

Analiza distribucije vozila prema zemlji porijekla na zagrebačkoj obilaznici primjenom metoda video detekcije

Palajsa, Sanja

Undergraduate thesis / Završni rad

2015

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Transport and Traffic Sciences / Sveučilište u Zagrebu, Fakultet prometnih znanosti**

Permanent link / Trajna poveznica: <https://urn.nsk.hr/urn:nbn:hr:119:526625>

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

Download date / Datum preuzimanja: **2025-03-28**



Repository / Repozitorij:

[Faculty of Transport and Traffic Sciences -
Institutional Repository](#)



SVEUČILIŠTE U ZAGREBU
FAKULTET PROMETNIH ZNANOSTI

Sanja Palajsa

**ANALIZA DISTRIBUCIJE VOZILA PREMA
ZEMLJI PORIJEKLA NA ZAGREBAČKOJ
OBILAZNICI PRIMJENOM METODA VIDEO
DETEKCIJE**

ZAVRŠNI RAD

Zagreb, 2015.



Sveučilište u Zagrebu
FAKULTET PROMETNIH ZNANOSTI
Vukelićeva 4, 10000 Zagreb
PREDDIPLOMSKI STUDIJ

Prediplomski studij: ITS i logistika

Zavod: Zavod za inteligentne transportne sustave

Predmet: Računalstvo

ZADATAK ZAVRŠNOG RADA

Pristupnik: Sanja Palajsa

Matični broj: 0135225970

Smjer: Inteligentni transportni sustavi

ZADATAK:

Analiza distribucije vozila prema zemlji porijekla na zagrebačkoj obilaznici primjenom metoda video detekcije

ENGLESKI NAZIV ZADATKA:

Analysis of the vehicle country of origin distribution on the Zagreb bypass using video detection methods

Opis zadatka:

Radi povećanja iskoristivosti cestovne infrastrukture sve se više koriste napredne metode upravljanja iz domene inteligentnih transportnih sustava (ITS). Takve metode podrazumijevaju korištenje povijesnih i stvarno-vremenskih prometnih parametara. Jedan od senzora koji se danas sve više koristi u ITSu je video kamera. Ona omogućuje mjerenje većeg broja prometnih parametara kao i praćenje vozila prepoznavanjem registarskih oznaka. Na osnovu prepoznatih registarskih oznaka moguće je odrediti zemlju porijekla cestovnog vozila te njihovu distribuciju. U radu je potrebno opisati kameru kao senzor za mjerenje prometnih parametara, prikazati arhitekturu sustava za prepoznavanje registarskih oznaka cestovnih vozila, napraviti pregled najčešće korištenih metoda za prepoznavanje registarskih oznaka cestovnih vozila, prilagoditi i testirati postojeću aplikaciju za prepoznavanje registarskih oznaka zasnovanu na biblioteci CARMEN za prepoznavanje registarskih oznaka vozila korištenjem stvarnih video snimki prometa zagrebačke obilaznice.

Zadatak uručen pristupniku:

17. prosinca 2014.

Nadzorni nastavnik:

Djelovođa:

Predsjednik povjerenstva za završni ispit:

Sveučilište u Zagrebu
Fakultet prometnih znanosti

ZAVRŠNI RAD

ANALIZA DISTRIBUCIJE VOZILA PREMA ZEMLJI
PORIJEKLA NA ZAGREBAČKOJ OBILAZNICI PRIMJENOM
METODA VIDEO DETEKCIJE

ANALYSIS OF THE VEHICLE COUNTRY OF ORIGIN
DISTRIBUTION ON THE ZAGREB BYPASS USING VIDEO
DETECTION METHODS

Mentor: doc. dr. sc. Edouard Ivanjko

Student: Sanja Palajsa, 0135225970

Zagreb, 2015.

Zahvala

Zahvaljujem se svome mentoru, docentu Edouardu Ivanjku na pruženoj prilici te na vođenju tijekom izrade ovog rada kao i na suradnji i angažmanu tijekom njegovog mentorstva.

Zatim bih se zahvalila kolegi Kristianu Kovačiću na uloženom vremenu, trudu i pomoći tijekom izrade ovog rada.

Hvala i mojoj obitelji na bezuvjetnoj podršci i strpljenju tijekom mog školovanja.

Sažetak

Radi povećanja iskoristivosti cestovne infrastrukture sve se više koriste napredne metode upravljanja iz domene inteligentnih transportnih sustava (ITS). Takve metode podrazumijevaju korištenje povijesnih i stvarno-vremenskih prometnih parametara. Jedan od senzora, koji se danas sve više koristi u ITSu, je video kamera koja omogućuje mjerenje većeg broja prometnih parametara kao i praćenje vozila prepoznavanjem registarskih oznaka. Na osnovu prepoznatih registarskih oznaka moguće je odrediti zemlju porijekla cestovnog vozila te njihovu distribuciju. U radu je opisana kamera kao senzor za mjerenje prometnih parametara, prikazana arhitektura sustava za prepoznavanje registarskih oznaka cestovnih vozila, napravljen pregled najčešće korištenih metoda za prepoznavanje registarskih oznaka cestovnih vozila, prilagođena i testirana postojeća aplikacija za prepoznavanje registarskih oznaka zasnovana na biblioteci CARMEN za prepoznavanje registarskih oznaka vozila korištenjem stvarnih video snimki prometa zagrebačke obilaznice.

Ključne riječi: Prometni parametri, metode prepoznavanja registarskih oznaka, arhitektura sustava, video kamera.

Abstract

Advanced management methods from the domain of intelligent transport systems (ITS) are being used today to increase the efficiency of road infrastructure. Those methods involve usage of historical and real-time traffic parameters. Video camera is one of the sensors that are used in ITS which allows measurement of several traffic parameters simultaneously, as well as vehicle tracking by recognizing the license plate. On the basis of the identified license plates it is possible to determine the country of origin of the vehicles and their distribution. In this thesis video camera is described as a sensor to measure traffic parameters, system architecture for license plate recognition is shown and described, overview of the most commonly used methods for recognizing license plates is made, the existing applications for license plate recognition based on the CARMEN license plate recognition library using actual video footage of traffic Zagreb bypass is adapted and tested.

Keywords: traffic parameters, recognition license plate methods, system architecture, video camera.

Sadržaj

1. Uvod	1
2. Mjerenje prometnih parametara video kamerom	2
3. Metode prepoznavanja registarskih oznaka cestovnih vozila	6
3.1. Estimacija prvog plana	6
3.2. Klasifikacija	7
3.3. Praćenje	9
3.4. Automatsko prepoznavanje registarskih oznaka	10
3.4.1. Obrada slike	11
3.4.2. Analiza točnosti izvođenja	13
3.5. Mjerenje OD matrica u cestovnom prometu	14
3.5.1. Detekcija vozila na temelju pozadinskog modeliranja	15
3.5.2. Estimacija OD matrica raskrižja	16
4. Arhitektura sustava za prepoznavanje registarskih oznaka	18
4.1. SCOCA	19
4.2. Arhitektura ALPR-a	21
4.3. Arhitektura aplikacije	23
5. Aplikacija za prepoznavanje registarskih oznaka	25
6. Rezultati analize video snimki prometa zagrebačke obilaznice	29
6.1. Rezultati testiranja aplikacije	29
6.2. Analiza i komentari rezultata	30
7. Zaključak	32
Literatura	34
Popis kratica	35
Popis tablica	36
Popis slika	37
Dodatak A Algoritam aplikacije za prepoznavanje registarskih oznaka	38

1. Uvod

Danas se sve više teži optimalnom korištenju postojeće cestovne infrastrukture, a ne samo izgradnji novih kapaciteta kako bi se riješili prometni problemi. U tu se svrhu koriste metode upravljanja prometom iz domene inteligentnih transportnih sustava (ITS). Kako bi se one mogle koristiti potrebno je upravljačkom sustavu dati pristup povijesnim i stvarnovremenskim prometnim parametrima. U tu svrhu se koriste različiti senzori kao što induktivne petlje, radari i sl. Danas se sve više koristi video kamera koja omogućuje istodobno mjerenje više prometnih parametara. Njeno korištenje je omogućio razvoj računala, kamere kao senzora te algoritama za stvarnovremenu obradu slike. Prednosti kamere su da se dobiveni podaci mogu se koristiti za izradu prometne statistike, a mogu se koristiti i da bi se riješio problem nadzora cestovnog prometa. Prednost video kamere nad ostalim sensorima je to što se bavi relativno neograničenom okolinom s vozilima koji idu u svim smjerovima.

U ovom završnom radu je opisana kamera kao senzor za mjerenje prometnih parametara, napravljen pregled najčešće korištenih metoda za prepoznavanje registarskih oznaka cestovnih vozila, prikazana arhitektura sustava za prepoznavanje registarskih oznaka cestovnih vozila, prilagođena i testirana postojeća aplikacija za prepoznavanje registarskih oznaka zasnovana na biblioteci CARMEN za prepoznavanje registarskih oznaka vozila korištenjem stvarnih video snimki prometa zagrebačke obilaznice.

Ovaj rad nakon uvoda u drugom poglavlju opisuje prometne parametre i kako se oni mjere pomoću jedne ili dvije video kamere. Na početku trećeg poglavlja nabrojane su metode za prepoznavanje registarskih oznaka cestovnih vozila. Opisan je i objašnjen način njihova funkcioniranja. U ovom radu fokus će biti na četiri tehnike, a to su tehnika klasifikacije odnosno identifikacija sudionika u prometu (npr. pješaci, automobili, bicikli), estimacija prvog plana (engl. *foreground estimation*), praćenje (engl. *tracking*), te automatsko prepoznavanje registarskih tablica (engl. *automatic license plate recognition*, skraćeno ALPR). ALPR se koristi samo kod specijaliziranih pogleda kamere (vidno polje kamere usmjereno je na registarske tablice i uvećava ih) i ne može dati opažanje širokog područja niti mjerenje interakcija između sudionika u prometu. Najkritičniji dio svakog transportnog sustava su njegovi čvorovi ili raskrižja. Dizajn, propusnost i kapacitet jednog raskrižja može utjecati čak i na udaljenije dijelove prometne mreže. U trećem poglavlju obraditi će se također problem upravljanja raskrižjima i prijedlozi kako te probleme riješiti pomoću estimacije OD matrica. U četvrtom poglavlju je opisana arhitektura sustava za prepoznavanje registarskih oznaka. Peto poglavlje sadrži aplikaciju za prepoznavanje registarskih oznaka. U šestom poglavlju se nalaze rezultati analize video snimki prometa zagrebačke obilaznice. Rad završava sa zaključkom i prijedlozima za budući rad.

2. Mjerenje prometnih parametara video kamerom

Prometni parametri služe za opisivanje prometnih tokova i zakonitosti kretanja motornih vozila u prometnim tokovima na cestovnim prometnicama. Ti parametri se još zovu osnovni parametri prometnog toka ili osnovne veličine prometnog toka. U prometnom toku djeluje međusobna interakcija vozila, što je i osnovna razlika u uvjetima kretanja vozila u prometnim tokovima u odnosu na uvjete kretanja pojedinačnih vozila.

Osnovni parametri prometnog toka su:

- Protok vozila, q ;
- Gustoća prometnog toka, g ;
- Brzina prometnog toka, v ;
- Vrijeme putovanja vozila u toku, t ;
- Jedinično vrijeme putovanja vozila u toku;
- Vremenski interval slijeđenja vozila u toku;
- Razmak slijeđenja vozila u toku, s [1].

Protok vozila je broj vozila koji prođu kroz promatrani presjek prometnice u jedinici vremena. Može se računati za jedan smjer za jednosmjerne prometnice ili u oba smjera za dvosmjerne prometnice. Ovisno o načinu promatranja razlikuje se:

- a) Protok vozila na presjeku ceste predstavlja protok koji se ostvaraje na promatranom presjeku ceste u jedinici vremena;
- b) Protok vozila na dijelu ili dionici ceste predstavlja aritmetičku sredinu protoka na n -presjeka na dijelu ili prometnoj dionici, gdje $n \rightarrow \infty$ [1].

Gustoća prometnog toka podrazumijeva broj vozila na jedinicu duljine prometnice po prometnoj traci po smjerovima (u jednom smjeru za jednosmjerne, odnosno oba smjera za dvosmjerne prometnice), što prikazuje Slika 1.



Slika 1. Gustoća prometnog toka [1]

Pojam gustoće vezan je prostorno za odsjek ili prometnu dionicu, a vremenski za trenutno stanje. S obzirom na vremenski period u kome se promatra, gustoća prometnog toka može predstavljati:

- a) Broj vozila po jedinici dužine promatrane dionice u trenutku promatranja;
- b) Broj vozila po jedinici duljine promatranog dijela dionice kao aritmetička sredina više trenutnih promatranja u nekom vremenskom periodu [1].

Brzina prometnog toka podrazumijeva određenu srednju vrijednost brzine svih vozila koja sudjeluju u promatranom prometnom toku. Postoje dva pojma koja služe za definiranje brzine prometnog toka kao odgovarajuće srednje vrijednosti brzina svih vozila koja čine promatrani prometni tok, a to su:

- a) Srednja prostorna brzina toka, koja je analogno gustoći vezana za odsjek puta, a vremenski za trenutak;
- b) Srednja vremenska brzina toka, koja je analogno protoku vozila prostorno vezana za presjek puta, a vremenski za period promatranja [1].

Srednja prostorna brzina prometnog toka predstavlja aritmetičku sredinu trenutnih brzina svih vozila u prometnom toku na promatranoj dionici puta, dok srednja vremenska brzina prometnog toka predstavlja aritmetičku sredinu brzina svih vozila u prometnom toku koja prolaze kroz promatrani presjek puta u određenom vremenskom periodu.

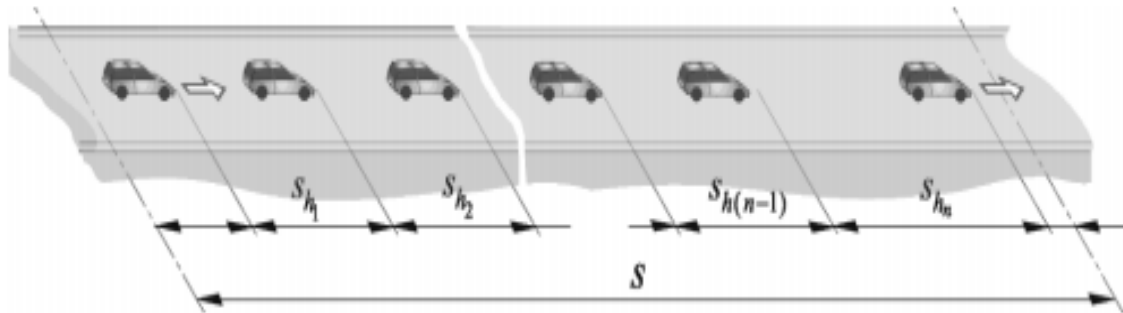
Interval slijeđenja vozila u prometnom toku predstavlja vrijeme između prolaska dva uzastopna vozila kroz zamišljeni presjek promatranog odsjeka puta. Razlikuje se:

- a) Interval praćenja pojedinačno za N vozila koja u periodu vremena T prođu promatranu dionicu puta;
- b) Srednju vrijednost intervala praćenja na promatranom presjeku puta za N vozila u vremenu T ;
- c) Interval slijeđenja na dionici puta, kao aritmetički presjek srednjih vrijednosti intervala praćenja na m promatranih presjeka puta u vremenu T [1].

Interval slijeđenja vozila ima veliku ulogu kod opisivanja uvjeta odvijanja prometa na prometnicama. Opisuje međuovisnost vozila u prometnom toku, kao i kvalitetu prometnog toka. Razmak u slijeđenju vozila podrazumijeva prostorni razmak između dva uzastopna vozila u prometnom toku. Također predstavlja srednju vrijednost svih razmaka praćenja između uzastopnih vozila u određenom toku na promatranoj dionici puta. Razlikuje se:

- a) Udaljenost između pojedinih vozila u prometnom toku koja su se našla u određenom trenutku na promatranoj dionici puta (vidi Sliku 2.);

- b) Srednja vrijednost trenutnih razmaka između svih vozila u prometnom toku koja su se našla u određenom trenutku na promatranom odsjeku ili dionici puta;
- c) Aritmetički prosjek m-srednjih trenutnih razmaka utvrđenih na promatranoj dionici u periodu vremena T [1].



Slika 2. Razmaci u slijeđenju vozila [1]

Svi navedeni prometni parametri mogu se izmjeriti pomoću prometnih informacija dobivenih video nadzorom na prometnicama. Prednost video snimke je u velikom broju dobivenih informacija koje ne samo da služe mjerenju prometnih parametara nego služe i za poboljšanje prometnih uvjeta na prometnicama i za povećanje kvalitete prometnih tokova.

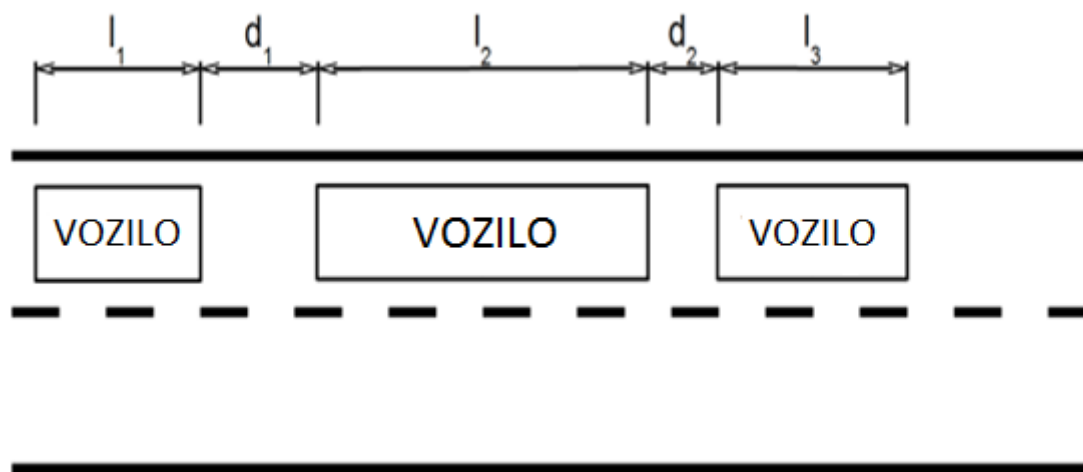
Sustavi za nadzor prometa zasnovanih na video detekciji najčešće upotrebljavaju jednu kameru po prometnoj traci kako bi izmjerili prometne parametre. Takvi sustavi su nedostatak na prometnicama s više traka jer je na takvim prometnicama potreban veliki broj kamera što povećava cijenu sustava.

Za učinkovito praćenje prometne situacije potrebno je mjeriti nekoliko osnovnih parametara koji su potrebni za nadgledanje u upravljanje cestovnom prometnom mrežom, a to su:

- Razmak između vozila;
- Duljina vozila;
- Brzina vozila;
- Putanja vozila [2].

Ovi parametri se trebaju mjeriti i izračunavati s velikom točnošću kako bi se dobili korisni statistički podaci. Iz ovih parametara mogu se dobiti neki složeniji parametri poput protok vozila, prometna struktura što se tiče tipova vozila i cestovno opterećenje. U svrhu procjene ovih podataka sustav treba koristiti značajke poput prosječne brzine svakog vozila koje je detektirano tokom svog putovanja kroz prometnu mrežu. Za dobivanje tih značajki potrebno je mjeriti na svakom čvoru odnosno na mjestima u prometnoj mreži gdje se promet nadzire putem video

kamera. Najčešće su to točke prometne mreže koja označavaju ulaz ili izlaz s autoceste, raskrižje ili naplatne postaje [2].



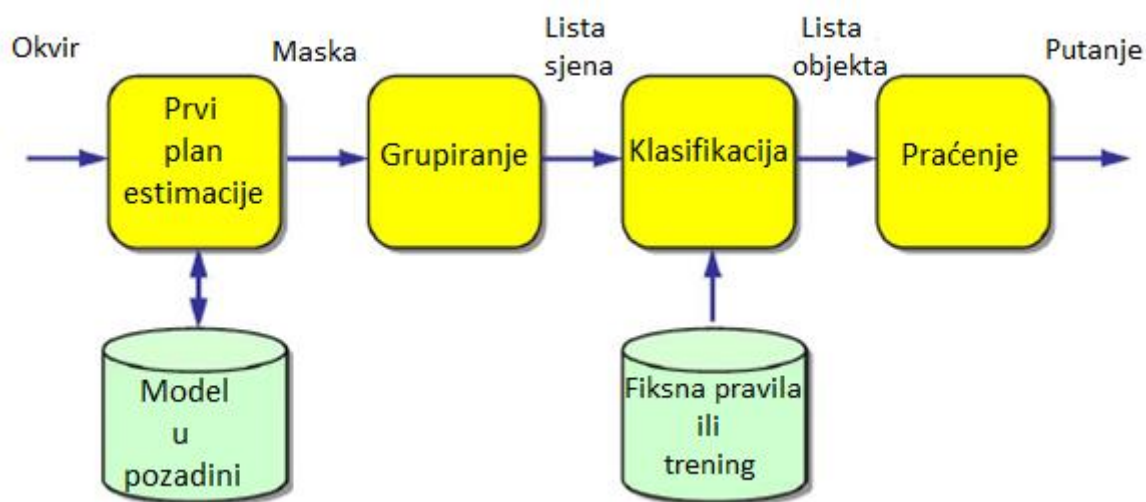
Slika 3. Osnovni prometni parametri [2]

Na Slici 3. su prikazani osnovni parametri, dužina vozila i razmak između vozila. Ovi parametri se mogu mjeriti tehnikama za obradu slika. Za izračun dinamičkih parametara potrebno je znati dodatne parametre poput prosječne brzine vozila i vremena [2].

Korištenje video kamera kao senzor u prometnoj mreži ima puno prednosti. Može se istovremeno mjeriti više prometnih parametara kao što su prometni protok, ishodišno-odredišne (OD) matrice, udaljenosti između vozila, brzine kretanja vozila, klasifikacije tipa vozila (motocikl, osobno / teretno vozilo i dr.), estimacija distribucije zemlje porijekla vozača. Zatim video kamere je jednostavno implementirati u postojeću prometnu infrastrukturu i mali su troškovi u usporedbi sa drugim senzorima. Postoje i neki nedostaci kao što je veliki udio grešaka u izmjeranim prometnim parametrima utjecajem loših vanjskih uvjeta okoline (kiša, magla, vibracija, nagla promjena osvjetljenja i dr.) i potrebno je održavanje (čišćenje stakla objektiva kamere).

3. Metode prepoznavanja registarskih oznaka cestovnih vozila

Potrebni elementi u analizi prometnog sustava su prvi plan estimacije (engl. *foreground estimation*), klasifikacija (engl. *classifier*) i praćenje (engl. *tracking*) (slika 4). Statistički model obično u prvom planu estimacije procjenjuje piksele koji su zatim grupirani u osnovni model i ide kroz sustav do klasifikacije. Klasifikacija koristi prethodne informacije o objektu da bi mu dodijelila oznaku klase.



Slika 4. Blok dijagram za nadzorni sustav [3]

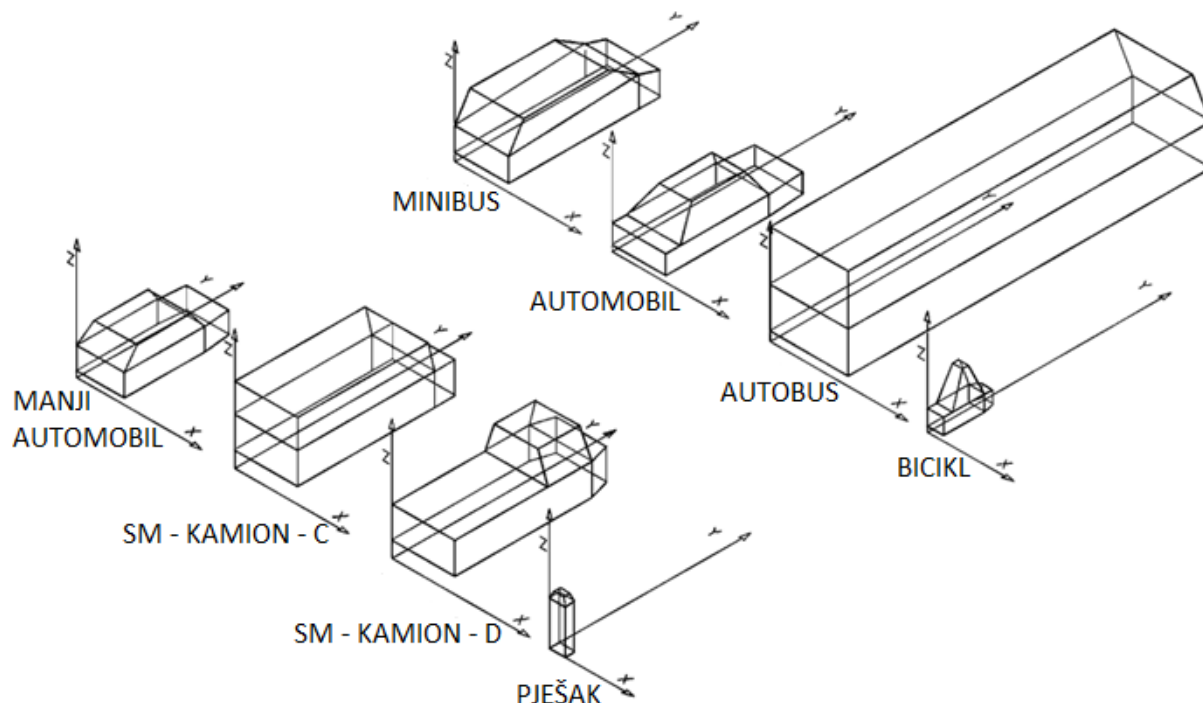
3.1. Estimacija prvog plana

Estimacija prvog plana i segmentacija je prva faza nekoliko vizualnih nadzornih sustava. Prvi plan je definirati svaki objekt koji nije fiksno smješten u prirodi, ali ovu definiciju je teško algoritamski provesti. Postoje dva glavna pristupa procjenjivanja prvog plana, a oba koriste snažne pretpostavke koje su u skladu s navedenom definicijom. Prvi pristup kaže da se u pozadini izabere jedan fiksni model. Taj model bi se uspoređivao s ostalim modelima i identificirale bi se razlike odnosno kretanja, ali pod uvjetom da kamera miruje. Ovaj koncept dovodi do problema kada se promet usporeno kreće ili se uopće ne kreće. Svako vozilo bi se trebalo uzeti u obzir u prvom planu, ali su onda stacionarni objekti neprepoznatljivi zbog nedostatka gibanja. Drugi pristup obavlja segmentaciju na temelju izgleda objekta. Ovaj pristup može se koristiti i kod kretanja i kod stacionarne kamere, ali prethodno je potrebno doći do informacije prvog plana o izgledu objekta [3].

3.2. Klasifikacija

Cilj faze klasifikacije je odrediti kojoj klasi pojedino vozilo pripada. Klasifikator kao ulaz uzima objekt koje zatim deskriptor objekata provjerava i to tako da provjerava poziciju objekta. On mora biti u skladu s gibanjem vozila, a ako nije, objekt je označen kao smetnja i biva odbačen. Objekti mogu biti svrstani u jednu od sljedećih kategorija: pješak, bicikl, motocikl, automobil, kombi, kamion, gradski bus i ekstra-urbani bus. Ako sustav ne može svrstati objekt u nijednu od navedenih kategorija onda ga svrsta u posebnu kategoriju nepoznato.

Ovaj se modul oslanja na skup 3D modela. Svaki od njih pruža grubi opis različitih oblika vozila. Glavni problem klasifikacije je potreba generalizacije predmeta koji pripadaju istoj klasi. Nemoguće je imati sustav koji posjeduje potpuno identični model za svaku pojedinu vrstu vozila. S druge stane, ne može postojati samo jedan model koji bi klasificirao jednu kategoriju kao na primjer automobil. Iz tog razloga, bilo je potrebno uvesti tri različita modela za opisivanje kategorije automobil. Jedan model se koristi za opisivanje kategorija bicikl i motocikl. 3D modeli su prikazani na slici 5 [4].



Slika 5. 3D modeli koji se koriste za prvu razinu klasifikacije [4]

Model pješaka je uveden kako bi se mogle prepoznati osobe u pokretu kako ih sustav ne bi zamijenio s vozilima, najčešće s modelom bicikla. Osnovna ideja je uzeti u obzir projekciju na površinu slike svakog 3D modela koja bi bila stavljena na nadzirana područja na svim mjestima i u svim orijentacijama te bi onda bila uspoređena s konveksnim tijelom objekta izraženim u slikovnim koordinatama. Međutim, ta usporedba je neizvediva u stvarnom vremenu. Kako bi se riješio ovaj problem, broj usporedba se smanjuje u skladu s konfiguracijskim parametrima. Neki put se dogodi da je objekt djelomično vidljiv na slici, ali ova klasifikacijska metoda iskorištava sve dostupne poglede na vozilo i bavi se slučajevima kada vozila nisu u potpunosti prepoznatljiva ili u vidnom polju kamere. Nakon što se svi pogledi na objekt u deskriptoru objekata klasificiraju, računa se ukupni rezultat klasifikacije za svaki 3D model. Rezultat je ponderirani prosjek klasifikacije za svaki od pogleda objekta. Ako je najbolji rezultat ispod danog praga, tada se taj objekt klasificira kao nepoznato. Inače 3D model koji daje povoda za najbolji rezultat uzima se kao rezultat prve razine klasifikacije. Primjer se nalazi na slici 5 [4].

						UKUPNI REZULTAT
BICIKL	0.25	0.17	0.18	0.32	0.62	0.31
MINIBUS	0.74	0.85	0.96	0.95	0.79	0.86
AUTOMOBIL	0.71	0.68	0.69	0.92	0.83	0.77
MANJI AUTOMOBIL	0.75	0.84	0.88	0.93	0.72	0.82
SM - KAMION - D	0.71	0.76	0.84	0.83	0.76	0.78
SM - KAMION - C	0.72	0.76	0.94	0.94	0.77	0.82
AUTOBUS	0.70	0.00	0.00	0.87	0.31	0.38
PJEŠAK	0.02	0.03	0.03	0.08	0.46	0.13

Slika 6. Primjer uzimanja 3D modela koji ima najbolji rezultat [4]

Druga razina klasifikacije je dodjeljivanje klase vozila na objekt, a ne na model. Ako 3D model najbolje odgovara jednoj klasi vozila (vidi Tablicu 1), predmet je jednostavno pridijeljen toj klasi.

Tablica 1. Klase objekata i njihovi pripadajući modeli [4]

Klasa	Model
Bicikl	Bicikl
Motocikl	Bicikl
Automobil	Manji automobil, automobil, minibus
Kombi	SM – kamion – O, SM – kamion – C
Urbani autobus	Autobus
Extra – urbani autobus	Autobus
Kamion	Autobus
Pješak	Pješak

U SCOCA okruženju, samo dva od svih 3D modela odgovaraju u više klasa: CYCLE model, u koji spada bicikl i motocikl, i BUS model u koji spada gradski bus, ekstra-urbani bus i kamion. Sustav izračunava putanju i brzinu svakog otkrivenog i klasificiranog vozila. Podaci dobiveni u konfiguraciji, inicijalizaciji i klasifikacijskoj fazi otkrivanja modela koriste se za obavljanje ovog zadatka. Koristeći podatke iz deskriptora objekta i iz klasifikacijske faze otkrivanja modela može se poboljšati položaj tog modela s obzirom na konveksni trup na svaki pogled na objekt. To nam daje preciznu lokaciju vozila u koordinatnom sustavu stvarnog svijeta svakih ΔT slika. Korištenje točne lokacije modela daje nam prednost implicitno uzimajući u obzir efekt perspektive, na primjer za isti pomak izmjeren na slici, objekti bliže kamere proći će veći pomak nego oni koji se nalaze dalje od kamere [4].

3.3. Praćenje

Praćenje se koristi za mjerenje putanje vozila u video sekvenci. Ovaj pristup se izvodi u dva koraka:

- 1) Značajke za objekt ili značajke područja u prvom planu generiraju se u svakoj slici video snimke;
- 2) Pridruživanje podataka (engl. *data association step*) mora osigurati povezanost između uzastopnih okvira na temelju mogućnosti i dinamičnosti modela.

Vremenske konzistencije ograničenja su potrebne kako bi se izbjegle konfuzije traka i smanjenje zašumljenih mjerenja položaja dobivenih iz izlaza detektora. Pridruživanje podataka može koristiti istu mjeru udaljenosti kao algoritam strojnog učenja. Rezultat klasifikacije i mjesto u slici obično je uključeno u značajku ove udruge. Sljedeći su odjeljci o modelima za praćenje u prometu aplikacije i moguću povezanost podataka na temelju predviđanja.

Aplikacije za praćenje pokretnih objekata u prometu omogućuje fuziju podataka koja se zasniva na predviđanju: Kalman filter, PF, S-T MRF, korespondencija grafa (engl. *graph correspondence*) i stožac događaja (engl. *event cones*) [3].

3.4. Automatsko prepoznavanje registarskih oznaka

ALPR je izumljen 1976. godine na ogranku policijskog znanstvenog razvoja u Velikoj Britaniji. Međutim, stekao je veliko zanimanje tek tijekom posljednjeg desetljeća uz poboljšanje digitalnih fotoaparata i povećanje računalne snage. Ima sposobnost automatski izvući i prepoznati znakove sa registarske oznake s vozila [5].

ALPR je specijalizirana aplikacija za analizu video snimki cestovnog prometa. Kamere daju slike registarskih tablica visoke razlučivosti, ali zato daje minimalno opažanje okoline u kojem se vozilo nalazi. Koristi se kod identifikacije vozila, brojanja vozila, provjera brzine vozila i kod naplatnih kućica. ALPR pojednostavljuje zadatak praćenja vozila zbog toga što svako vozilo ima svoju jedinstvenu registarsku oznaku. ALPR se najčešće koristi kod naplatnih kućica na autocestama na način da registrirani korisnici mogu proći kroz naplatne kućice bez zaustavljanja [3].

Postoji niz mogućih poteškoća s kojima se programska podrška mora biti u stanju nositi, a to su:

- Niska rezolucija slike koja nastaje zbog prevelike udaljenosti vozila od kamere, a ponekad je to rezultat slabe kvalitete kamera koje se koriste;
- Mutne slike koje nastaju zbog toga što su vozila u pokretu odnosno zbog brzine vozila;
- Loša rasvjeta i kontrast zbog sunčeve svjetlosti ili sjena,
- Ne vidi se dio registarske tablice zbog prljavštine ili je odsječen dio pri kreiranju slike iz video snimke;
- Tehnike zaobilaženja;
- Nedostatak koordinacije između država (dva automobila iz različitih država mogu imati iste brojeve na tablici, ali drugačiji dizajn tablice) [6].

Dok se neki od tih problema mogu ispraviti unutar programske podrške, drugi problemu se mogu riješiti u pripadnoj sklopovskoj podršci odnosno montaži video kamere. Povećanje visine kamera mogu se riješiti neki od problema (kao na primjer kada jedno vozilo sakrije registarsku tablicu vozila iza sebe), ali se time povećavaju neki drugi problemi (kao na primjer smanjuje se rezolucija slike zbog udaljenosti između kamera i vozila).

Neki manji sustavi dopuštaju pogreške kod registarskih tablica. Može se odlučiti da sustav ima prihvatljivu stopu od jednog pogrešnog znaka zbog male

vjerojatnosti da neki drugi automobil ima skoro identičnu tablicu. Međutim, ta stopa prihvatljivosti nije dopuštena u ANPR sustavima.

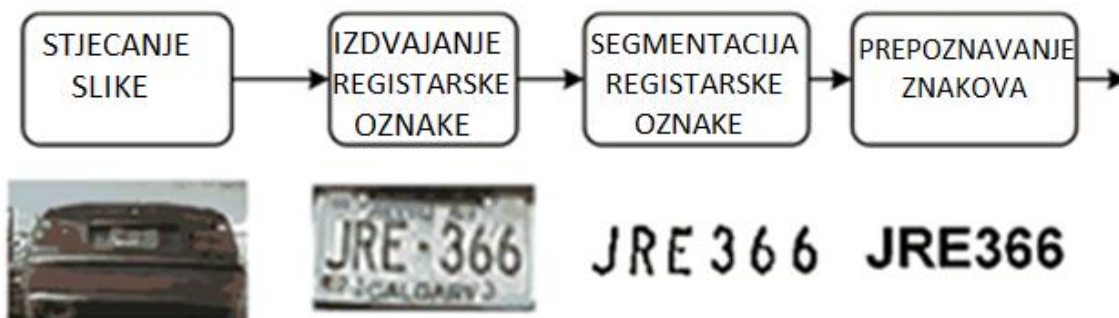
3.4.1. Obrada slike

Obrada slike je vađenje bitnih i korisnih informacija iz digitalnih slika pomoću digitalnih tehnika obrade slike (skraćeno DIP, engl. *Digital Image Processing*). DIP je oblik odrade signala koji kao ulaznu jedinicu uzima digitalnu sliku ili video snimak. Izlazna jedinica može biti slika ili skup korisnih značajki koje se koriste za rješavanje problema u aplikaciji. Ulazna jedinica koristi sliku u boji ili u sivim tonovima, dok je izlaz niz znakova [7].

ALPR obično obuhvaća tri koraka:

- Detekciju registarske oznake;
- Segmentacija znakova;
- Optičko prepoznavanje znakova [7].

U sva tri koraka ima puno prostora za implementaciju analize slike i sheme umjetne inteligencije. Prikazano na Slici 7.



Slika 7. Koraci ALPR-a [7]

3.4.1.1. Detekcija registracijske oznake

Detekcija registracijske oznake teži prostornoj identifikaciji registracijske oznake unutar ulazne slike. Proces dobivanja informacija inicira smanjenje količine podataka sa slike. Što se tiče izdvajanja zemlje porijekla s registracijske oznake,

koriste se tehnike koje se temelje na kombinaciji statistike, teksture i matematičke morfologije.

Izuzetno bitan korak u detekciji registracijske oznake je analiza povezanih komponenti (skraćeno CCA, engl. *Connected Component Analysis*). CCA je vitalna tehnika u binarnoj obradi slike koja skenira sliku i pretvara piksele u komponente na temelju povezanosti piksela [7].

Nakon što su određene sve skupine piksela, svaki piksel je označen s vrijednošću po komponenti kojoj je dodijeljen. Ekstrakcija i označavanje raznih razdvojenih i spojenih komponenti na slici je glavni dio mnogih automatiziranih aplikacija analize slike jer se pomoću njih mogu dobiti korisne informacije i mjerenja binarnih objekata. Neke od informacija su područje i orijentacija, prikazano na Slici 8. Zatim korištenjem jednostavnih tehnika filtriranja, binarni objekti koji prelaze željene granice mogu se ukloniti u algoritamskim koracima koji slijede [7].



Slika 8. Oblik analize u binarnom objektu[13]

3.4.1.2. Segmentacija znakova

Dobivene slike registracijskih oznaka ulaze u fazu segmentacije znakova. Postoji puno tehnika koje se mogu koristiti pri segmentaciji. Najčešća i najjednostavnija tehnika je metoda koja pregledava vertikalne i horizontalne projekcije piksela. Projekcije se koriste za segmentaciju u mnogim sustavima za prepoznavanja teksta. Bitno je naglasiti da se za puno modula za segmentaciju znakova koristi više metoda (na primjer, CCA ili filtriranje objekta praćen metodama projekcije). Postoji i faza u kojoj se briše zvuk, povećava slika i detektira i centralizira slika registracijske oznake. Izlazna jedinica je slika koja predstavlja ispravljenju i povećanu registracijsku oznaku [7].

Segmentacija slike je potrebna da bi se moglo provesti prepoznavanje znakova koja se potpuno oslanja na izoliranim znakovima. Pogrešno segmentirani znakovi u većini slučajeva neće biti uspješno prepoznati. Činjenica je da je većina pogrešno prepoznatih znakova posljedica pogrešaka kod segmentacije [7].

3.4.1.3. Optičko prepoznavanje znakova

Za prepoznavanje izdvojenih znakova putem segmentacije koriste se mnogobrojni algoritmi koji se pojavljuju u aplikacijama za optičko prepoznavanje znakova. Kod prepoznavanja znakova stvaraju problem prljavština, oštećenja i nepredvidive sjene na registracijskim oznakama.

Optičko prepoznavanje znakova (skraćeno OCR, engl. *optical character recognition*) bi trebalo uspješno savladati nejasnoće koje bi se mogle pojaviti kod pogrešno segmentiranih znakova iz prijašnje faze segmentacije znakova. Dobiveni su vrlo dobri rezultati korištenjem umjetnih neuronskih mreža (engl. *artificial neural network*) i statističkih klasifikatora. Motori za optičko prepoznavanje znakova (engl. *optical character recognition*, skraćeno OCR) su danas vrlo pouzdani i s vremenom su ažurirani pa programeri svoju pozornost usredotočuju na poboljšanja unutar OCR koji se odnose na nejasnoće kod znakova, nego na redizajn ili prekvalifikaciju modula za prepoznavanje znakova [7].

3.4.2. Analiza točnosti izvođenja

Zbog širokog raspona primjene ALPR sustava ne postoji zajednički konsenzus o tome kako programeri ili krajnji korisnici mogu ocijeniti ukupnu stopu učinkovitost ovog sustava, ali najčešće se provode tri metode. Prva metoda za mjerenje ukupne učinkovitosti je izračunavanje postotka ispravno detektiranih registracijskih oznaka i ispravno prepoznatih znakova. Druga metoda procjenjuje ukupnu učinkovitost putem formule koja povezuje ukupnu stopu prepoznavanja znakova s očekivanim brojem znakova. Treća metoda, koja je najtočnija, koja kaže da bi se u obzir trebalo uzeti samo postotak točno interpretiranih znakova sa cijele registracijske oznake.

Najbitniji dio procjene učinkovitosti ALPR sustava su uvjeti snimke u kojima se nalazila registracijska oznaka kada je sustav detektirao registarsku oznaku. Za utvrđivanje točnosti sustava neophodno je uzeti u obzir pokazatelje neuspješno prepoznatih oznaka i razna ograničenja koja su se dogodila tijekom tog procesa [7].

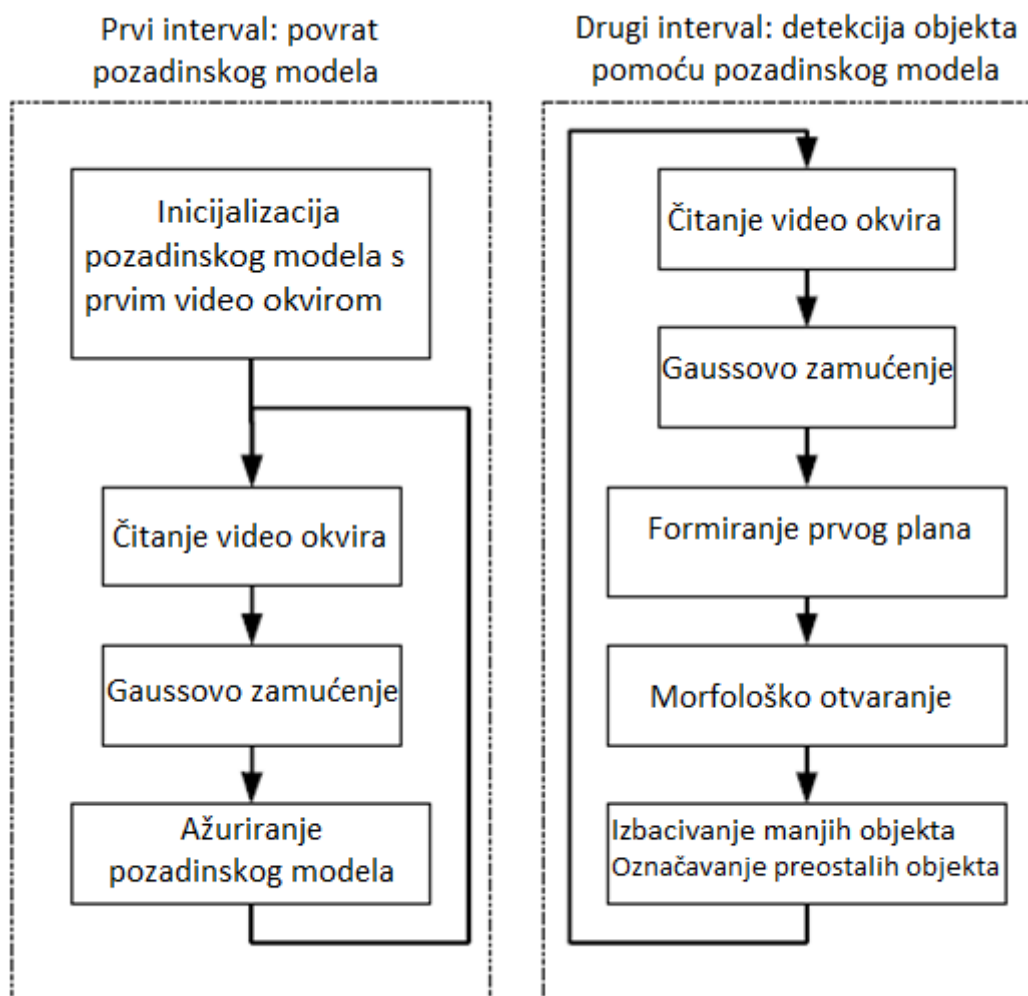
3.5. Mjerenje OD matrica u cestovnom prometu

OD (engl. *origin-destination*) matrice sadrže broj vozila koja ulaze i broj vozila koja izlaze iz raskrižja ili zatvorene prometne mreže u promatranom vremenskom intervalu. Mjerenje OD matrica je posebno izazovno na složenim križanjima i međusobno povezanih čvorištima gdje je brojanje vozila vrlo nepraktično i skupo za ljudske operatore i senzore. Stoga postoji snažna potreba za bolju senzorsku tehnologiju koje će biti u mogućnosti izračunati broj vozila na složenijim raskrižjima.

Da bi sustav detektirao vozilo potrebno je odvojiti fiksnu okolinu od pokretnih objekata, odnosno pozadinske objekte od objekata u prvom planu. Mogao bi se kreirati model kojem bi baš to bio zadatak. Sustav za otkrivanje objekta na temelju takvog modela bi trebao uključiti sljedeće korake za obradu:

- Stvaranje pozadinskog modela željenu okolinu;
- Uspoređujući trenutni video okvir sa pozadinskim modelom da bi se identificirali pikseli koji su u prvom planu;
- Grupiranje piksela u prvom planu u objekte visoke razine.

Pozadinski model je često građen i ažuriran istodobno s procesom detektiranja objekta. Glavna prednost takvog pristupa je mogućnost da adaptivni model može tolerirati male promjene zbog različitih doba dana. Međutim, taj lik nam nije potreban jer se parametri prometa procjenjuju u ograničenim vremenskim razmacima i to obično u vrijeme vršnog prometa. Zato se gradi pozadinski model. Detekcija se izvodi na jednoj video snimci prometa nakon čega se vraća na početak, kao što je to prikazano na Slici 9. [8].



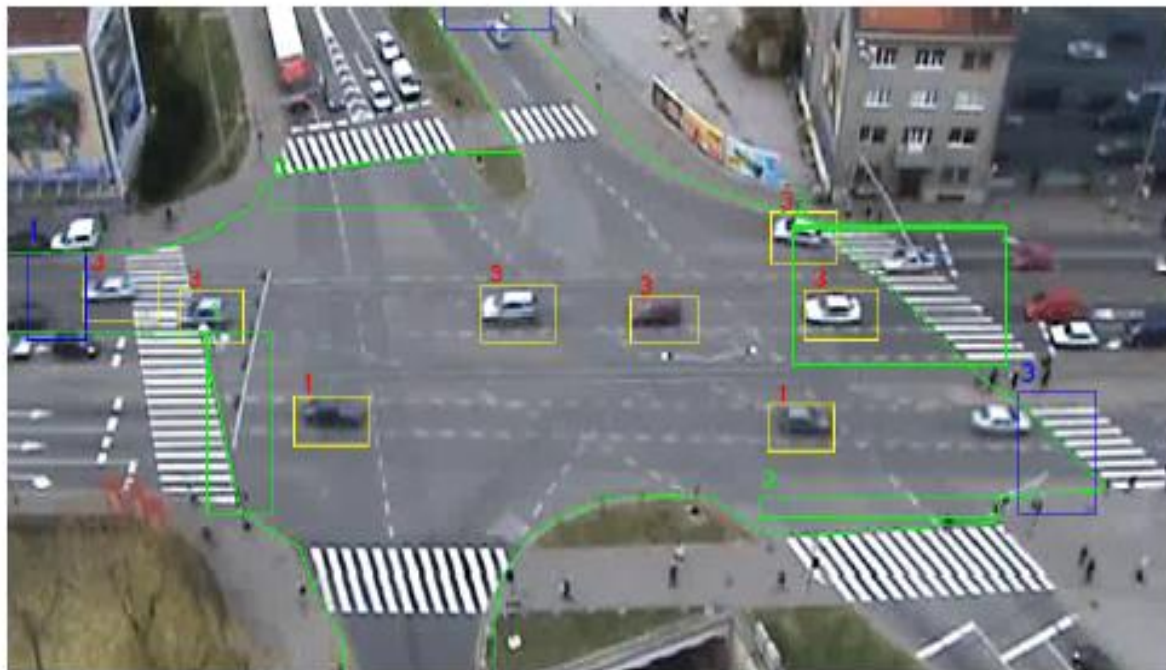
Slika 9. Dijagram predloženog postupka [8]

3.5.1. Detekcija vozila na temelju pozadinskog modeliranja

Izrada pozadinskog modela za računanje prosjeka pomaka koristi manje memorije, ali zahtjeva više vremena za računanje. Postoji i pozadinski modeli temeljeni na histogramu koji su nešto brži, ali zahtijevaju više memorije. Oba pristupa imaju unaprijed izračunati pristup i mogu se koristiti u stvarnom vremenu. Kod unaprijed izračunatog pristupa se izbjegava čekanje na konvergenciju modela i sustav je usredotočen na prepoznavanje objekata, a ne na pozadinsko modeliranje. Glavni nedostatak unaprijed izračunatog pristupa je nesposobnost prilagodbe modela na promjene. Sustav ponekad pogriješi pa susjedna vozila detektira kao jedno. Osim toga, neka su vozila detektira kao dva objekta. To se može dogoditi kada je automobil sličan pozadini ili zbog odraza sivog nebo na vjetrobranskom staklu. Najveći problem je ulaz tramvaja u promatranu okolinu. Sustav otkrije veliki objekt koji uključuje tramvaj, ali i sva vozila koja se kreću u njegovoj blizini [8].

3.5.2. Estimacija OD matrica raskrižja

Ovaj sustav omogućuje korisniku da definira regije koje predstavljaju ulaze i izlaze u raskrižju i za brojanje objekata koja prolaze ulazom i izlazom. OD matrica je prikazana kao sloj u videu i spremjena je u datoteku za daljnju analizu. Slika 8 prikazuje raskrižje s označenim ulazima (zelena) i izlazima (plava).



Slika 10. Rezultati detekcije objekata [8]

Svi objekti koji se nalaze izvan poligona se odbacuju. To se moralo učiniti zbog refleksija na prozorima od zgrada i zbog grupa pješaka koji mogu biti dovoljno velike da se detektiraju kao vozilo. U obzir se ne uzimaju slike u kojima se nalaze tramvaji zbog toga što sustav detektira tramvaj i sva vozila koja se nalaze blizu njega kao jedno vozilo [8].

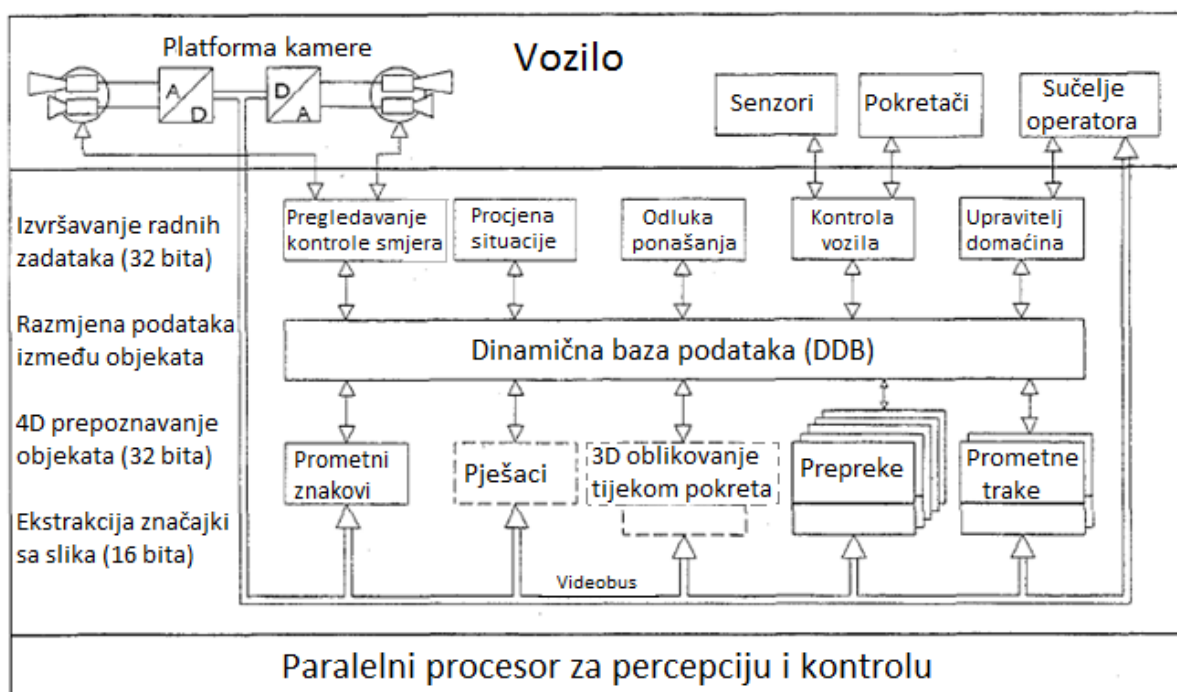
Tablica 2. Usporedba estimiranih OD matrica sa mjerenjima [8]

	Dobivena mjerenja			
Porijeklo	0	1	2	3
0	0:0	9:9	15:17	42:44
1	14:14	0:0	8:8	24:17
2	22:27	20:23	1:1	0:0
3	28:30	32:32	26:24	3:3

Dobiveni rezultati su sažeti u Tablici 2. Unosi u tablicama pokazuju broj estimiranih vozila u odnosu na broj vozila dobiven stvarnim mjerenjem (engl. *groundtruth*).

4. Arhitektura sustava za prepoznavanje registarskih oznaka

U posljednjih nekoliko godina došlo je do povećanja mogućnosti za automatsku analizu prometne aktivnosti. Poboljšana je mjerna infrastruktura postavljanjem kamera i drugih senzora što potrebne podatke čini pristupačnijima. Na slici 1 je prikazana ukupna arhitektura sustava. Kamere su postavljene u svrhu praćenja prometa jer pružaju bogat izvor informacija, a video analiza sada može pružiti dodatnu vrijednost kamere tako da automatski vadi relevantne informacije. Na taj način video analiza postaje sve važnija za ITS.

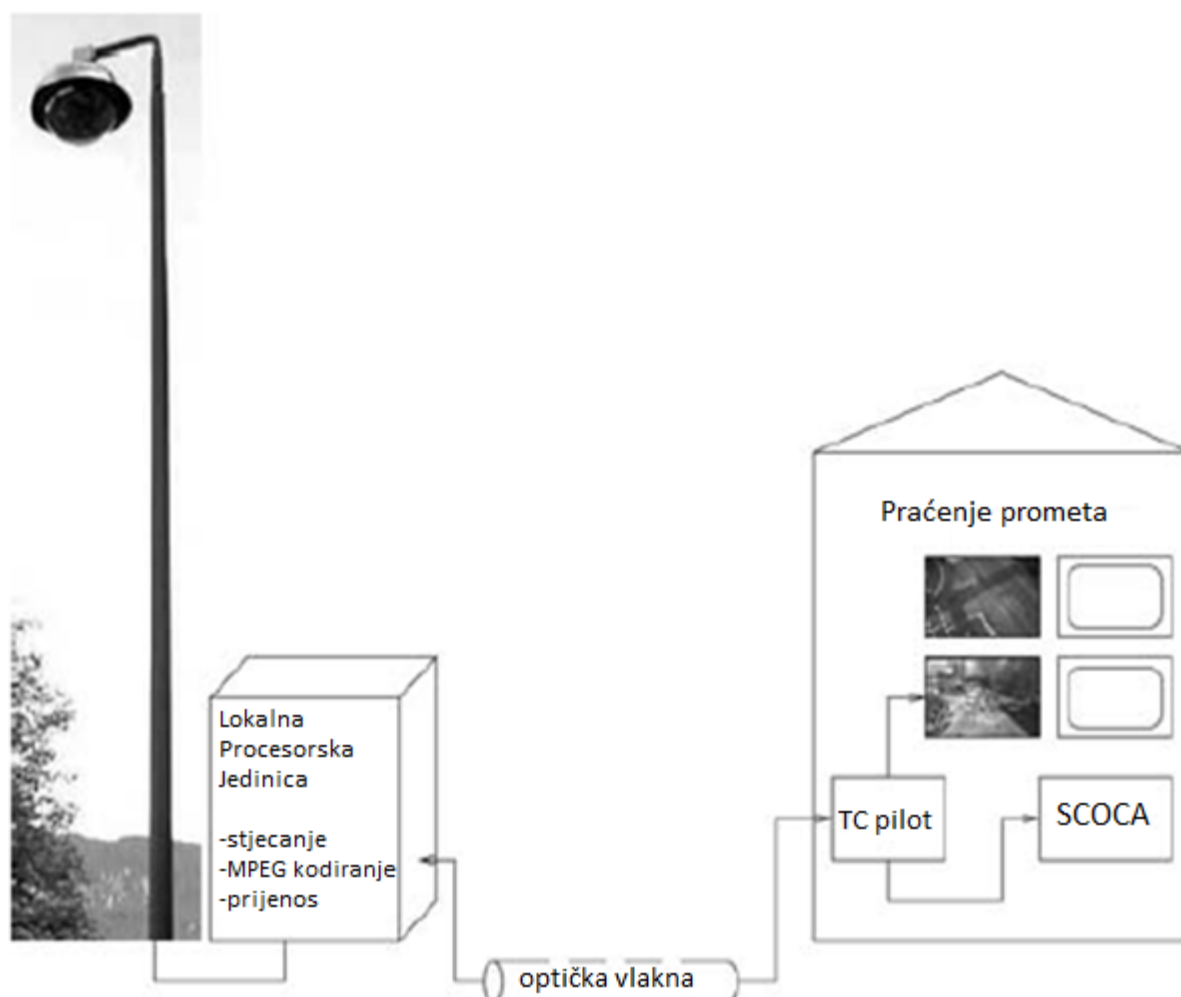


Slika 11. Ukupna arhitektura sustava [9]

Danas su nadzorne videokamere instalirane na mnogim mjestima. One koje su montirane uz autoceste prate trenutačnu prometnu situaciju koju šalju prometnim kontrolorima. U urbanim sredinama kamere se prvenstveno nalaze na velikim uličnim raskrižjima. Cilj korištenja kamera je automatizirati zadatke kao što su otkrivanje potencijalno opasnih situacija (na primjer ako se vozilo zaustavi u tunelu), prepoznavanja registarskih tablica i brzine vozila u slučaju kršenja zakona, odnosno mjerenje statističkih podataka o prometu.

4.1. SCOCA

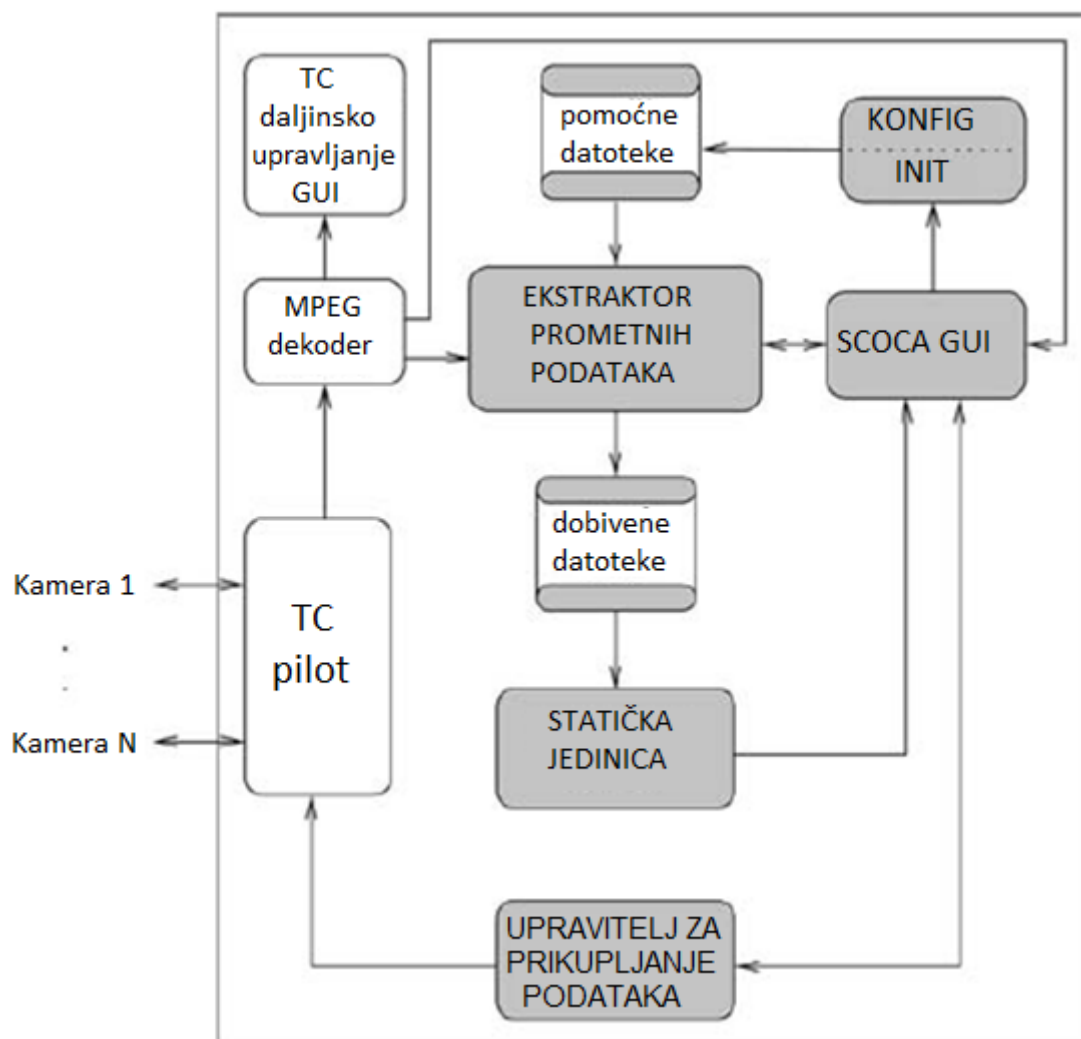
Jedan od sustava koji služi za automatsko brojanje i razvrstavanje vozila je SCOCA (engl. *System for COunting and Classifying Automatically vehicles*). Njegov cilj je prikupiti podatke o prometu na raskrižjima u statističke svrhe. Kamere se montiraju na stupove koji se nalaze pored cesta u gradskom raskrižju. Svaka kamera (Slika 12) je spojena na lokalnu procesorsku jedinicu koja snima slike i kodira video za prijenos u stvarnom vremenu u MPEG2 formatu. Lokalne jedinice povezane su putem optičke mreže na središnji procesor TMC, gdje se komprimirani tokovi mogu biti primljeni, dekomprimirani i vizualizirani na monitorima za potrebe daljinskog upravljanja. SCOCA se nalazi na središnjem procesoru. Sustav je u stanju otkriti vozila dok se kreću, a kako bi ih pratili i klasificirali svaki se pojedini objekt u realnom vremenu razvrstava u jednu od osam kategorija: pješak, bicikl, motocikl, automobil, kombi, kamion, gradski bus i ekstra-urbani bus [4].



Slika 12. Prošireni sustav za praćenje prometa [4]

Struktura SCOCA ilustriran je na dijagramu prikazanog na slici 13 gdje su SCOCA moduli:

- grafičko sučelje SCOCA GUI;
- off-line rad modula KONFIG, INIT;
- upravitelj za prikupljanje podataka;
- ekstraktor prometnih podataka;
- statička jedinica koja pruža sažet opis prikupljenih podataka [4].



Slika 13. Arhitektura SCOCA [4]

4.2. Arhitektura ALPR-a

Klasični ALPR sastoji se od sljedećih komponenata:

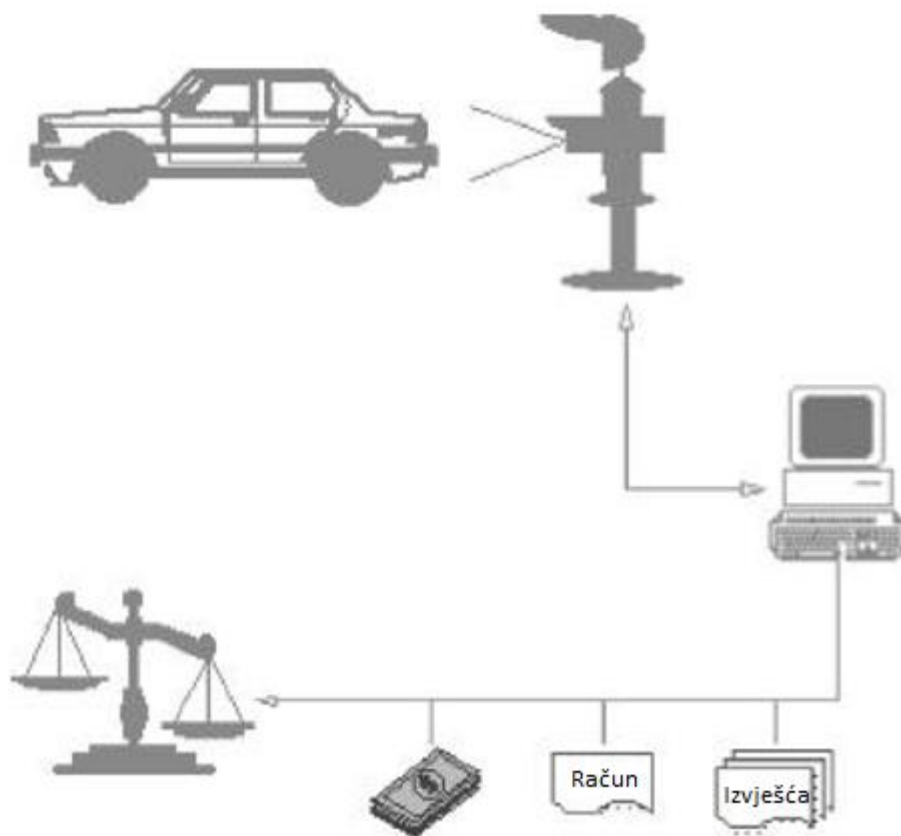
- Kamera – uzima slike vozila s prednje i stražnje strane;
- Osvjetljenje – kontrolirano svjetlo koje osvjetljava registarsku oznaku i može se koristiti tijekom dana te tijekom noći. U većini slučajeva osvjetljenje je infracrveno koje nije vidljivo vozaču;
- Video kartica (engl. *frame grabber*) – sučelje između kamere i računala koje dozvoljava programu čitanje informacija iz kamere;
- Računalo – najčešće računalo s Windows ili Linux sustavom. Pokreće ALPR aplikaciju koja kontrolira sustav, čita slike, analizira i prepoznaje registarske oznake i komunicira s drugim aplikacijama i sustavima.
- Programska podrška - Najčešće paket za prepoznavanje se prikazuje kao DLL (engl. *dynamic link library*);
- Hardver – sklopovska podrška (na primjer kontrolne i mrežne kartice);
- Baza podataka – podaci se zapisuju u lokalnoj bazi podataka ili se odašilju kroz mrežu. Podaci sadržavaju rezultate prepoznavanja i, ako je potrebno, vozilo ili sliku lica vozača [10].



Slika 14. Tipična konfiguracija ALPR sustava [10]

Na Slici 14. je prikazana tipična konfiguracija ALPR sustava (na primjer kontrolni sustav za dvije ulazne i dvije izlazne trake). Sustav SeeLane je tipični primjer takvog sustava. SeeLane aplikacija se pokreće kao pozadinska Windows aplikacija u računalu (prikazano u sredini slike) te sučelje za skup kamera i jedinica za osvjetljenje (jedna za svako vozilo) koje su povezane s video karticom. Aplikacija kontrolira senzore i kontrole preko I/O kartice koja je povezana preko terminala za ulazne i izlazne jedinice. Aplikacija prikazuje rezultate i može ih poslati preko serijske komunikacije i putem DDE poruka drugim aplikacijama. Zapisuje informacije u lokalnoj bazi podataka ili u nekoj drugoj izabranoj bazi podataka (putem mreže) [10].

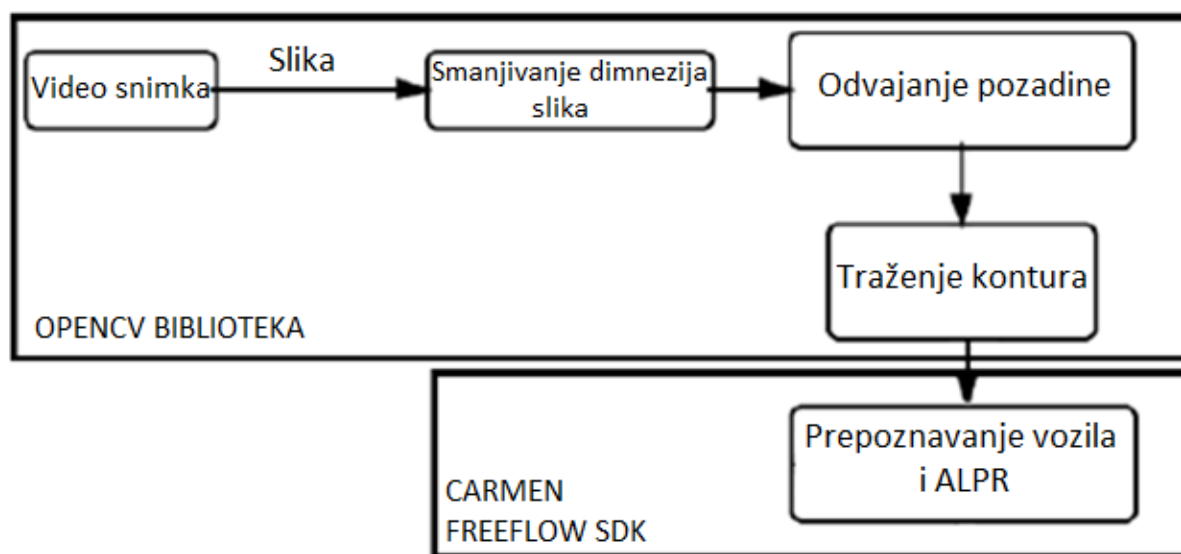
Planiranje prijevoza i razvoj infrastrukture su veliki problemi u proširenju globalne mreže. Tehnološka dostignuća su omogućila nove metode skupljanja podataka iz prometa i mogu se koristiti za planiranje i održavanje učinkovite cestovne politike. ALPR tehnologije počinju biti prihvaćene zbog širokog spektra aplikacija pomoću kojih mogu biti primijenjene. Na Slici 15. se nalaze komponente ALPR-a: osvjetljenje, kamera, senzor, stroj za obradu slika, izvor napajanja i računalu [11].



Slika 15. Komponente ALPR sustava [12]

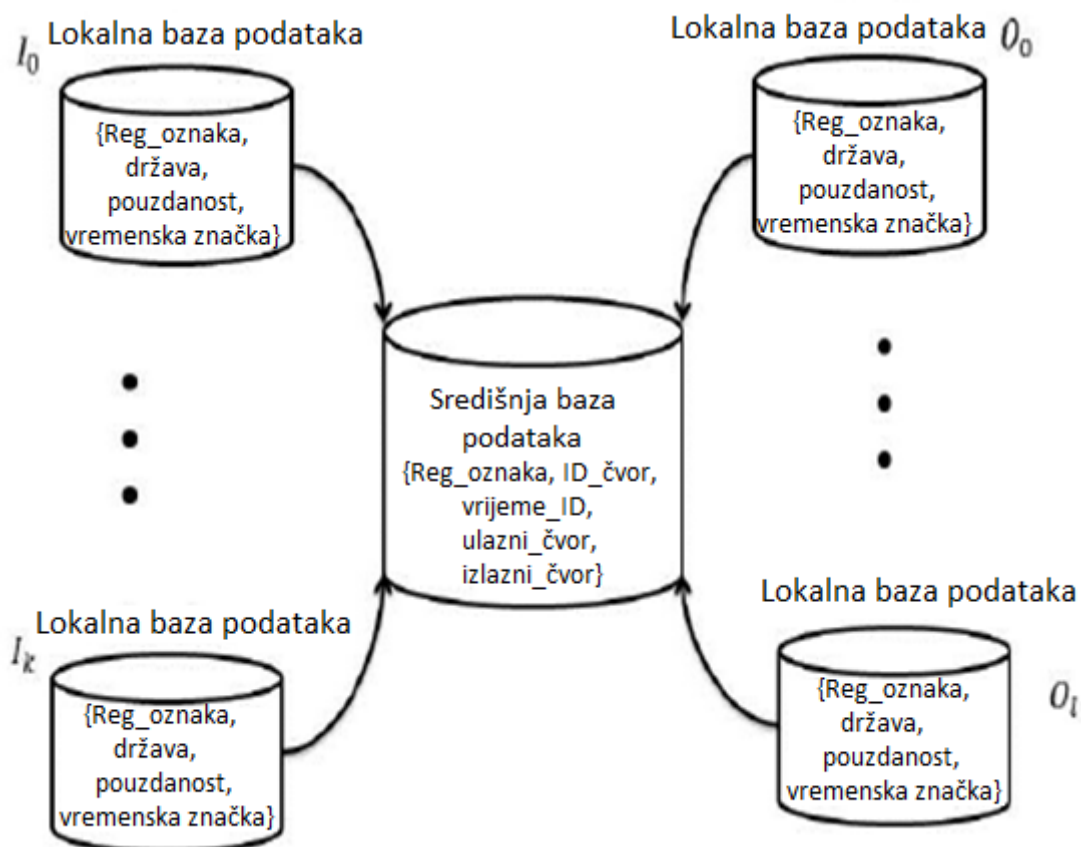
4.3. Arhitektura aplikacije

Aplikacija sustava korištena u ovom završnom radu se sastoji od dva dijela. Prvi dio obrađuje sliku što se izvršava u svakom pojedinom čvoru. Slika 16. prikazuje strukturu obrade slike kako bi se otkrilo vozilo i kako bi se izvukli podaci s registarske oznake. Iz video zapisa uzima se slika u HD rezoluciji. Kako bi se ubrzalo izvođenje aplikacije, slika se smanjuje za određeni koeficijent (npr. $0,5 = 50\%$). Na obrađenoj se slici dalje izvršava detekcija vozila pomoću metode odvajanja pozadine (engl. *background subtraction method*) [2].



Slika 16. Arhitektura aplikacije [2]

Rezultat metode odvajanja pozadine se proslijeđuje algoritmu za pronalazak kontura na slici ('find contours' blok). Izlazni parametri bloka za traženje kontura su lokacija i veličina pravokutnog okvira unutar kojega je detektiran pokretni objekt odnosno vozilo.

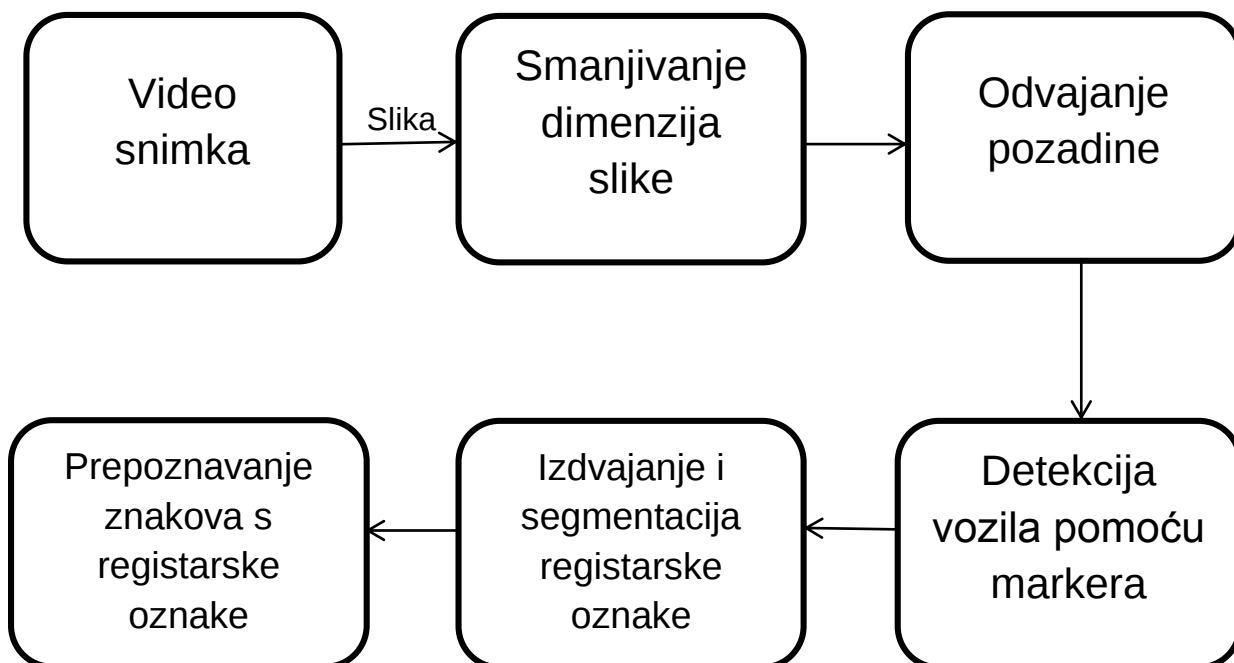


Slika 17. Arhitektura autoceste sa stajališta baze podataka [2]

Drugi dio aplikacije koji provodi automatsko prepoznavanje registarske oznake vozila koristi sliku u HD rezoluciji. ALPR se ne provodi na cijeloj slici već samo na dijelu slike (okviru) unutar kojeg je detektirano vozilo. Dobiveni podaci vozila su pohranjeni u lokalnoj bazi podataka koja je postavljena za svaki čvor. Paralelno s tim, podaci koji se odnose na registarsku oznaku, ID broj čvora i vremena kada je vozilo registrirano, pohranjuje se u središnjoj bazi podataka što je prikazano na Slici 17. Čvor se može razlikovati između ulaznih i izlaznih čvorova na autocesti prometne mreže [2].

5. Aplikacija za prepoznavanje registarskih oznaka

Slika 18. prikazuje blokovsku shemu rada aplikacije.



Slika 18. Blokovska shema rada aplikacije

U aplikaciji za prepoznavanje registarskih oznaka prvo je potrebno odrediti parametre rada aplikacije. Nakon što je unesena putanja koja vodi do video datoteke, potrebno je smanjiti sliku iz video datoteke da bi aplikacija radila brže. Slika se smanjuje po x i y osi. Parametar koji to označuje izražava se u postocima (na primer 0,25 = 25% smanjiti sliku po x ili y osi). Zatim je potrebno odrediti početak i kraj video obrade koji se izražava u milisekundama.

```
CHAR ime_video_datoteke[] = "C:\\Lucko_1_mpeg4.mp4";  
  
double umanji_sliku_po_x = 0.225;  
double umanji_sliku_po_y = 0.225;  
  
double pocni_video = 10500.0;  
double zavrsi_video = 180000.0;
```

Nakon što se aplikacija pokrene, na konzoli se otvore dva prozora koji prikazuju slike videa i slike detektora.

```
namedWindow("Video");  
namedWindow("Detektor");
```

Nadalje je potrebno napraviti distribuciju po zemljama, klasu za odvajanje pozadine, klasu za čitanje video zapisa, matrice unutar kojih će se spremati slike i markere pomoću kojih se detektiraju objekti.

```
vector<distribucija> dist;

BackgroundSubtractorMOG2 detektor(100, 15, true);

VideoCapture video(ime_video_datoteke);

Mat slika_velika, slika_umanjena, slika_detektor;

video.set(CV_CAP_PROP_POS_MSEC, pocni_video);

vector<Marker> markeri;
izradi_markere(markeri, umanji_sliku_po_x, umanji_sliku_po_y);
```

Zatim je potrebno inicijalizirati klasu za prepoznavanje registarskih oznaka vozila. Treba se napraviti datoteka unutar koje će se spremati ispis registarskih oznaka, zemlje porijekla, pouzdanost sustava da je točno prepoznao znakove na registarskoj oznaci i vrijeme u videu kada je marker detektirao registarsku oznaku.

```
LPR prep_reg_oz;

ofstream podaci;
podaci.open("listaRegistarskihOznaka.txt", ios_base::ate);
podaci << "Ispis registarskih oznaka, zemlje porijekla, pouzdanost sustava da je  
točno prepoznao znakove na registarskoj oznaci i vrijeme u videu kada je marker  
detektirao registarsku oznaku." << "\n" << "\n" << "Redni broj" << "\t" <<  
"Registarska oznaka" << "\t" << "Drzava" << "\t" << "\t" << "Pouzdanost" << "\t"  
<< "Vrijeme" << "\t" << "\t" << "Broj markera" << "\n";
podaci.close();
```

Zatim je potrebno napraviti detektore koji će detektirati vozilo i markere koji se nalaze na ekranu. Markeri koji se nalaze na ekranu nisu fiksni. Njihova lokacija i veličina se mogu mijenjati. Možemo odabrati lokaciju gdje marker počinje po x i y osi, a veličina se regulira po visini i širini.

```
detektor(slika_umanjena, slika_detektor);
provjeri_markere(markeri, slika_detektor);

for (size_t i = 0; i < markeri.size(); i++)
{
    if (markeri[i].VoziloDetektirano == true &&
        markeri[i].RegOznakaProcitana == false)
    {
        prep_reg_oz.Prepoznaj(slika_velika,
                               markeri[i].PodrucjeVeliko);

        cmNP *rezultat = prep_reg_oz.Prepoznaj(slika_velika,
                                                markeri[i].PodrucjeVeliko);

        if (rezultat)
        {
```

```

        CHAR reg_oznaka[4];

        prep_reg_oz.PrevediRegOznaku(rezultat->type, reg_oznaka);

        Mat slika_vozila =
        slika_velika(markeri[i].PodrucjeVeliko);

        markeri[i].RegOznakaProcitana =
        detektirano_vozilo(rezultat->text, reg_oznaka, rezultat-
        >confidence, slika_vozila, vrijeme, i, dist,
        redniBrojVozila);
    }
}

nacrtaj_markere(markeri, slika_umanjena);

imshow("Video", slika_umanjena);
imshow("Detektor", slika_detektor);
waitKey(1);
}

```

Kao što smo napravili datoteku u koju će se spremati ispis registarskih vozila, potrebno je napraviti i datoteku u koju će se spremati ispis liste država, broj vozila detektiranih iz određene države i postotak tog broja naspram ukupnog broja vozila.

```

ofstream datoteka;
datoteka.open("listaDrzava.txt", ios_base::ate);
datoteka << "Ispis liste drzava, broj vozila detektiranih iz odredene drzave i
postotak tog broja naspram ukupnog broja vozila." << "\n" << "\n" << "Redni
broj" << "\t" << "Država" << "\t" << "\t" << "Broj vozila" << "\t" << "Postotak"
<< "\n\r";
double postotakVozilo = 0;
for (int i = 0; i < dist.size(); i++)
{
    postotakVozilo = (double) dist[i].brojilo / (double) redniBrojVozila *
    100.0;
    datoteka << (i + 1) << "\t" << "\t" << dist[i].zemlja << "\t" << "\t" <<
    dist[i].brojilo << "\t" << "\t" << setprecision(2) << postotakVozilo <<
    "\n\r";
}
datoteka.close();
return S_OK;
}

```

Da bi aplikacija dala sve podatke za svako vozilo koje detektira potrebno je napraviti funkciju koja se poziva svaki put kada se detektira vozilo. Podatke koje će ta funkcija ispisivati su registarska oznaka (sadrži string (niz znakova) pročitane oznake), država (predstavlja tri slova države iz koje vozilo dolazi), pouzdanost (u intervalu [0-100] i predstavlja postotak pouzdanosti pročitane registarske oznake odnosno koliko je algoritam siguran da je dobro pročitao registarsku oznaku), sliku vozila (matrica koja predstavlja sliku detektiranog vozila), vrijeme (trenutno vrijeme u milisekundama) i broj markera (broj markera na kojem je trenutno detektirano vozilo).

```

bool detektirano_vozilo(LPCSTR reg_oznaka, LPCSTR drzava, int pouzdanost, Mat
&slika_vozila, int vrijeme, int broj_markera, vector<distribucija> &dist, int
&redniBrojVozila)
{

```

```

redniBrojVozila++;

ofstream podaci;
podaci.open("listaRegistarskihOznaka.txt", ios_base::app);
podaci << redniBrojVozila << "\t" << "\t" << reg_oznaka << "\t" << "\t" << "\t"
<< drzava << "\t" << "\t" << pouzdanost << "\t" << "\t" << vrijeme << "\t" <<
"\t" << broj_markera << "\n";
podaci.close();

int pozicijaElementa = -1;
bool drzavaNadena = false;

for (int i = 0; i < dist.size(); i++)
{
    if (!strcmp(drzava, dist[i].zemlja))
    {
        drzavaNadena = true;
        pozicijaElementa = i;
        break;
    }
}

if (drzavaNadena == true)
{
    dist[pozicijaElementa].brojilo++;
}

else
{
    distribucija var;
    var.brojilo = 1;
    strcpy_s(var.zemlja, drzava);
    dist.push_back(var);
}

char imeDatoteke[100];
sprintf_s(imeDatoteke, "%i.png", vrijeme);
imwrite(imeDatoteke, slika_vozila);

return true;
};

```

Cijeli algoritam za aplikaciju za prepoznavanje registarskih oznaka se nalazi u Dodatku A. U sljedećem poglavlju su objašnjeni rezultati dobiveni pomoću ove aplikacije.

6. Rezultati analize video snimki prometa zagrebačke obilaznice

Aplikacija je testirana pomoću stvarne video snimke cestovnog prometa zagrebačke obilaznice. Komercijalno dostupan SDK CARMEN je upotrijebljen za prepoznavanje registarskih oznaka. Za pokretanje aplikacije je korišten Visual Studio 2013 na 64-bitnom računalu sa instaliranim Windows 7 operacijskim sustavom sljedećih sklopovskih karakteristika: 4GB RAM memorije, Intelov Pentium CPU procesor.

6.1. Rezultati testiranja aplikacije

Aplikacijom za prepoznavanje registarskih oznaka dobivene su dvije datoteke. U prvu datoteku se pohranjuje prepoznata registarska oznaka, država iz koje vozilo dolazi, pouzdanost kojom je aplikacija točno prepoznala registarsku oznaku, vrijeme i broj markera koji je detektirao vozilo odnosno registarsku oznaku.

Aplikacija je detektirala 534 vozila na zagrebačkoj obilaznici od 10,5 sekunde do 1800,0 sekunde odnosno do kraja videa snimke. Pouzdanost točno prepoznatih oznaka varira u intervalu [0,100]. U aplikaciji je korišten jedan marker koji detektira registarske oznake (prikazano na Slici 19.).



Slika 19. Prikaz brojanja vozila pomoću markera

U Tablici 3. je prikazan broj vozila kojima je pouzdanost manja odnosno veća od 40 radi daljnje analize.

Tablica 3. Broj vozila s manjom odnosno većom pouzdanosti od 40

Broj vozila	Pouzdanost
41	<40
493	>40

U druga datoteku je zapisan broj vozila koja dolaze iz određene države i postotak tog broja od ukupnog broja vozila. Aplikacija je prepoznala 24 različite države, a države koje aplikacija nije mogla prepoznati spremila je pod nazivom nepoznato. U Tablici 4. su prikazani rezultati te datoteke.

Tablica 4. Lista prepoznatih država

Redni broj	Država	Broj vozila	Postotak
1	DEU	163	31%
2	POL	82	15%
3	AUT	77	14,63%
4	CZE	71	13%
5	Ostalo	62	11,57%
6	HRV	45	8,4%
7	Nepoznato	18	3,4%
8	SVN	16	3%
		Ukupno: 534	Ukupno: 100%

U nastavku slijedi analiza dobivenih rezultata, te dodatni komentari.

6.2. Analiza i komentari rezultata

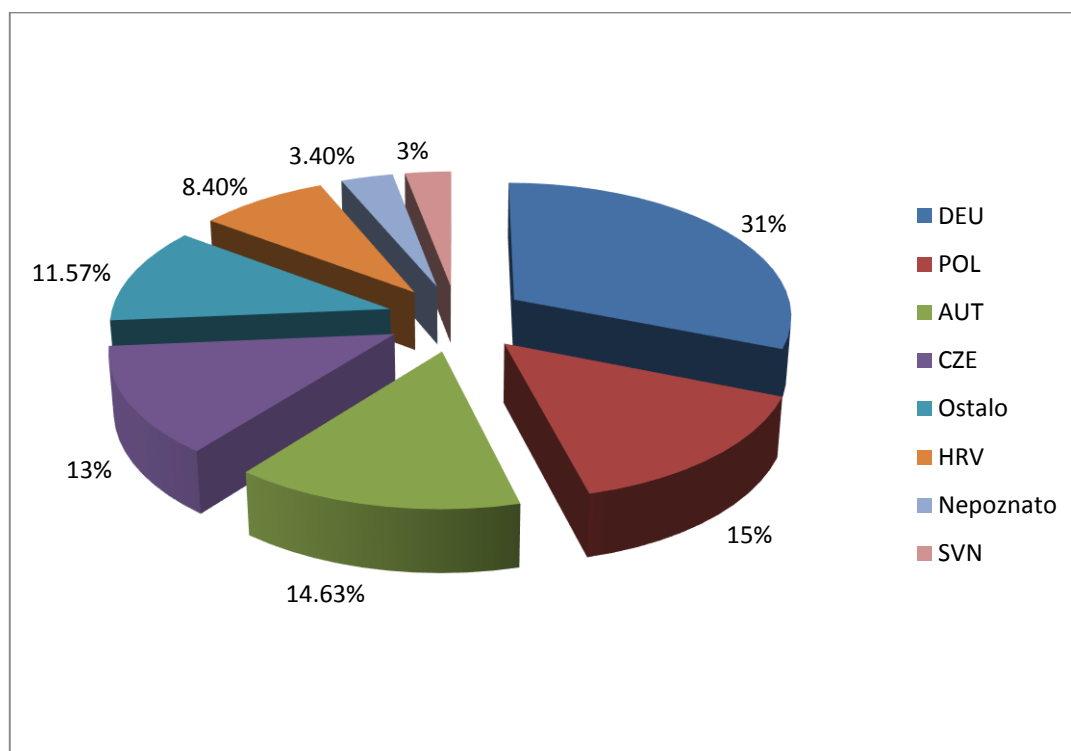
Rezultati prve datoteke prikazuju dobru učinkovitost sustava za prepoznavanje registarskih oznaka. Tablica 3. prikazuje broj vozila pri kojima je pouzdanost automatskog prepoznavanja registarske oznake vozila veća odnosno manja od 40 %. Vrlo je mali broj vozila pogrešno prepoznatih registarskih oznaka naspram ukupnog broja vozila.

Zbog toga što stvarnovremensko izvršavanje nije zajamčeno, potrebno je provesti daljnju optimizaciju. Većina primijenjenih algoritama radi dovoljno dobro da bi se moglo provesti izvršavanje u stvarnom vremenu, ali su primjenom ALPR

metode stvoreni problemi po tom pitanju. U ekstrakciji cestovnog vozila i korištenje algoritma koji koristi fiksnu veličinu slike steklo se poboljšanje performansi.

Rezultati pokazuju da predloženi sustav može prikupljati podatke o cestovnim vozilima koja putuju u jednom smjeru. Sustav je većinom uspješno detektirao vozilo i njegovu registarsku oznaku. Podaci koji su prikupljeni se mogu koristiti za daljnju analizu i prometnu statistiku zagrebačke obilaznice odnosno drugih cestovnih prometnica.

Rezultati druge datoteke prikazuju broj i postotak vozila koja dolaze iz određene države u odnosu na ukupan broj vozila na zagrebačkoj obilaznici. Rezultati se nalaze u Tablici 4. Aplikacija je otkrila 24 različitih zemalja. Od ukupnog broja otkrivenih vozila, program nije mogao prepoznati zemlju podrijetla kod samo 18 vozila. Njemačka je zemlja s najviše prepoznatih vozila. Možemo zaključiti da je ova zagrebačka obilaznica važna za hrvatsku prometnu mrežu zbog velikog broja turista koji se njome koriste i stvaraju značajan protok prometa tijekom turističke sezone. Na Slici 20. se nalazi grafikon koji prikazuje popis država s postotkom vozila od ukupnog broja vozila na zagrebačkoj obilaznici.



Slika 20. Popis država s postotkom vozila od ukupnog broja vozila na zagrebačkoj obilaznici

Daljnji razvoj predloženog sustava trebao bi se usredotočiti na praćenje većeg broja vozila odnosno praćenje prometa veće cestovne prometne mreže te omogućiti procjenu OD matrice, mjerenje srednje brzine vozila i procjenu prometnog toka.

7. Zaključak

Današnji problemi u cestovnom prometu uključuju problem povećane potražnje prometa u vrijeme vršnih sati, brz odaziv žurnih službi u slučaju incidentne situacije, problem zastoja i sl. Sve je više prostora u inteligentnim transportnim sustavima (ITS) za usvajanje sustava video analize snimke cestovnog prometa u svrhu upravljanja cestovnim prometom. Takvi sustavi se mogu primijeniti na autocestama, ali i u izazovnijim urbanim područjima. Sustav koristi kombinaciju segmentacije i informacije o pokretu za prepoznavanje i praćenje pokretnih objekata na cesti korištenjem različitih metoda. Glavne zadaće koje bi ovaj sustav obavljao je brojanje vozila, mjerenje brzine vozila, klasifikacija vozila i praćenje vozila prepoznavanjem njihovih registarskih oznaka. Pri obavljanju tih poslova sustav će imati određenih problema kao što je izlučivanje značajki sa slike, raspoznavanje te detekcija vozila i segmentacija slike.

ITS bi trebao usvojiti video analizu kao metodu praćenja cestovnog prometa zbog toga što ovakav sustav može prikupljati podatke u stvarnom vremenu, brojiti vozila, kategorizirati vozila u osam kategorija, može odrediti ulaz i izlaz vozila iz raskrižja te može izmjeriti brzinu vozila. Sustav ima sposobnost da se bavi relativno neograničenom okolinom s vozilima koji idu u svim smjerovima, a moguća je i detekcija prisutnosti pješaka odnosno pješaci ne predstavljaju smetnju.

Sustav ima i nekih nedostataka kao na primjer može raditi samo tijekom dana zbog sjaja ulične rasvjete na kolniku ukoliko nije opremljen odgovarajućim vlastitim osvjetljenjem. Kad je noć sustav prepozna nepovoljne uvijete i automatski zaustavlja ekstrakciju podataka ili uključuje vlastitu rasvjetu nevidljivu ljudskom oku. Uz taj nedostatak, sustav ima i drugih problema kao što je detekcija dva vozila, koja se nalaze blizu jedno drugom kao jedno ili nemogućnost detekcija većih objekata bez uključivanja drugih bliskih objekata kao što je na primjer tramvaj. Također, sustavu predstavljaju problem veće skupine ljudi, sjene, jaka osvjetljenja i slaba rezolucija slike.

Što se tiče OD matrica, dobiveni su dobri početni rezultati, ali preostaje za riješiti velike probleme kao što su detekcija dva vozila kao jedno i nemogućnost identifikacije većih vozila (na primjer tramvaja). Uбудuće, ti će se problemi nastojati riješiti.

Današnji problemi upravljanja cestovnim prometom uključuju rješavanje prometnih situacija koje se sastoje od velikog broja zagušenja, povećanja zahtjeva za protokom vozila tijekom vršnog sata, te potrebe za velikom mobilnošću i brzinom reakcije tijekom incidenta. Video senzor ili kamera u kombinaciji s najsuvremenijim algoritmima za obradu slika sve više postaje pristup rješavanju problema nadzora cestovnog prometa.

Rezultati koji su dobiveni aplikacijom za prepoznavanje registarskih oznaka prikazuju prepoznatu registarsku oznaku, pouzdanost kojom je aplikacija točno prepoznala registarsku oznaku, državu iz koje vozilo dolazi, vrijeme i broj markera koji je detektirao vozilo odnosno registarsku oznaku.

Aplikacija je testirana pomoću stvarne video snimke sa zagrebačke obilaznice Lučko. Komercijalno dostupan SDK CARMEN je upotrebljen za prepoznavanje registarskih oznaka. Rezultati prikazu dobru učinkovitost sustava za prepoznavanje registarskih oznaka. Pouzdanost kod većine prepoznatih registarskih oznaka je vrlo dobra. Aplikacija je također zapisivala broj vozila po određenoj zemlji. Ti podaci nam prikazuju da većina vozila koja putuju zagrebačkom obilaznicom dolazi iz Njemačke. Dobiveni podaci mogu se koristiti za izradu prometne statistike na bilo kojoj cestovnoj prometnici.

Literatura

- [1] I. Dadić, G. Kos, Teorija i organizacija prometnih tokova (skripta), Sveučilište u zagrebu, Fakultet prometnih znanosti, Zagreb, 2007.
- [2] S. Varela Ramirez, K. Kovačić, Origin-Destination Matrix Estimation of Traffic Flow on Highway Network, 2013.
- [3] N. Buch, S. A. Velastin, J. Orwell, A Review of Computer Vision Techniques for the Analysis of Urban Traffic, 2011.
- [4] S. Messelodi, C.M. Modena, M. Zanin, A computer vision system for the detection and classification of vehicles at urban road intersections, London, 2005.
- [5] M. Tahir Qadri, M. Asif, Automatic number plate recognition system for vehicle indentification using optical character recognition, 2009.
- [6] http://en.wikipedia.org/wiki/Automatic_number_plate_recognition, pristupljeno 4.6.2015.
- [7] K. Kovačić, E. Ivanjko, S. Varela, Real Time Vehicle Countra of origin Classification Based on Computer Vision, ISEP, 2014.
- [8] M. Maurer, R. Behringer, S. Furst, F. Thomanek, E.D. Dickmanns, A compact vision system for road vehicle guidance, Universitat der Bundeswehr Munchen (UBM), 1996.
- [9] <http://www.licenseplaterecognition.com/>, pristupljeno 10.6.2015.
- [10] J. Baker, M. D. Rossetti, Vendor Information and Evaluation Techniques for an Automated License Plate Reading System, The University of VirginiaVirginia, Department of Systems Engineering, 2001.
- [11] <http://www.alphatech.com/>, pristupljeno 11.6.2015.
- [12] C. Anagnostopoulos, License Plate Recognition: A Brief Tutorial, IEEE Intelligent Transportation Systems Magazine, 2014. (objavljeni rad, znanstveni)

Popis kratica

ITS		inteligentni transportni sustavi
ANPR	<i>automatic number plate recognition</i>	automatsko prepoznavanje registrarskih tablica
SCOCA	<i>System for COunting and Classifying Automatically vehicles</i>	automatsko brojanje i razdvajanje vozila
OD	<i>Origin-desination</i>	podrijetlo-odredište
DIP	<i>Digital Image Processing</i>	digitalna tehnika obrade slike
CCA	<i>Connected Component Analysis</i>	analiza povezanih komponenti
OCR	<i>Optical character recognition</i>	optičko prepoznavanje znakova
DLL	<i>dynamic link library</i>	biblioteka programa s dinamičkim povezivanjem

Popis tablica

Tablica 1. Klase objekata i njihovi pripadajući modeli [4]	9
Tablica 2. Usporedba estimiranih OD matrica sa mjerenjima [8]	17
Tablica 3. Broj vozila s manjom odnosno većom pouzdanosti od 40	30
Tablica 4. Lista prepoznatih država	30

Popis slika

Slika 1. Gustoća prometnog toka [1].....	2
Slika 2. Razmaci u slijeđenju vozila [1]	4
Slika 3. Osnovni prometni parametri [2]	5
Slika 4. Blok dijagram za nadzorni sustav [3].....	6
Slika 5. 3D modeli koji se koriste za prvu razinu klasifikacije [4].....	7
Slika 6. Primjer uzimanja 3D modela koji ima najbolji rezultat [4].....	8
Slika 7. Koraci ALPR-a [7].....	11
Slika 8. Oblik analize u binarnom objektu[13]	12
Slika 9. Dijagram predloženog postupka [8]	15
Slika 10. Rezultati detekcije objekata [8]	16
Slika 11. Ukupna arhitektura sustava [9]	18
Slika 12. Prošireni sustav za praćenje prometa [4]	19
Slika 13. Arhitektura SCOCA [4]	20
Slika 14. Tipična konfiguracija ALPR sustava [10]	21
Slika 15. Komponente ALPR sustava [12].....	22
Slika 16. Arhitektura aplikacije [2]	23
Slika 17. Arhitektura autoceste sa stajališta baze podataka [2]	24
Slika 18. Blokovska shema rada aplikacije.....	25
Slika 19. Prikaz brojanja vozila pomoću markera	29
Slika 20. Popis država s postotkom vozila od ukupnog broja vozila na zagrebačkoj obilaznici	31

Dodatak A Algoritam aplikacije za prepoznavanje registarskih oznaka

```
#pragma once

#include <gxsdldr.cpp>
#include <iomanip>
#include <math.h>

#include "Include.h"
#include "Marker.h"
#include "LPR.h"

struct distribucija
{
    int brojilo;
    CHAR zemlja[4];
};

bool detektirano_vozilo(LPCSTR reg_oznaka, LPCSTR drzava, int pouzdanost, Mat
&slika_vozila, int vrijeme, int broj_markera, vector<distribucija> &dist, int
&redniBrojVozila);
void izradi_markere(vector<Marker> &markeri, double omjerX, double omjerY);
bool vozilo_na_markeru(Marker &marker, Mat &slika_detekcije);
void provjeri_markere(vector<Marker> &markeri, Mat &slika_detekcije);
void nacrtaj_markere(vector<Marker> &markeri, Mat &slika);

int main(int argc, char **argv, char **envp)
{
    int redniBrojVozila = 0;

    CHAR ime_video_datoteke[] = "C:\\\\Lucko_1_mpeg4.mp4";

    double umanji_sliku_po_x = 0.225;

    double umanji_sliku_po_y = 0.225;

    double pocni_video = 10500.0;
    double zavrsi_video = 1800000.0;

    namedWindow("Video");
    namedWindow("Detektor");

    vector<distribucija> dist;

    BackgroundSubtractorMOG2 detektor(100, 15, true);

    VideoCapture video(ime_video_datoteke);

    Mat slika_velika, slika_umanjena, slika_detektor;

    video.set(CV_CAP_PROP_POS_MSEC, pocni_video);
```

```

vector<Marker> markeri;
izradi_markere(markeri, umanji_sliku_po_x, umanji_sliku_po_y);

LPR prep_reg_oz;

ofstream podaci;
podaci.open("listaRegistarskihOznaka.txt", ios_base::ate);
podaci << "Ispis registarskih oznaka, zemlje porijekla, pouzdanost sustava
da je točno prepoznao znakove na registarskoj oznaci i vrijeme u videu kada je
marker detektirao registarsku oznaku." << "\n" << "\n" << "Redni broj" << "\t" <<
"Registarska oznaka" << "\t" << "Drzava" << "\t" << "\t" << "Pouzdanost" << "\t"
<< "Vrijeme" << "\t" << "\t" << "Broj markera" << "\n";
podaci.close();

while (video.read(slika_velika))
{

    int vrijeme = static_cast<int>(video.get(CV_CAP_PROP_POS_MSEC));

    if (vrijeme >= zavrshi_video) { break; }

    resize(slika_velika, slika_umanjena, Size(), umanji_sliku_po_x,
    umanji_sliku_po_y);

    detektor(slika_umanjena, slika_detektor);
    provjeri_markere(markeri, slika_detektor);

    for (size_t i = 0; i < markeri.size(); i++)
    {
        if (markeri[i].VoziloDetektirano == true &&
            markeri[i].RegOznakaProcitana == false)
        {
            prep_reg_oz.Prepoznaj(slika_velika,
            markeri[i].PodrucjeVeliko);

            cmNP *rezultat = prep_reg_oz.Prepoznaj(slika_velika,
            markeri[i].PodrucjeVeliko);

            if (rezultat)
            {
                CHAR reg_oznaka[4];

                prep_reg_oz.PrevediRegOznaku(rezultat->type,
                reg_oznaka);

                Mat slika_vozila =
                slika_velika(markeri[i].PodrucjeVeliko);

                markeri[i].RegOznakaProcitana =
                detektirano_vozilo(rezultat->text, reg_oznaka, rezultat-
                >confidence, slika_vozila, vrijeme, i, dist,
                redniBrojVozila);
            }
        }
    }
}

```

```

        }
    }
}

nacrtaaj_markere(markeri, slika_umanjena);

imshow("Video", slika_umanjena);
imshow("Detektor", slika_detektor);
waitKey(1);
}

ofstream datoteka;
datoteka.open("listaDrzava.txt", ios_base::ate);
datoteka << "Ispis liste drzava, broj vozila detektiranih iz odredene drzave
i postotak tog broja naspram ukupnog broja vozila." << "\n" << "\n" << "Redni
broj" << "\t" << "Država" << "\t" << "\t" << "Broj vozila" << "\t" << "Postotak"
<< "\n\r";
double postotakVozilo = 0;
for (int i = 0; i < dist.size(); i++)
{
    postotakVozilo = (double) dist[i].brojilo / (double) redniBrojVozila
* 100.0;
    datoteka << (i + 1) << "\t" << "\t" << dist[i].zemlja << "\t" << "\t"
<< dist[i].brojilo << "\t" << "\t" << setprecision(2) << postotakVozilo << "\n\r";
}
datoteka.close();
return S_OK;
}

void izradi_markere(vector<Marker> &markeri, double omjerX, double omjerY)
{
    markeri.push_back(Marker(Rect(
        780, 280,
        1200-780, 600-280),
        omjerX,
        omjerY));
}

bool vozilo_na_markeru(Marker &marker, Mat &slika_detekcije)
{
    Mat dio_slike(slika_detekcije(marker.PodrucjeMalo));

    int broj_pokretnih_piksela = 0;

    for (int i = 0; i < dio_slike.rows * dio_slike.cols; i++)
    {
        unsigned char piksel = dio_slike.at<unsigned char>(i);

        if (piksel != 0)
        {
            broj_pokretnih_piksela++;
        }
    }
}

```

```

        marker.BrojPokretnihPiksela = broj_pokretnih_piksela;

        if (broj_pokretnih_piksela > 500)
        {
            return true;
        }
        else
        {
            return false;
        }
    }

void provjeri_markere(vector<Marker> &markeri, Mat &slika_detekcije)
{
    for (size_t i = 0; i < markeri.size(); i++)
    {
        bool detektirano = vozilo_na_markeru(markeri[i], slika_detekcije);

        if (detektirano == true)
        {
            if (markeri[i].VoziloDetektirano == false && markeri[i].Aktivan
>= 10)
            {
                markeri[i].BrojAktivacija++;
                markeri[i].VoziloDetektirano = true;
            }

            markeri[i].Neaktivan = 0;
            markeri[i].Aktivan++;
        }
        else
        {
            if (markeri[i].Neaktivan < 3)
            {
                markeri[i].Neaktivan++;
            }
            else
            {
                markeri[i].Aktivan = 0;
                markeri[i].VoziloDetektirano = false;
                markeri[i].RegOznakaProcitana = false;
            }
        }
    }
}

void nacrtaj_markere(vector<Marker> &markeri, Mat &slika)
{
    Scalar boja_crvena(0.0, 0.0, 255.0, 0.0);
    Scalar boja_zelena(0.0, 255.0, 0.0, 0.0);

    char txt_prebrojana_vozila[20];

    for (size_t i = 0; i < markeri.size(); i++)
    {
        sprintf_s(txt_prebrojana_vozila, "%i", markeri[i].BrojAktivacija);
        Point lokacija_teksta(
            markeri[i].PodrucjeMalo.x + markeri[i].PodrucjeMalo.width / 2,

```

```

        markeri[i].PodrucjeMalo.y + markeri[i].PodrucjeMalo.height /
2);

    if (markeri[i].Neaktivan < 3)
    {
        rectangle(slika, markeri[i].PodrucjeMalo, boja_crvena, 1);
        putText(slika, txt_prebrojana_vozila, lokacija_teksta,
FONT_HERSHEY_SIMPLEX, 2.0, boja_crvena);
    }
    else
    {
        rectangle(slika, markeri[i].PodrucjeMalo, boja_zelena, 1);
        putText(slika, txt_prebrojana_vozila, lokacija_teksta,
FONT_HERSHEY_SIMPLEX, 2.0, boja_zelena);
    }

}

}

bool detektirano_vozilo(LPCSTR reg_oznaka, LPCSTR drzava, int pouzdanost, Mat
&slika_vozila, int vrijeme, int broj_markera, vector<distribucija> &dist, int
&redniBrojVozila)
{
    redniBrojVozila++;

    ofstream podaci;
    podaci.open("listaRegistarskihOznaka.txt", ios_base::app);
    podaci << redniBrojVozila << "\t" << "\t" << reg_oznaka << "\t" << "\t" <<
"\t" << drzava << "\t" << "\t" << pouzdanost << "\t" << "\t" << vrijeme << "\t" <<
"\t" << broj_markera << "\n";
    podaci.close();

    int pozicijaElementa = -1;
    bool drzavaNadena = false;

    for (int i = 0; i < dist.size(); i++)
    {
        if (!strcmp(drzava, dist[i].zemlja))
        {
            drzavaNadena = true;
            pozicijaElementa = i;
            break;
        }
    }

    if (drzavaNadena == true)
    {
        dist[pozicijaElementa].brojilo++;
    }

    else
    {
        distribucija var;
        var.brojilo = 1;
        strcpy_s(var.zemlja, drzava);
        dist.push_back(var);
    }
}

```

```
char imeDatoteke[100];  
sprintf_s(imeDatoteke, "%i.png", vrijeme);  
imwrite(imeDatoteke, slika_vozila);  
  
return true;  
};
```



Sveučilište u Zagrebu
Fakultet prometnih znanosti
10000 Zagreb
Vukelićeva 4

METAPODACI

Naslov rada: Analiza distribucije vozila prema zemlji porijekla na zagrebačkoj obilaznici
primjenom metoda video detekcije

Autor: Sanja Palajsa, 0135225970

Mentor: doc. dr. sc. Edouard Ivanjko

Naslov na drugom jeziku (engleski):

Analysis of the Vehicle Country of Origin Distribution on the Zagreb Bypass using Video
Detection Methods

Povjerenstvo za obranu:

- doc. dr. sc. Niko Jelušić, predsjednik
- doc. dr. sc. Edouard Ivanjko, mentor
- Mario Buntić, mag. ing. traff. (neposredni voditelj), član
- prof. dr. sc. Hrvoje Gold, zamjena

Ustanova koja je dodjelila akademski stupanj: Fakultet prometnih znanosti Sveučilišta u Zagrebu

Zavod: Zavod za inteligentne transportne sustave

Vrsta studija: sveučilišni

Naziv studijskog programa: Inteligentni transportni sustavi i logistika

Stupanj: preddiplomski

Akademski naziv: univ. bacc. ing. traff.

Datum obrane završnog rada: 15. rujna 2015.



Sveučilište u Zagrebu
Fakultet prometnih znanosti
10000 Zagreb
Vukelićeva 4

IZJAVA O AKADEMSKOJ ČESTITOSTI I SUGLASNOST

Izjavljujem i svojim potpisom potvrđujem kako je ovaj _____ završni rad

isključivo rezultat mog vlastitog rada koji se temelji na mojim istraživanjima i oslanja se na objavljenu literaturu što pokazuju korištene bilješke i bibliografija.

Izjavljujem kako nijedan dio rada nije napisan na nedozvoljen način, niti je prepisan iz necitiranog rada, te nijedan dio rada ne krši bilo čija autorska prava.

Izjavljujem također, kako nijedan dio rada nije iskorišten za bilo koji drugi rad u bilo kojoj drugoj visokoškolskoj, znanstvenoj ili obrazovnoj ustanovi.

Svojim potpisom potvrđujem i dajem suglasnost za javnu objavu _____ završnog rada

pod naslovom **Analiza distribucije vozila prema zemlji porijekla na zagrebačkoj
obilaznici primjenom metoda video detekcije**

na internetskim stranicama i repozitoriju Fakulteta prometnih znanosti, Digitalnom akademskom repozitoriju (DAR) pri Nacionalnoj i sveučilišnoj knjižnici u Zagrebu.

Student/ica:

U Zagrebu, 9/2/2015

(potpis)