**SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU**

**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA**

**SUSTAV ZA PREPOZNAVANJE MARKE AUTOMOBILA NA TEMELJU SLIKE**

**Projektni zadatak iz kolegija Multimedijski sustavi**

**Studenti:**

Filip Žagar

Hrvoje Kokošarević

Mislav Marić

Ivan Marošević

**Osijek, 2018.**

# UVOD

Danas, u vrijeme kada računala ulaze u sve aspekte ljudskog života, sve se veći broj procesa pokušava automatizirati pomoću računala. Jedan takav proces je nadzor prometa na cestama. Kako bi se vršio nadzor prometa, potrebno je izgraditi sustav koji je u mogućnosti detektirati vozilo na slici, a zatim po potrebi uzimati dio slike koji mu treba. Takvi sustavi koriste se primjerice kod automatske naplate parkinga, cestarine itd. Velik je broj sustava koji rade jednu od spomenutih funkcija s jako velikom preciznošću i velikom brzinom rada. U sklopu ovog rada razmatrani su problemi detekcije i određivanja pozicije znaka na zadanoj slici automobila te njegovo prepoznavanje pomoću određenog algoritma. Razvojem ovoga sustava moguća je daljnja implementacija i korištenje u razne svrhe. Tehnike obrađivanja slike izuzetno su složene, a jako veliki utjecaj u ovakvim sustavima igraju raznolikost automobila, različiti vremenski uvjeti, količina svjetlosti, kvaliteta slike i mnogi drugi. Cilj ovoga rada je razviti sustav koji će dati korisne informacije odnosno podatak koji govori o kojoj marki automobila je riječ na slici, a algoritam je izrađen u MATLAB programu.

# OPIS POSTOJEĆIH RJEŠENJA

Prva metoda koja je obrađena predstavljena je od strane Burkhard, Minich, Li; u radu pod nazivom Vehicle Logo Recognition and Classification: Feature Descriptors vs. Shape Descriptors. U ovome radu najviše pažnje posvećuje se uspoređivanju loga automobila sa bazom podataka korištenjem izračunatih sličnosti, te na temelju njih se određuje najvjerojatniji proizvođač, odnosno logo koji ima najviše sličnosti sa testiranim primjerom. Prepoznavanje marki automobila bazira se na SIFT algoritmu. SIFT (eng. Scale-Invariant Feature Transform) jest algoritam koji se koristi u računalnom vidu, da bi se prepoznala i opisala svojstva koja se nalaze na slici.

Prilikom prepoznavanja marke vozila, koristi se pretpostavka da svaki proizvođač sa namjerom dizajnira i pozicionira logo tako da bude istaknut. Odnosno, većina loga automobilskih marki se na neki način ističe od tijela vozila; bilo fizički (stvarno izbočenje na vozilu) ili svjetlijom bojom (većinom su loga kontrastne boje od vozila); koriste oštre rubove te imaju neki oblik granice oko sebe (kružnica, elipsa, pravokutnik). SIFT je lako upotrebljiv jer za prepoznavanje nisu bitni uvjeti, usmjerenje, perspektiva i rotacija loga. Kako sve fotografije vozila ne mogu biti savršeno uhvaćene, postoje fotografije koje jako odskaču od prosjeka (eng. Outliers). Kako bi se te fotografije izbacile iz baze koristi se RANSAC (eng. Random Sample Consensus).

Postupak korištenja SIFT algoritma obavlja se slijedećim koracima:

* Prvi korak je prebacivanje fotografije u sive tonove (eng. Grayscale).
* Primjenjivanje niskopropusnog filtra te smanjivanje razine šuma.
* Izoliranje interesnog područja.
* Primjenjivanje SIFT algoritma.
* Primjena 'ratio' test-a.
* Odbacivanje 'outliers'-a.
* Uspoređivanje ključnih točaka testirane fotografije sa referentnim logom iz baze.
* Izračunavanje sličnosti između testirane fotografije i različitih referentnih fotografija iz baze. Sličnost se određuje na temelju najbližeg susjeda(odnosno najmanje udaljenosti).

Problem koji se pojavljuje kod klasične primjene SIFT-a jest ta da je jako ovisan o količini svjetlosti i količini kontrasta na slici. Također, jedan od problema jest bio i to da je veliki udio loga automobila kromirano, pa se od takve boje svjetlost jako lako odbija te nije moguće primijeniti neke od koraka SIFT-a.

Druga metoda koja je obrađena, predstavljena je od strane Jelene Novosel u radu pod nazivom Sustav računalnog vida za automatsko prepoznavanje vozila u svrhu nadzora prometa. Cilj ovoga rada je razvoj sustava računalnog vida za nadzor prometa. Sustav uključuje detekciju vozila na prometnicama, prepoznavanje registarskih oznaka, te određivanje tipa i marke vozila na prometnici.

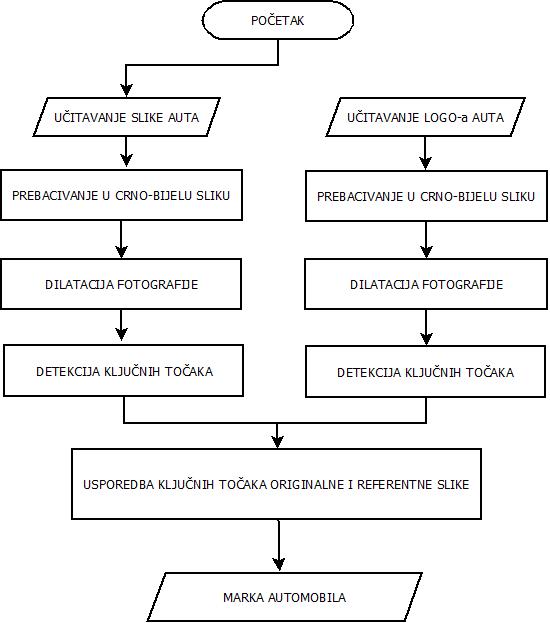
Prvi korak detekcije vozila jest odabir interesnog područja, odnosno odvajanje vozila od pozadine. Odabrana slika pretvara se u sivu sliku koja za reprezentaciju posjeduje jedan kanal s 8 bitova. Daljnja obrada kao rezultat daje informaciju je li vozilo prošlo prometnicom ili ne, a ova informacija se dobiva promatranjem područja od interesa. Lokalizacija registarske tablice se vrši pomoću detekcije rubova slike, te se pomoću vertikalne i horizontalne projekcije odlučuje koji prostor na vozilu pripada registarskoj tablici. Završetkom ove obrade pronađeni su mogući kandidati za registarske tablice koji se potom ispituju na temelju značajki tablice: značajka omjera stranica tablice, količina bijele boje i položaj tablice.

Nakon uspješne lokalizacije registarske tablice, moguće je započeti sa procesom detekcije logo oznake na automobilu. Ukoliko se pronađe položaj registarske oznake, moguće je detektiranu regiju podignuti za određen broj piksela kako bi detektirana regija obuhvatila logo oznaku. Visina podizanja regije mora biti dovoljno velika jer različiti automobili posjeduju logo oznake na različitim mjestima. Izdvajanje logo oznake iz regije približnog položaja provodi se pomoću detekcije rubova, dilatacije i labeliranja. Metoda prepoznavanja logo oznake radi na principu korelacije između predložaka oznake i detektirane logo oznake. Izdvajanje logo oznake predstavlja najveći problem prilikom klasifikacije marke automobila. Jedan od mogućih problema predstavlja boja automobila, naime svijetlije boje automobila otežavaju prepoznavanje logo oznake jer je većina logo oznake metalik boje te se stopi sa ostatkom automobila. U radu su testirana 142 automobila, od čega je ispravno klasificirani njih 110. te ovaj algoritam posjeduje točnost od 77,46%.

# PREDSTAVLJANJE VLASTITOG RJEŠENJA

Algoritam radi tako da uspoređuje originalnu sliku automobila sa određenim znakovima koji se nalaze u bazi podataka te na osnovu dobivenih rezultata procjenjuje o kojem je automobilu riječ. U bazi automobila nalaze se znakovi deset različitih marki automobila koji služe za usporedbu s originalnom slikom. Ovaj program radi tako da se najprije učita slika, izvrše određene korekcije te filtriranja slike. Nakon toga, detektira se tablica što pomaže u određivanju položaja znaka na slici, a zatim se provedi usporedba s referentnim slikama u bazi podataka. Usporedba se vrši tako da se originalna slika usporedi redom sa svim referentima u bazi te na osnovu rezultata odluči kojoj je najsličnija.

Dijagram toka:



## Učitavanje slike auta i logo-a

Učitavanje slike u MATLAB vrši se pomoću funkcije *imread.* Funkcija učitava sliku iz datoteke i sprema ju u matricu. Primjer: M *= imread('auto.png');* koja učitava sliku „auto“ i sprema ju u matricu M.



Slika 1: Učitana fotografija automobila



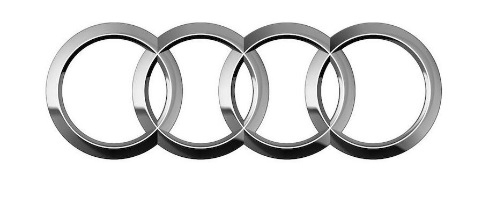
Slika 2: Učitan logo automobila

## Prebacivanje fotografije u crno-bijelu fotografiju

Prebacivanje RGB fotografije u crno-bijelu fotografiju radi se pomoću funkcije *rgb2gray* koja je implementirana u MATLAB. Ova funkcija pretvara RGB fotografiju u nijanse sive boje uklanjanjem nijansi i zasićenja zadržavajući osvjetljenje.



Slika 3: Crno-bijela fotografija



Slika 4: Crno-bijeli logo

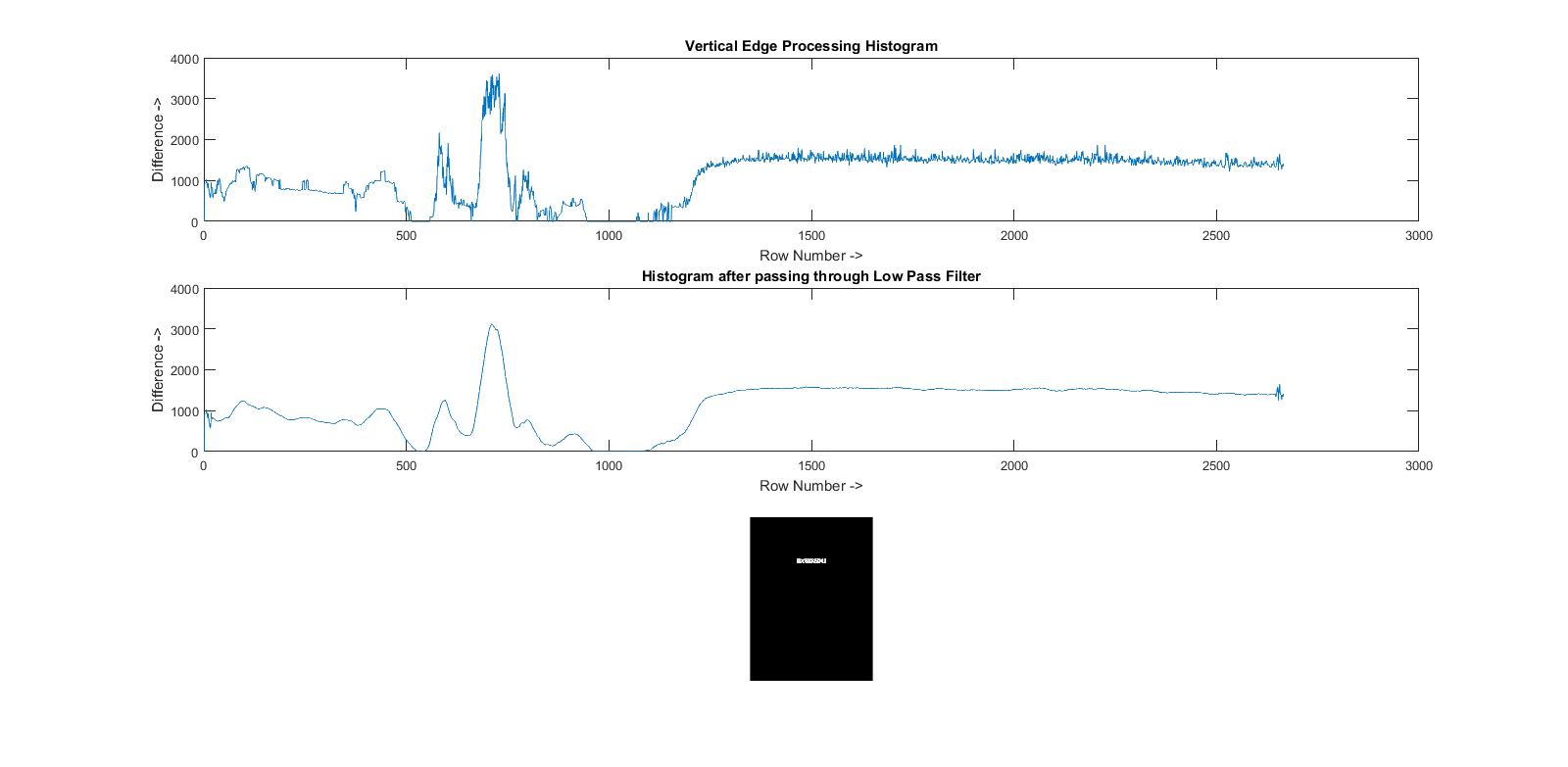
## Dilatacija fotografije

Dilatacija se vrši pomoću MATLAB funkcije *imdilate.* Funkcija proširuje sliku sivim tonovima ili binarnom slikom vraćajući proširenu sliku.

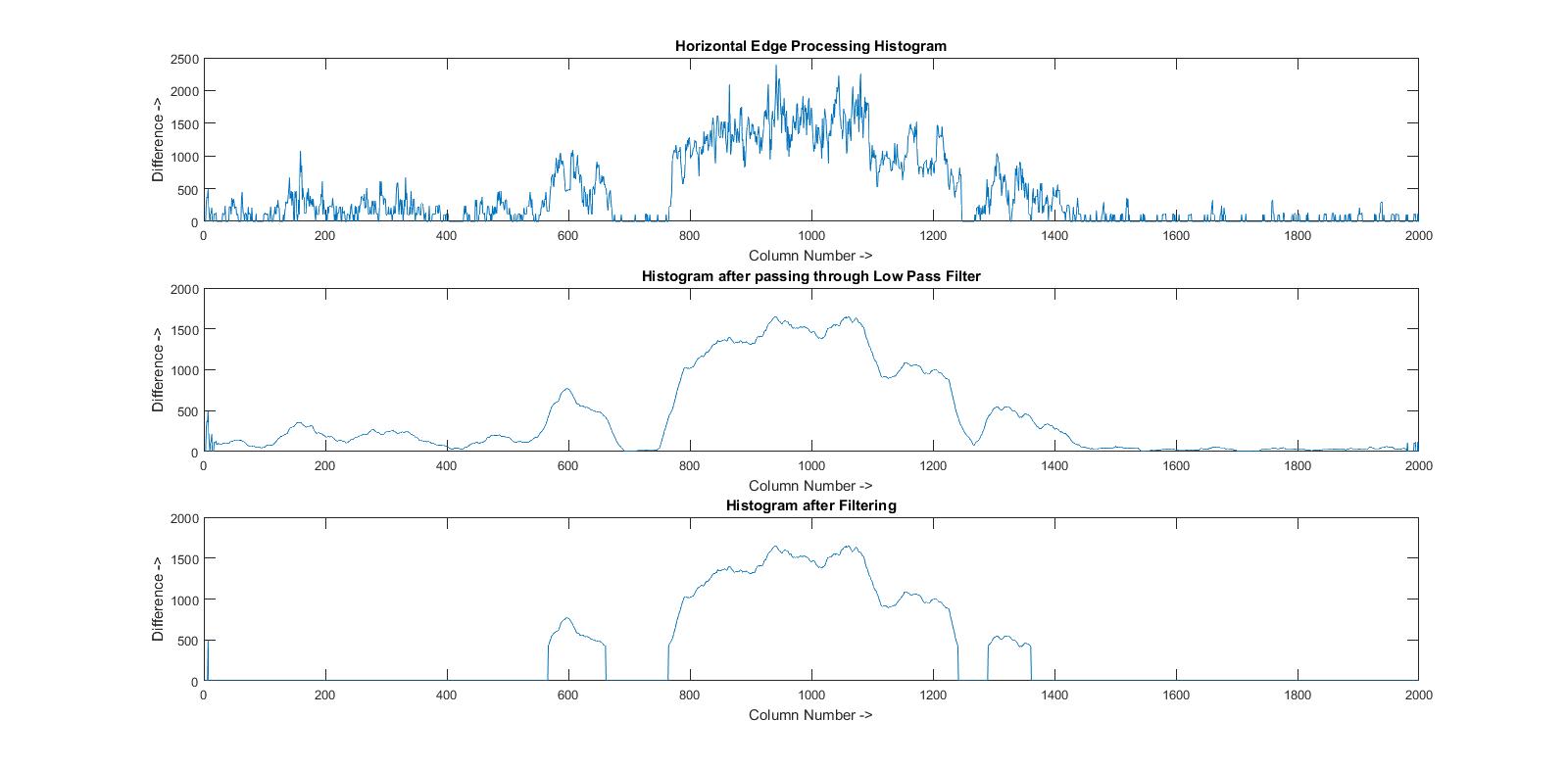
SLIKAA????

## Detekcija tablice na slici

Osnovna ideja za pronalazak tablice na slici temeljena je na činjenici da na tablici postoje izraženi prijelazi, koji se javljaju kao posljedica bijele pozadine i crnih slova. Za izdvajanje registarske pločice potrebno je analizirati rubove na slici te ih statistički obraditi. Proces započinje detekcijom rubova slike i to tako da se odrede i horizontalni i vertikalni rubovi na slici. Vertikalni rubovi dobivaju se pomoću vertikalne projekcije pomoću koje dobivamo graf na kojem su svi pikseli slike preslikani na y-os. Na mjestu gdje su rubovi izraženiji, javljaju se i veće amplitude na grafu. Na taj način moguće je izdvojiti pojedino područje slike. Horizontalna projekcija radi na isti način kao i vertikalna samo se pikseli preslikavaju na x-os. Analizom vertikalne projekcije dobivaju se mogući kandidati za tablicu odnosno „trake“, koje se dalje obrađuju i u konačnici se dobiva izrezana tablica sa slike. Budući da su vrhovi grafa jako oštri, što je jako nezgodno za daljnju analizu slike, graf se najprije propušta kroz niskopropusni filter kako bi se vrhovi „zagladili“. Kandidati se odabiru na temelju razlike dva susjedna elementa slike, te se promatraju samo vrhovi grafa koji su veći od određene razlike između dva „susjeda“. Kombinacijom horizontalne i vertikalne projekcije dobiva se lokacija tablice koja je početak za pronalazak znaka na automobilu, jer se znak obično nalazi iznad tablice.



Slika 5: Graf vertikalne projekcije



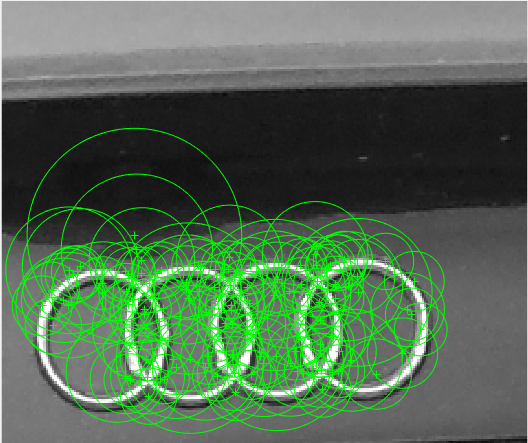
Slika 6: Graf horizontalne projekcije

* 1. **Izdvajanje logo oznake automobila pomoću lokacije registarske tablice**

Detekciju i izdvajanje logo oznake automobila nije moguće napraviti bez prethodno opisane detekcije registarske tablice. Naime kod većine automobila logo oznaka se nalazi na sredini iznad registarske tablice. Visina na kojoj je logo oznaka nalazi mijenja se ovisno od vrste i marke automobila, te se na različitim automobilima ova oznaka nalazi na različitim visinama od registarske oznake. Nakon što je uspješno izdvojena lokacija tablice, te je tablica izrezana iz početne fotografije, može se krenuti sa izdvajanjem logo oznake. Definiraju se varijable x i y, varijabla x predstavlja redak, a varijabla y predstavlja stupac. Vrijednosti ovih varijabli se postavljaju na nulu kako bi se dobila početna točka fotografije registarske oznake. Ukoliko je ovaj uvjet ispunjen, varijable x i y se definiraju kao pikselix i pikseliy. Definiraju se maksimalna i minimalna vrijednost varijabli pikselix i pikseliy koje predstavljaju početnu i krajnju točku retka, odnosno stupca navedene fotografije registarske oznake, a označene su sa minx, miny, maxx i maxy. Širina registarske oznake se definira oduzimanjem maksimalne i minimalne vrijednosti varijable pikselix, dok se visina registarske oznake definira oduzimanjem maksimalne i minimalne vrijednosti varijable pikseliy. Zatim se definiraju točke koje predstavljaju rubove registarske tablice. Ove točke su definirane na sljedeći način: tocka1 sadrži koordinate varijabli minx i miny, točka2 sadrži koordinate varijabli maxx i miny, te su analogno tome definirane točka3 i točka4. lokacije logo oznake dobije se varijabla height poveća za 2,9 puta, dok se varijabla width smanji za 3,5 puta kako bi se dobila dovoljna širina za prikaz logo oznake. Varijabla znakx definirana je pomoću minx i height, dok je varijabla znaky definirana pomoću miny i width. Pomoću varijabli znakx i znaky definirana je lokacija logo oznake koja se zatim izrezuje iz početne slike automobila.

* 1. **Detekcija ključnih točaka**

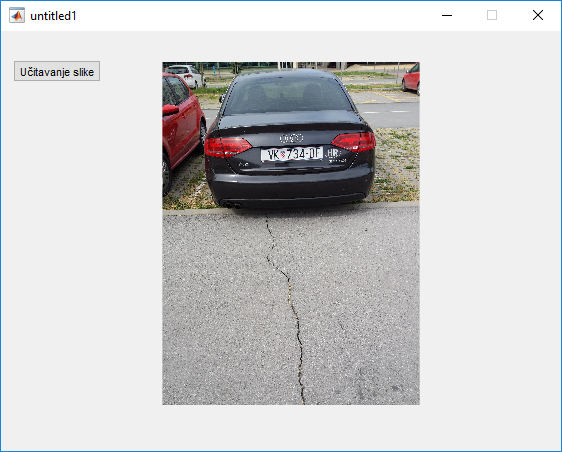
Prilikom detekcije ključnih točaka detektiraju se rubovi korištenjem SURF algoritma. Korištenjem funckcije detectSURFFeatures dobiva se objekt sa SURF karakterističnim točkama. SURF točke služe za prosljeđivanje podataka prema extractFeatures funkciji. ExtractFeatures funcija vraća vektore deskriptora i prirpadajuće im lokacije. Funkcija dobiva deskriptore iz piksela koji okružuju interesnu točku. Pikseli predstavljaju i odgovaraju značajkama određenih lokacijom jedne točke. Svaka točka predstavlja centralnu lokaciju susjednih točaka.



Slika. Prikaz pronađenih karakterističnih točaka na slici znaka automobila

* 1. **GUI**

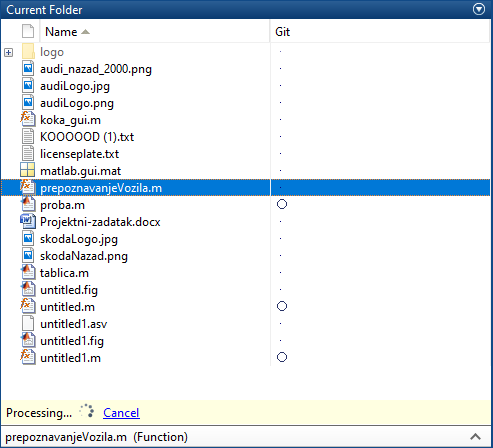
Grafičko korisničko sučelje izvedeno je pomoću tipke „Učitavanje slike“ koja otvara OpenFileManager, iz kojeg se odabire .jpg slika automobila kojemu je potrebno prepoznati marku. Nakon učitavanja slike poziva se ostatak koda iz skripte. GUI je prikazan na slici 4. Nakon što se učita slika i kod nastavi sa izvršavanjem, na GUI-u će se prikazati učitana slika u originalnoj verziji, a u novim prozorima otvorit će se prikaz značajnih točaka, te usporedbe istih.



Slika 4

## Uputstva za pokretanje

Prvo je potrebno raspakirati programski paket. Nakon toga potrebno je navigirati trenutni folder u MATLAB-u na onaj u kojem se nalaze sve raspakirane datoteke. Nakon toga potrebno je otvoriti datoteku „prepoznavanjeVozila.m“ te stisnuti pokretanje „run“ – vidljivo na slici 5.



Slika 5

U datoteci „logo“ nalaze se referentni logo-i svih marki vozila. Iz te datoteke povlače se .jpg slike s kojima se uspoređuje učitana slika.

# 3. REZULTATI I DISKUSIJA

Prilikom izrade rada korišteno je MATLAB programsko sučelje. MATLAB je programski jezik visoke razine i koristi se za matrično računanje, vizualizaciju i programiranje. Također, MATLAB je alat koji je jako zastupljen u područjima računalnog vida te se pomoću njega mogu izraditi algoritmi, aplikacije te analizirati podaci. Jednostavan je za korištenje i optimiziran za rad s matricama što nam je bilo od velikog značaja u ovome radu, budući da se slike spremaju kao matrice. Budući da je potrebno uspoređivati slike, u rad je implementiran SURF algoritam za pronalazak ključnih točaka te njihovu usporedbu. U računalnom vidu, SURF je algoritam pomoću kojeg se detektiraju tzv. „deskriptori“ odnosno ključne točke. Ovaj algoritam se također koristi prilikom prepoznavanja objekata, klasifikaciju ili 3D rekonstrukciju.

Prilikom testiranja algoritma, napravljena je baza podataka sa fotografijama automobila. Baza sadrži ukupno 100 različitih fotografija automobila te 10 referentnih slika loga automobila. Fotografije su napravljene na različitim lokacijama, snimane u različito doba dana i u različitim okolinama. Ovime je postignuto testiranje u različitim kako vremenskim, tako i prostornim uvjetima što daje dobar uvid u stvarnu primjenjivost algoritma. Budući da se na slikama nalaze automobili u prvom planu, nije bilo potrebe za njihovom detekcijom. Ključan korak u algoritmu bio je pronalazak registarske oznake kako bi dobili referencu za detekciju znaka na automobilu.

## 3.1. Detekcija registarske oznake

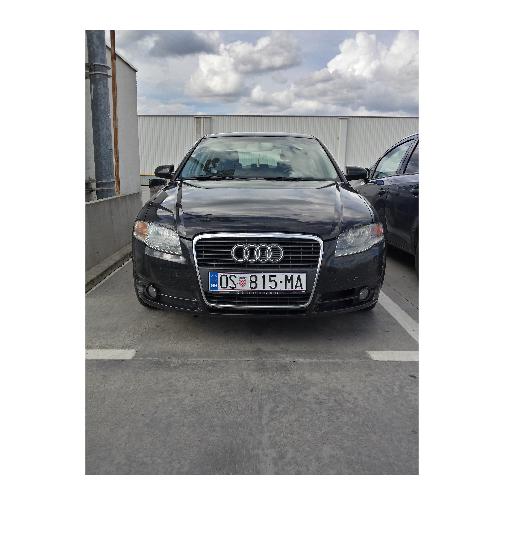
Budući da se znak automobila nalazi uvijek negdje u sredini slike, te je vrlo malih dimenzija, potrebno je pronaći način kako doći do njega. Promatrajući fotografije, može se zaključiti da je u većini slučajeva znak točno iznad njegove registarske oznake. Sukladno zaključenom, odlučili smo najprije detektirati i izdvojiti registarsku oznaku automobila. Zbog svog karakterističnog izgleda, registarsku oznako je relativno lako detektirati na vozilu. Registarska oznaka se ističe na fotografiji automobila jer sadrži izražene prijelaze i rubove koji su posljedica crnih slova na bijeloj podlozi. Algoritam za detekciju registarske oznake korišten je na cijeloj bazi automobila.

Tablica 1: Detekcija reg. oznake

|  |  |
| --- | --- |
| Ukupan broj testiranih vozila | 100 |
| Broj uspješno detektiranih reg. oznaka | 41 |
| Broj pogrešno detektiranih reg. oznaka | 57 |
| Broj reg. oznaka koje nisu detektirane | 2 |
| **Točnost algoritma** | **42.00%** |

### 3.1.1. Primjer uspješne detekcije

Algoritam odabire nekoliko mogućih kandidata za registarsku oznaku. Prilikom uspješne detekcije, algoritam ispravno određuje položaj registarske oznake te ju izdvaja iz slike. Sam proces obrade i izdvajanja tablice vidljiv je na slici ispod.



Slika 69: Prikaz uspješne detekcije registarske oznake

### 3.1.2. Primjer pogrešne detekcije

Kod pogrešne detekcije događa se to da algoritam iz nekog razloga odabere krivog kandidata za registarsku oznaku. Do krive detekcije dolazi zbog rubova na slici koji su za njegove kriterije izraženiji nego rubovi na tablici.

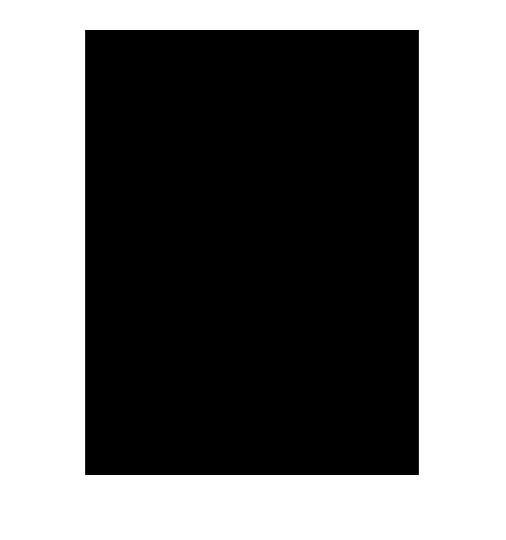


Slika 101: Prikaz pogrešne detekcije registarske oznake

### 3.1.3. Primjer registarskih oznaka koje nisu detektirane

Također postoje slučajevi u kojim algoritam ne uspijeva ništa detektirati. Događa se u slučajevima kada prema zadanim kriterijima nema rubova koji su dovoljno istaknuti da bi bili tablica.

Slika 91: Prikaz registarskih oznaka koje nisu uspješno detektirane



## 3.2. Izdvajanje znaka nakon detekcije registarske oznake

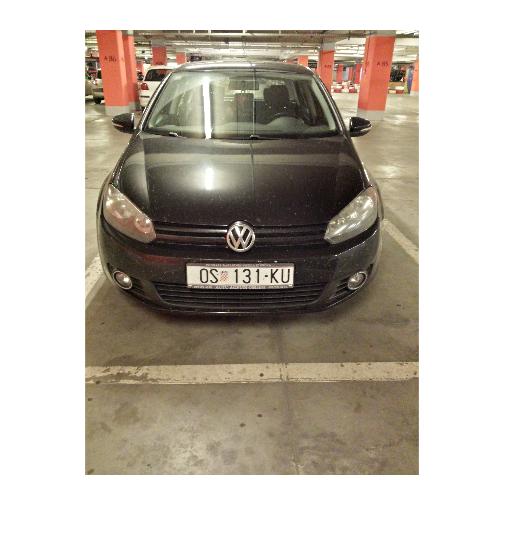
Nakon što je algoritam na određenom broju fotografija uspio definirati područje registarske oznake, slijedi izdvajanje znaka s manjim područjem oko njega kako bi imao manje mjesta za pogrešku prilikom konačne usporedbe znaka s vozila i referentnog znaka. Izdvajanje znaka vrši se pomoću tablice tako što se lokacija registarske oznake pomiče za određen broj piksela prema gore.

Tablica 2: Detekcija znaka

|  |  |
| --- | --- |
| Ukupan broj testiranih vozila | 41 |
| Broj uspješno detektiranih znakova | 25 |
| Broj pogrešno detektiranih znakova | 12 |
| Broj znakova koji nisu detektirani | 4 |
| **Točnost algoritma** | **60.97%** |

### 3.2.1. Primjer uspješne detekcije znaka

Nakon uspješne detekcije registarske oznake, algoritam vrlo uspješno izdvaja znak iz fotografije.



Slika 8: Prikaz uspješne detekcije znaka

### 3.2.2. Primjer pogrešne detekcije znaka

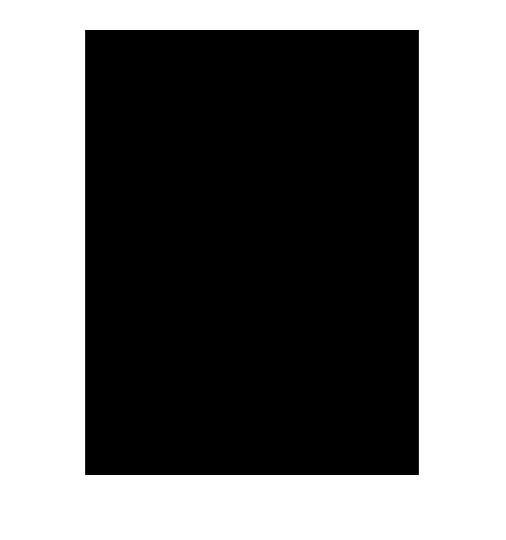
Nakon uspješne detekcije registarske oznake dolazi do pogrešne detekcije znaka. Takve pojave događaju se u našem slučaju najčešće zbog znaka koji je previsoko u odnosu na tablicu ili kad je detektirana lokacija registarske oznake obuhvatila područje izvan ruba registarske oznake. Primjer ispod pokazuje onu situaciju kada je logo pozicioniran previsoko u odnosu na registarsku oznaku.



Slika 1: Prikaz pogrešne detekcije znaka

### 3.2.3. Primjer neispravne detekcije znaka

Nakon uspješne detekcije registarske oznake ne dolazi do detekcije znaka. Razlozi mogu biti različiti



Slika 2: Prikaz neuspješne detekcije znaka