GB
Kapacitet diska	1,3 TB

Testiranje alata za automatsko označavanje objekata provedeno je na svih osam dostupnih mapa u CARLA simulatoru te je na svakoj mapi provedeno snimanje 3 puta. Snimanje je svaki puta provedeno sa zadanim postavkama za broj pješaka i vozila te za rezoluciju slike. Svaki puta su označavane sve klase objekata te je svaki puta snimljeno tisuću slika. Nakon što je pokrenuta skripta pokrenuo se autopilot i započelo je snimanje. Svaki puta kada se snima podatkovni skup, pozicioniranje Ego vozila bude nasumično odnosno gdje se stvori prvotno na mapi, od te pozicije kreće snimanje. Podaci koji su zabilježeni su: broj pojavljivanja semafora, broj pojavljivanja prometnih znakova, broj pojavljivanja pješaka, broj pojavljivanja vozila, najmanji granični okvir, najveći granični okvir i prosječna veličina graničnog okvira. Slike snimljene tijekom testiranja su rezolucije 1280x720 piksela. Rezultati testiranja alata za automatsko označavanje objekata nalaze se u tablici 4.2. Pristup snimljenim podatkovnim skupovima moguće je ostvariti na upit, a poveznica se nalazi u prilogu P.4.1.

**Tablica 4.2.** Rezultati testiranja alata za automatsko označavanje objekata

Mapa	Broj pojavljivanja semafora	Broj pojavljivanja prometnih znakova	Broj pojavljivanja pješaka	Broj pojavljivanja vozila	Najmanji granični okvir	Najveći granični okvir	Prosječni granični okvir
Town_01	423	287	97	155	4x17	380x180	34x51
	535	129	0	107	1x3	571x299	33x46
	273	150	391	29	4x17	359x257	20x46



Town_02	1103	376	137	182	5x13	605x476	34x52
	1007	351	116	473	4x18	534x363	38x53
	887	606	220	298	5x11	362x189	31x47
Town_03	2054	0	85	45	2x8	131x20	7x14
	3336	129	94	87	2x8	205x122	7x15
	2370	0	0	445	2x8	340x155	23x18
Town_04	360	683	0	66	2x7	165x43	20x17
	340	289	0	11	3x6	181x55	11x12
	0	438	0	86	1x2	418x199	28x23
Town_05	3368	10	39	448	2x5	326x178	17x17
	604	314	0	12	3x8	105x60	11x16
	3152	0	150	522	2x8	198x98	15x17
Town_06	1193	875	32	123	2x3	106x131	14x20
	348	528	0	9	3x8	105x103	14x17
	572	504	0	464	2x6	309x129	29x21
Town_07	7	1036	0	108	1x2	405x202	40x31
	476	655	74	799	2x9	338x317	50x41
	591	335	129	25	3x9	118x171	17x32
Town_10	385	0	147	1080	2x12	493x291	48x39
	1195	0	209	711	3x10	304x143	43x36
	1378	0	359	1158	2x10	605x415	42x36

Rezultati testiranja ovise o tome gdje se stvorilo Ego vozilo na mapi te kojom rutom je autopilot odlučio ići pa tako na određenim rutama koje vode na autocestu rijetko će se pojaviti semafor ili pješak dok na rutama koje idu kroz grad ima puno semafora i pješaka. Na velikim mapama poput Town\_04, Town\_05 i Town\_06 rijetko će Ego vozilo naići na drugo vozilo i pješaka jer je stvoreno samo 10 vozila i 10 pješaka na mapi. Na mapi Town\_04 niti jedan pješak nije označen zato što je autopilot sva tri snimanja odlučio ići na autocestu. Količina pojavljivanja prometnih znakova ovisi o tome koliko ih je na mapi i hoće li Ego vozilo proći kraj njih. Može se primijetiti kako na mapi Town\_10 nema niti jednog detektiranog prometnog znaka. To je zato što na mapi postoje samo 3 prometna znaka koja se nalaze na istome raskrižju tako da je vrlo mala mogućnost da se snime ti prometni znakovi odnosno da Ego vozilo prođe kroz to raskrižje. Veličina graničnog okvira ovisi o udaljenosti na kojoj se označava objekt od interesa, svi granični okviri u testiranju označavani su na objektima od interesa koji su na udaljenosti od 50 metara ili bliže od Ego vozila.

Iz rezultata se može primjetiti da postoje vrlo mali označeni objekti čiji je granični okvir veličine 20-ak piksela ili manje. Tu se radi o prometnim znakovima i semaforima koji su na udaljenosti blizu 50 metara od Ego vozila, ali su uz korištenu rezoluciju od 1280x720 piksela na slici vrlo mali i teško se razaznaju pregledavanjem snimljenih slika. Navedeni problem moguće je riješiti povećanjem rezolucije kamere ili mijenjanjem ograničenja s obzirom na udaljenost u programskom kodu.

Kako bi se prikazalo da broj pješaka i vozila uistinu ovisi o tome koliko ih je stvoreno na mapi, provedena su dodatna mjerenja na mapi Town\_01. Prvi set mjerenja proveden je uz veću gustoću prometa odnosno na mapi je stvoreno 25 vozila i 25 pješaka dok je drugi set mjerenja proveden uz manju gustoću prometa odnosno na mapi je stvoreno 5 vozila i 5 pješaka. Rezultati mjerenja za gušći i rjeđi promet prikazani su u tablici 4.3.

**Tablica 4.3.** Rezultati testiranja alata za automatsko označavanje objekata uz veću i manju gustoću prometa

Mapa	Broj pojavljivanja semafora	Broj pojavljivanja prometnih znakova	Broj pojavljivanja pješaka	Broj pojavljivanja vozila	Najmanji granični okvir	Najveći granični okvir	Prosječni granični okvir
Town_01 gušće	570	168	494	249	4x17	472x254	34x56
	863	57	378	888	3x8	678x370	52x54
	657	94	535	253	4x17	598x299	34x54
Town_01 rjeđe	775	99	89	49	4x17	122x258	21x41
	590	126	116	94	4x17	293x117	30x48
	718	53	0	15	4x17	491x231	19x49

Iz dobivenih rezultata mjerenja s gušćim i rjeđim prometom možemo zaključiti da se uz gušći promet pojavljuje više označenih pješaka i vozila dok ih za rjeđi promet ima manje.

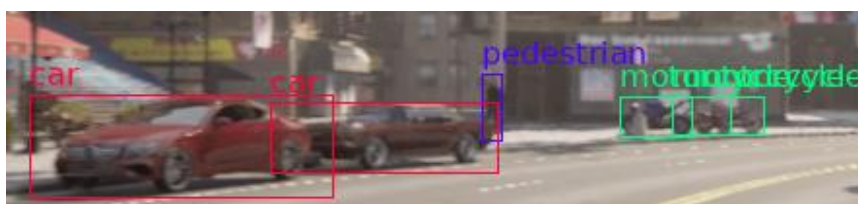
Pomoću izrađenog alata za automatsko označavanje objekata moguće je kreirati raznolike skupove podataka koji mogu imati raznovrsne označene objekte u raznovrsnim vremenskim uvjetima. Na slici 4.1. prikazani su primjeri raznovrsnosti objekata od interesa koji mogu biti označeni dok su na slici 4.2. prikazani su neki od raznovrsnih vremenskih uvjeta u kojima se može snimati podatkovni skup.



a)



b)



c)



d)

**Slika 4.1** Prikaz raznovrsnosti objekata: a) vozila i prometni znak, b) pješak, prometni znak i semafor, c) vozila i pješak i d) vozila i semafor



a)



b)



c)



d)

**Slika 4.2.** Prikaz raznovrsnosti vremenskih uvjeta: a) noć, b) kiša po noći, c) kiša po danu i d) zalazak sunca

Zbog toga što simulacija u CARLA simulatoru nije savršeno usklađena ponekad se pojavljuju pogreške u snimljenom podatkovnom skupu pomoću izgrađenog alata. Na slikama dobivenim iz simulatora dogodi se da neki objekt od interesa, koji treba biti označen, ne bude označen ili bude označen s netočnim graničnim okvirom. Pregledavajući podatkovne skupove pri testiranju i analiziranju alata za automatsko označavanje objekata utvrđeno je da je 7,8% slika pogrešno označeno. Na slici 4.3. prikazani su primjeri nepravilno označenih odnosno ne označenih objekata.



a)



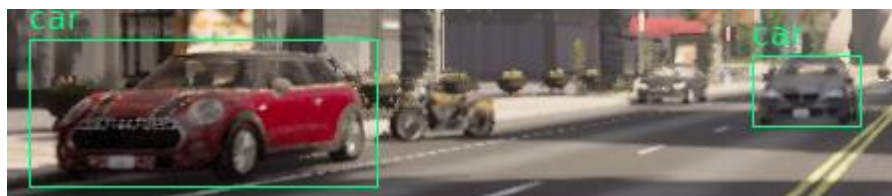
b)



c)



d)



e)

**Slika 4.3.** Prikaz pogreški pri označavanju objekata od interesa: a) nepravilno označeni pješak, b) neoznačeni biciklist, c) označeni bicikli iza grmlja, d) neoznačeni motocikl i e) neoznačeni motocikl

## 5. ZAKLJUČAK

U ovom radu predstavljeni su izazovi prikupljanja i označavanja podataka iz stvarnoga svijeta te kako je dohvaćanje i označavanje podataka iz simulacije autonomne vožnje jeftinije i efikasnije. Opisano je nekoliko postojećih podatkovnih skupova od kojih neki sadrže podatke iz stvarnoga svijeta, a neki podatke iz simulacije autonomne vožnje. Proučeni su postojeći alati za automatsko označavanje objekata na slikama dobivenih iz simulacije autonomne vožnje te je predstavljen vlastiti alat za automatsko označavanje objekata dobivenih iz simulatora autonomne vožnje. Alat koji je napravljen u sklopu rada moguće je koristiti za generiranje vlastitog podatkovnog skupa sa željenim parametrima kao što su vremenski uvjeti, rezolucija slika te gustoća prometa. Podatkovni skup može sadržavati proizvoljnu količinu snimljenih slika te može sadržavati proizvoljne oznake objekata od interesa kao što su semafori, prometni znakovi, vozila i pješaci. Alat je testiran na svih osam mogućih mapa koje sadrži CARLA simulator. Rezultati testiranja alata za automatsko označavanje objekata na slikama dobivenim iz simulacije autonomne vožnje ukazuju na to da konačni podaci ovise o odabranim parametrima simulacije i rute kojom Ego vozilo prolazi po odabranoj mapi, također mogu se dobiti raznoliki podaci te postoji mogućnost pogreške u podacima.

## LITERATURA

- [1] “BBD100k.” <https://www.bdd100k.com/> (accessed Sep. 13, 2022).
- [2] T. L. Berg *et al.*, “It’s All About the Data,” *Proc. IEEE*, vol. 98, no. 8, pp. 1434–1452, Aug. 2010, doi: 10.1109/JPROC.2009.2032355.
- [3] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.
- [4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” *Adv. Neural Inf. Process. Syst.*, vol. 25, 2012.
- [5] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, “Large-scale video classification with convolutional neural networks,” in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2014, pp. 1725–1732.
- [6] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, “Indoor segmentation and support inference from rgb-d images,” in *European conference on computer vision*, 2012, pp. 746–760.
- [7] R. Mottaghi *et al.*, “The role of context for object detection and semantic segmentation in the wild,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 891–898.
- [8] T.-Y. Lin *et al.*, “Microsoft coco: Common objects in context,” in *European conference on computer vision*, 2014, pp. 740–755.
- [9] G. J. Brostow, J. Fauqueur, and R. Cipolla, “Semantic object classes in video: A high-definition ground truth database,” *Pattern Recognit. Lett.*, vol. 30, no. 2, pp. 88–97, 2009.
- [10] T. Scharwächter, M. Enzweiler, U. Franke, and S. Roth, “Efficient multi-cue scene segmentation,” in *German Conference on Pattern Recognition*, 2013, pp. 435–445.
- [11] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, “Vision meets robotics: The kitti dataset,” *Int. J. Robot. Res.*, vol. 32, no. 11, pp. 1231–1237, 2013.
- [12] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman, “LabelMe: a database and web-based tool for image annotation,” *Int. J. Comput. Vis.*, vol. 77, no. 1, pp. 157–173, 2008.
- [13] S. M. Bileschi, “StreetScenes: Towards Scene Understanding in Still Images,” p. 182.
- [14] G. Ros, S. Stent, P. F. Alcantarilla, and T. Watanabe, “Training constrained deconvolutional networks for road scene semantic segmentation,” *ArXiv Prepr. ArXiv160401545*, 2016.
- [15] M. Cordts *et al.*, “The cityscapes dataset,” in *CVPR Workshop on the Future of Datasets in Vision*, 2015, vol. 2.
- [16] G. Neuhold, T. Ollmann, S. Rota Bulò, and P. Kotschieder, “The mapillary vistas dataset for semantic understanding of street scenes,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 4990–4999.
- [17] Klein, Itay, “NEXET — The Largest and Most Diverse Road Dataset in the World.” <https://data.getnexar.com/blog/nexet-the-largest-and-most-diverse-road-dataset-in-the-world/> (accessed Jul. 13, 2022).
- [18] B. Kaneva, A. Torralba, and W. T. Freeman, “Evaluation of image features using a photorealistic virtual world,” in *2011 International conference on computer vision*, 2011, pp. 2282–2289.
- [19] P. P. Busto, J. Liebelt, and J. Gall, “Adaptation of synthetic data for coarse-to-fine viewpoint refinement,” in *BMVC*, 2015, p. 14.1-14.12.
- [20] A. Handa, V. Patraucean, V. Badrinarayanan, S. Stent, and R. Cipolla, “SynthCam3D: Semantic Understanding With Synthetic Indoor Scenes.” arXiv, May 01, 2015. Accessed: Aug. 16, 2022. [Online]. Available: <http://arxiv.org/abs/1505.00171>
- [21] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, “The SYNTHIA Dataset: A Large Collection of Synthetic Images for Semantic Segmentation of Urban Scenes,” in *2016*

- IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 3234–3243. doi: 10.1109/CVPR.2016.352.
- [22] M. Wrenninge and J. Unger, “Synscapes: A photorealistic synthetic dataset for street scene parsing,” *ArXiv Prepr. ArXiv181008705*, 2018.
- [23] S. R. Richter, V. Vineet, S. Roth, and V. Koltun, “Playing for Data: Ground Truth from Computer Games.” *arXiv*, Aug. 07, 2016. Accessed: Aug. 16, 2022. [Online]. Available: <http://arxiv.org/abs/1608.02192>
- [24] A. Gaidon, Q. Wang, Y. Cabon, and E. Vig, “Virtual worlds as proxy for multi-object tracking analysis,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4340–4349.
- [25] T. Sun *et al.*, “SHIFT: A Synthetic Driving Dataset for Continuous Multi-Task Domain Adaptation,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 21371–21382.
- [26] “Cognata | Autonomous and ADAS Vehicles Simulation Software.” <https://www.cognata.com/> (accessed Aug. 18, 2022).
- [27] Y. Scharf, “The spring of self-driving cars.” <https://towardsdatascience.com/the-spring-of-self-driving-cars-e37f76c11586> (accessed Aug. 18, 2022).
- [28] “Anyverse.” <https://anyverse.ai/> (accessed Aug. 18, 2022).
- [29] “CARLA Simulator.” <https://carla.readthedocs.io/en/latest/> (accessed Jul. 06, 2022).
- [30] “Python.” <https://www.python.org/> (accessed Aug. 18, 2022).
- [31] “pascal-voc-writer.” <https://pypi.org/project/pascal-voc-writer/> (accessed Sep. 13, 2022).
- [32] “ImageSet Viewer.” <https://github.com/zchrissirhcz/imageset-viewer> (accessed Sep. 13, 2022).
- [33] “Beginner’s Guide to Python.” <https://wiki.python.org/moin/BeginnersGuide/Download> (accessed Jul. 09, 2022).
- [34] “Quick start package installation.” [https://carla.readthedocs.io/en/latest/start\\_quickstart/](https://carla.readthedocs.io/en/latest/start_quickstart/) (accessed Jul. 09, 2022).



## SAŽETAK

Tema ovog diplomskog rada je izrada alata za automatsko označavanje objekata na slikama dobivenim iz simulatora autonomne vožnje. U radu je objašnjen problem prikupljanja podataka iz stvarnog svijeta te označavanje objekata od interesa na prikupljenim podacima. Objasnjeno je zašto je generiranje podataka pomoću simulacije stvarnoga svijeta jeftinije i efikasnije od prikupljanja i označavanja podataka iz stvarnog svijeta. Dan je pregled postojećih podatkovnih skupova s podacima iz stvarnoga svijeta kao i pregled postojećih podatkovnih skupova s podacima iz simulacije stvarnoga svijeta odnosno sa sintetičkim podacima. Proučeni su postojeći alati za automatsko označavanje objekata na scenama dobivenim iz računalne simulacije te je predstavljeno vlastito rješenje alata za automatsko označavanje objekata. Na kraju su prikazani podaci dobiveni korištenjem vlastitog rješenja te rezultati testiranja i osvrt na rezultate.

### **Ključne riječi:**

označavanje objekata, podatkovni skup, računalna simulacija, sintetički podaci, strojno učenje

## **ABSTRACT**

### **DEVELOPMENT OF A TOOL FOR AUTOMATIC LABELING OF OBJECTS IN IMAGES OBTAINED FROM AN AUTONOMOUS DRIVING SIMULATOR**

The topic of this thesis is the creation of a tool for automatic labeling of objects on images obtained from an autonomous driving simulator. The paper explains the problem of collecting data from the real world and labeling objects of interest on the collected data. It is explained why generating data using real-world simulation is cheaper and more efficient than collecting and labeling real-world data. An overview of existing datasets with data from the real world is given, as well as an overview of existing datasets with data from simulation of the real world, i.e. with synthetic data. The existing tools for automatic marking of objects on scenes obtained from computer simulation were studied, and the own solution of tool for automatic labeling of objects was presented. At the end, the data obtained using own solution are presented, as well as the test results and a review of the results.

#### **Keywords:**

object labeling, dataset, computer simulation, synthetic data, machine learning

## ŽIVOTOPIS

Zvonimir Benčević rođen je 10. travnja 1997. godine u Slavonskom Brodu. Završio je Osnovnu školu Đuro Pilar te Srednju tehničku školu u Slavonskom Brodu, smjer elektrotehnika. Po završetku srednje škole upisuje preddiplomski sveučilišni studij na Fakultetu elektrotehnike, računarstva i informacijskih tehnologija u Osijeku, smjer Elektrotehnika. Pri upisivanju druge godine studija odabire smjer Komunikacije i informatika. 2020. godine uspješno završava preddiplomski studij te upisuje diplomski sveučilišni studij automobilsko računarstvo i komunikacije.

Potpis:

---

## **PRILOZI**

Prilog P.3.1 poveznica na programski kod alata za automatsko označavanje objekata na slikama dobivenim iz simulatora autonomne vožnje:  
[https://github.com/ZBencevic/diplomski/blob/main/alat\\_za\\_automatsko\\_oznacavanje\\_objekata.py](https://github.com/ZBencevic/diplomski/blob/main/alat_za_automatsko_oznacavanje_objekata.py)

Prilog P.3.2 poveznica na Python skriptu za dohvaćanje podataka iz NumPy nizova:  
[https://github.com/ZBencevic/diplomski/blob/main/dataset\\_open.py](https://github.com/ZBencevic/diplomski/blob/main/dataset_open.py)

Prilog P.4.1 poveznica na snimljene podatkovne skupove u svrhu testiranja alata za automatsko označavanje objekata: [dl.rtrk.com/benc](http://dl.rtrk.com/benc)