SVEUČILIŠTE U ZAGREBU

FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 1486

SUSTAV ZA SEGMENTACIJU SLIKA PROMETNIH ZNAKOVA NA TEMELJU BOJE

Toni Benussi

Zagreb, lipanj 2010.

Zahvala

Zahvaljujem se mentoru doc. dr. sc. Zoranu Kalafatiću na pruženoj pomoći i savjetima koji su bili od velike pomoći u izradi ovog završnog rada. Također sam mu zahvalan jer mi je omogućio da samostalno donosim većinu važnih odluka vezanih uz realizaciju ovog zadatka, zbog čega sam naučio nove načine pristupa i rješavanja problema. Zahvaljujem se i svim autorima literature i algoritama koje sam koristio, jer su svoje radove i dostignuća učinili javno dostupnima za korištenje.

Sadržaj

1. Uvod

Računalni vid jedna je od važnijih disciplina u području umjetne inteligencije, koja se bavi izradom sustava kojima je cilj izvlačenje informacija iz slika. Pritom se ne misli samo na slike u doslovnom smislu riječi, već na cijeli skup vizualnih izvora informacija kao što su na primjer video zapis, višedimenzionalni prikaz dobiven iz medicinskog skenera i slično. Kako je većina informacija koje čovjek dobiva iz svoje okoline upravo vizualnog tipa, odmah postaje jasno da je računalni vid jedan od temeljnih smjerova u razvoju računala koje bi oponašalo ljudski način razmišljanja i djelovanja, te moglo u određenim zadaćama uspješno zamijeniti čovjeka.

Računalni vid se razvio relativno kasno kao računarska disciplina, prvenstveno zbog ograničenosti procesorske moći računala, te su se tek u zadnje vrijeme tehnike iz ovog područja počele češće primjenjivati u praktične svrhe. Za sada se nije uspio stvoriti sustav koji uspješno nadomješćuje čovjekovu percepciju u cjelini i u općenitoj situaciji, ali su se uspjeli izraditi specijalizirani sustavi koji u određenim usko specijaliziranim područjima ponekad čak i nadmašuju čovjeka (pronalaženje tumora na medicinskim slikama, prepoznavanje i rekonstruiranje oštećenih slika i slično).

Segmentacija slike jedan je od postupaka koji se koriste u kontekstu računalnog vida, a njena svrha je dijeljenje slike u segmente (dijelove slike koji sadrže piksele koji su po nekoj karakteristici slični i mogu se zajedno grupirati). Glavne zadaće segmentacije su pojednostavljenje ili općenito transformiranje slike, koje omogućava lakšu analizu slike koja se obrađuje, te se dobiva bolja interpretacija informacija na slici. Također segmentacija slike može poslužiti za izdvajanje samo onih dijelova slike koji su zanimljivi za obradu i time smanjiti količinu podataka koje treba dalje obrađivati, čime se može znatno uštedjeti na vremenu.

Tehnike segmentacije slike mogu se podijeliti u dvije osnovne skupine:

- Tehnike koje traže istovjetna područja na slici. Takve se tehnike dalje dijele na one koje koriste segmentaciju na temelju amplituda određenih karakteristika piksela (boja, svjetlina, tekstura), najčešće prikazanih u obliku histograma, te na tehnike koje koriste metode koje grupiraju piksele (clustering), metode širenja regija i spajanja istovjetnih područja

- Druga skupina tehnika ne traži cijela područja već se fokusira na određivanje granica između susjednih, ali različitih područja, te se na temelju dobivenih granica utvrđuje koji dijelovi slike pripadaju pojedinom segmentu

Algoritam koji je korišten u ovom radu pripada prvoj od ove dvije skupine. Riječ je o algoritmu K srednjih vrijednosti koji će biti opisan u nastavku.

Općenito gledano vrlo je teško izraditi kvalitetan sustav za segmentaciju slike koji će davati dobre rezultate bez obzira na područje primjene, pa kako je u ovom slučaju područje primjene ograničeno na prometne znakove ideja je bila stvoriti algoritam koji će upravo u tom segmentu davati zadovoljavajuće rezultate, a ujedno i imati dovoljno dobre performanse da bude primjenjiv u praksi.

Ideja primjene segmentacije slike na temelju boje proizlazi iz činjenice da su boje jedna od temeljnih značajki koje služe za raspoznavanje prometnih znakova kod ljudi, pa je logično kromatske karakteristike znaka pokušati iskoristiti i kod detekcije i raspoznavanja prometnih znakove od strane računala. Međutim raspon intenziteta i tonaliteta u kojem se određena boja javlja na prometnim znakovima je vrlo velik, djelomično zbog boje samih znakova, a ponajviše zbog različitih uvjeta osvjetljenja, kvalitete slike, udaljenosti znaka na slici i slično. Zato ideja ovog rada je da se odredi skup boja koje se javljaju na prometnim znakovima, kako bi se takva informacija mogla iskoristiti u drugim sustavima koji obrađuju slike prometnih znakova.

U nastavku ovog rada također će biti opisan jedan jednostavan test koji primjenjuje postupak segmentacije s ciljem utvrđivanja da li se na slici nalazi prometni znak, te se razmatra potencijalna korisnost takvog sustava kao potpore nekom drugom sustavu koji se bavi raspoznavanjem prometnih znakova.

2. Algoritam K srednjih vrijednosti

2.1. Općeniti algoritam K srednjih vrijednosti

Algoritam K srednjih vrijednosti je algoritam koji se koristi na području statistike, strojnog učenja, računalnog vida i drugih područja kod kojih se javlja potreba za generaliziranjem i pojednostavljivanjem velikog broja podataka koje treba analizirati i interpretirati. Temeljna ideja algoritma je da se skup od *n* zadanih pojava podijeli na *k* grupa (eng. clusters) , tako da se svakoj pojavi dodijeli ona grupa čija je srednja vrijednost po nekom kriteriju najsličnija danoj pojavi. Upravo od te ideje podjele na grupe na temelju sličnosti sa njihovim srednjim vrijednostima dolazi i naziv algoritma.

Prvu ideju takvog algoritma imao je još 1956. Hugo Steinhaus, dok je standardni algoritam predložio 1957. Stuart Lloyd, kao tehniku za pulsno-kodnu modulaciju, ali nije objavljen sve do 1982. godine. Kako vidimo algoritam prethodi problemu segmentacije slike pomoću računala, ali unatoč tome pokazao se vrlo uspješnim u toj zadaći.

Kako bi se uopće mogao provoditi algoritam potrebno je da pojave sadrže skup atributa koja se mogu kvantificirati i prikazati u *n*-dimenzionalnom Euklidskom prostoru kako bi se vrijednosti mogle međusobno uspoređivati i prema tome odrediti kojoj od *k* skupina će biti pridijeljena koja pojava. Cilj algoritma je dobiti takve grupe u kojima je varijanca između pojava unutar svake grupe što manja.

Kao i većina algoritama takvog tipa algoritam K srednjih vrijednosti je iterativni algoritam koji se sastoji od sljedećih koraka:

1. Odabire se *k* početnih srednjih vrijednosti. To se može učiniti slučajnim odabirom ili pomoću neke heuristike.
2. Svaka se pojava pridjeljuje grupi s onom srednjom vrijednosti koja je u Euklidskom prostoru najbliža vrijednosti atributa te pojave.
3. Kad se sve pojave rasporede u grupe računaju se nove srednje vrijednosti svake grupe na temelju vrijednosti atributa pojava pridijeljenih svakoj od tih grupa.
4. Koraci 2. i 3. se ponavljaju sve dok niti jedna pojava (ili zanemarivo mali udio pojava) ne promijeni grupu između dvije iteracije algoritma.

Matematički opisano, za skup srednjih vrijednosti $m_1^{(1)}, m_2^{(1)},..., m_k^{(1)}$ gdje eksponent predstavlja iteraciju algoritma, a indeks redni broj srednje vrijednosti, vrijedi sljedeće pravilo o pridjeljivaju pojava grupama $S_1,..., S_k$:

$$S_i^{(t)} = \left\{ \mathbf{x}_j : \left\| \mathbf{x}_j - \mathbf{m}_i^{(t)} \right\| \leq \left\| \mathbf{x}_j - \mathbf{m}_{i^*}^{(t)} \right\| \text{ for all } i^* = 1, \ldots, k \right\}$$

tj. Svaka pojava x_j se pridjeljuje onoj grupi za koju vrijedi da je udaljenost pojave x_j od srednje vrijednosti $m_i^{(t)}$ manja ili jednaka udaljenosti pojave x_j od srednjih vrijednosti svih ostalih grupa.

Nakon pridjeljivanja računaju se nove srednje vrijednosti na sljedeći klasičan način:

$$\mathbf{m}_i^{(t+1)} = \frac{1}{|S_i^{(t)}|} \sum_{\mathbf{x}_j \in S_i^{(t)}} \mathbf{x}_j$$

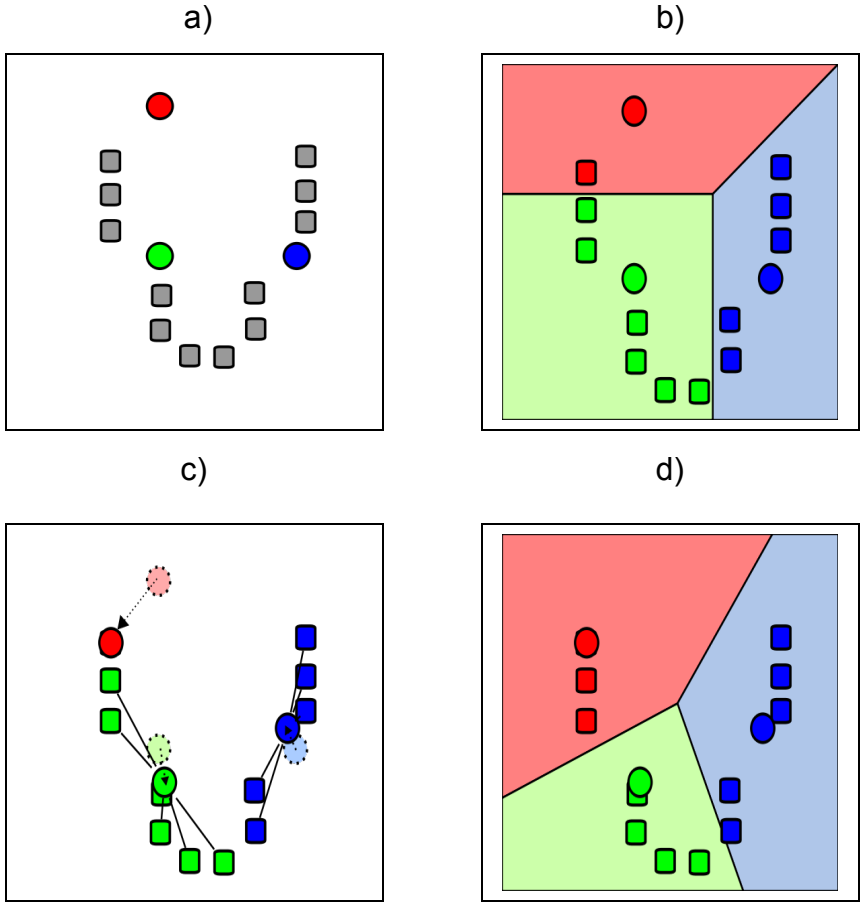
Uvjet izlaska iz iterativnog postupka je kad za svaku pojavu x_j vrijedi da je u iteraciji *t* i u iteraciji *t+1* pridijeljena istoj grupi.

Algoritam ne teži nužno globalnom optimumu raspodjele pojava u grupe, jer podjela ovisi o izboru početnih srednjih vrijednosti, te uopće o broju srednjih vrijednosti.

Po pitanju složenosti algoritam K srednjih vrijednosti je NP-težak problem, a ako su unaprijed poznati dimenzionalnost *d* Euklidskog prostora atributa pojava i broj srednjih vrijednosti *k* onda se složenost može egzaktno definirati sljedećom formulom $O(n^{dk+1} \log n)$. Iako je očito da je apriorna složenost algoritma vrlo velika, pokazalo se da brzo konvergira prema stabilnom stanju (uvjetu prestanka iteriranja), posebice ako je izbor početnih srednjih vrijednosti segmenata dobar, osim u specifičnim slučajevima kada dolazi do titranja između dvije ili više podjela skupa pojava na segmente, što se jednostavno može riješiti ograničavanjem broja iteracija na neku fiksnu vrijednost za koju se smatra da je dovoljno velika da će algoritam u velikoj većini slučaja dati dobre rezultate, ili izmjenom uvjeta izlaska iz petlje (npr. Ako je broj pojava koji je promijenio segment u zadnjoj iteraciji manji od nekog broja *n* ili nekog postotka *p* onda se iteriranje zaustavlja).

Glavni nedostaci upravo opisanog osnovnog algoritma su činjenica da je broj srednjih vrijednosti unaprijed zadan parametar, što je u konkretnoj implementaciji izbjegnuto uporabom heurističkog postupka u izboru srednjih

vrijednosti. Drugi nedostatak je uporaba Euklidske udaljenosti kao kriterija za određivanje pripadnosti pojedinoj grupi jer se u mnogim primjenama takav kriterij pokazao nedostatnim za kvalitetnu segmentaciju.



Slika 1: Primjer jedne iteracije algoritma K srednjih vrijednosti

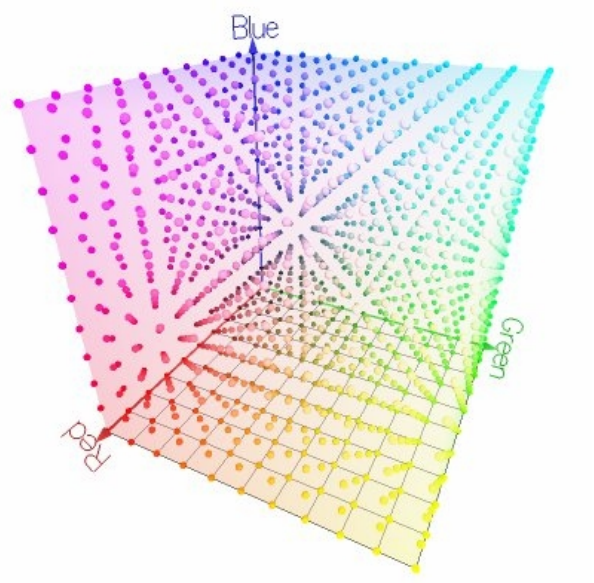
Na gornjoj slici je prikazana jedna iteracija algoritma K srednjih vrijednosti. Prvo se odaberu početne srednje vrijednosti (a). Nakon toga se odredi podjela na segmente na temelju tih početnih srednjih vrijednosti (b). Pomoću vrijednosti točaka pridruženih pojedinom segmentu računa se nova srednja vrijednost tog segmenta (c). Na kraju se vrši nova podjela na segmente na temelju novih srednjih vrijednosti koje smo dobili u prethodnoj iteraciji (d).

2.2. Primjena algoritma K srednjih vrijednosti za segmentaciju slike na temelju boje

Kada govorimo o segmentaciji slike na temelju boje moramo voditi računa o dva prostora, jedan je prostor slike, koji predstavlja dvodimenzionalno polje piksela koji čine tu slike, a drugi prostor je prostor boja. Dimenzionalnost i oblik prostora boja ovise o načinu na koji su boje prikazane na slici.

Prilikom izrade implementacije za ovaj rad korišten je RGB prostor boja, jer je to prostor boja u kojemu su prikazane boje u većini formata u kojima se slike prikazuju. To je ujedno i način na koji se boje prikazuju na zaslonu računala, a uz to lako je interpretirati RGB vrijednosti i povezati ih sa bojom koju predstavljaju.

Ideja RGB načina prikaza boja je da se boje prikazuju pomoću tri komponente, crvene (R), zelene(G) i plave(B), a rezultirajuća boja dobije se kao zbroj vrijednosti te tri komponente. Kombiniranjem različitih odnosa komponenti moguće je dobiti većinu boja koje ljudsko oko može vidjeti, od crne koja je definirana tako da su sve tri komponente jednake nuli, do bijele koju dobivamo kad sve komponente postižu maksimalnu vrijednost.



Slika 2: RGB prostor boja

Kako je RGB prostor u stvari samo Euklidski trodimenzionalni prostor u kojem svaka os predstavlja jednu od tri osnovne boje, onda se i za uspoređivanje vrijednosti u tom prostoru kao mjerilo može koristiti Euklidska udaljenost u trodimenzionalnom prostoru koja se računa na sljedeći način:

$$d_{a,b} = \sqrt{(R_a - R_b)^2 + (G_a - G_b)^2 + (B_a - B_b)^2}$$

Ova jednakost se kod algoritma K srednjih vrijednosti koristi da bi se odredio segment kojem će biti pridružen pojedini piksel tako da se on pridruži segmentu *j* čija je srednja RGB vrijednost $sred_j$ najbliža RGB vrijednosti piksela za kojeg određujemo segment kojem pripada:

$$S_j = \left\{ pixel_i : d_{i,sred_j} <= d_{i,sred_n}, n = 1..k \right\}$$

Valja naglasiti da algoritam definiran sa navedenim kriterijem raspodjele piksela po segmentima ne vodi računa o fizičkoj udaljenosti piksela na slici, te se time ne stvaraju segmenti koji čine jednu uniformnu regiju na slici, međutim to se nije pokazalo presudnim u konkretnoj implementaciji, a ovako jednostavnim kriterijem usporedbe značajno se ubrzao rad algoritma.

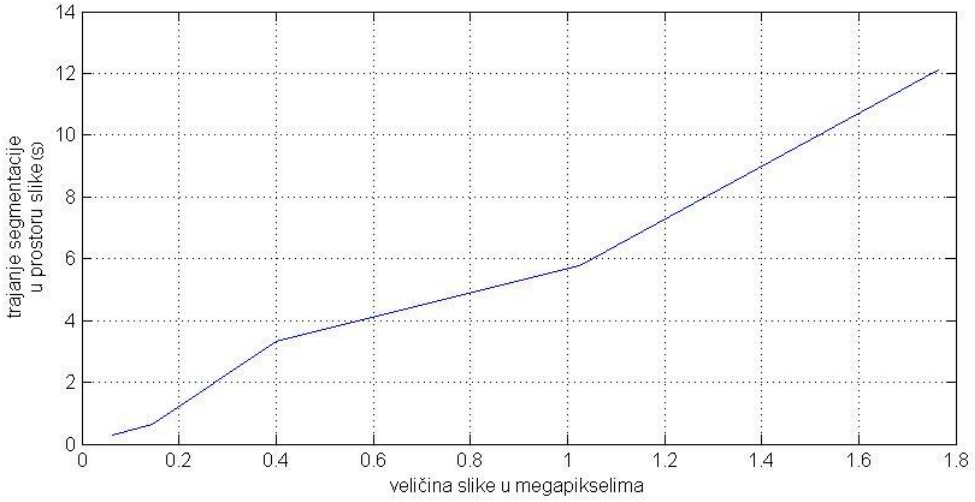
Prilikom implementiranja ovog algoritma pojavila su se dva moguća pristupa problemu, koja su direktno povezana sa već spomenutim dva prostora (prostor slike i prostor boja) koji su relevantni za postupak segmentacije. Prvi pristup je da se svakom pikselu slike na temelju njegove boje odredi segment kojem pripada. Takav postupak implicira da se svakom pikselu slike pristupi barem jednom prilikom svake iteracije algoritma, što rezultira jako velikim ukupnim brojem pristupa memoriji što automatski za sobom povlači i niske performanse sustava segmentacije. Međutim takav pristup ima i određene prednosti, a to su jako lako stvaranje segmentirane slike, jer postoji direktna veza između svakog piksela i segmenta kojemu taj piksel pripada. Također prostor slike (ukupni broj piksela slike) je puno manji od prostora boja (ukupan broj različitih boja koje je moguće prikazati u RGB sustavu), pa se na prvi pogled višestruko prelaženje kroz cijeli prostor slike čini puno ekonomičnija solucija nego višestruko prolaženje kroz prostor boja. Na primjer uobičajena veličina slike je 800x600 piksela što rezultira prostorom slike od $4.8*10^5$ piksela, a u konkretnoj implementaciji segmentiraju se prometni znakovi koji rijetko prelaze veličinu od 50x50 piksela što rezultira prostorom slike od samo $2.5*10^3$ piksela, što je više nego prihvatljiva veličina. S druge strane RGB prostor najčešće sadrži vrijednosti od 0 do 255 za svaku komponentu što rezultira prostorom boja od $1.67*10^7$ boja što je za nekoliko reda veličine više od prostora slike.

Drugi pristup je upravo taj da se, umjesto raspoređivanja piksela slike po segmentima, segmenti stvaraju u prostoru boja tako da se svaka boja grupira u segment čija je RGB srednja vrijednost najbliža toj boji, a tek nakon što iteriranje završi, na temelju originalne boje piksela, određuje se koja se srednja vrijednost pridjeljuje pojedinom pikselu. Uzimajući u obzir veličinu prostora slike i kompleksniji način pridjeljivanja nove boje pikselima ova se metoda u teoriji čini mnogo sporijom. Međutim valja naglasiti da svaka slika sadrži samo mali dio ukupnog skupa boja kojeg sadrži RGB prostor, a često su slike zadane sa samo 256 različitih vrijednosti boje iz cijelog RGB prostora pomoću tzv. look-up tablice (LUT).

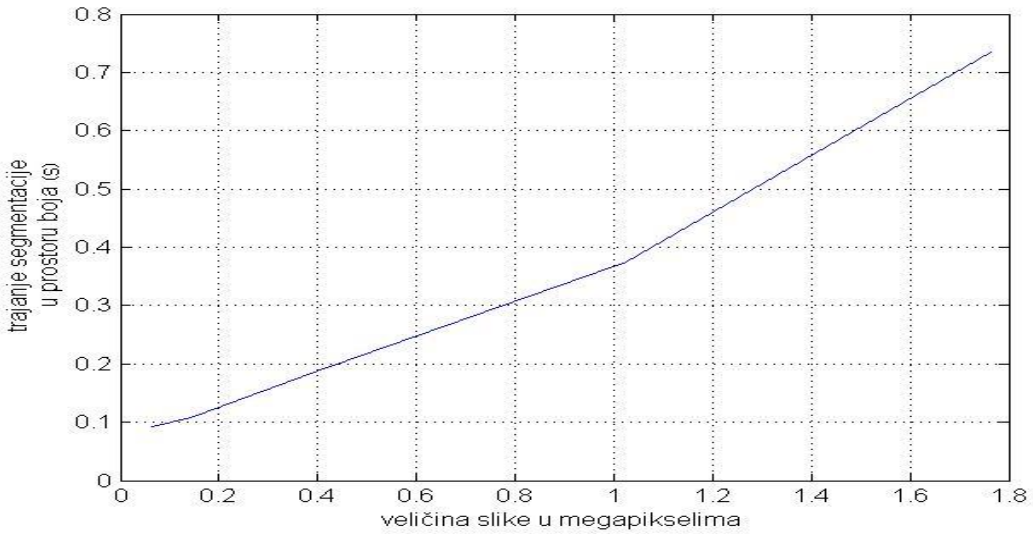
Slika 3: Usporedba slike sa 256 boja (lijevo) i slike sa 16.7 milijuna boja (desno)

Sa slike 3 se vidi da je razlika u kvaliteti reprezentacije boja kad se koristi samo 256 boja vrlo mala naspram slučaja kad se koristi cijeli RGB prostor boja, a računajući da će segmentirana slika imati daleko manje boja od originala, ta razlika praktički ne predstavlja nikakvo ograničenje.

Također brzina pristupa tablici boja je daleko veća od brzine pristupa pikselima slike zbog daleko jednostavnije strukture podataka koja se koristi za reprezentaciju boja od strukture podataka koja sadrži sliku. Druga velika prednost ovakvog sustava je potreba da se slici pristupa svega 2 puta prilikom segmentacije. Prvim pristupom se u prostor boja mapiraju one boje koje slika sadrži, a drugi pristup je potreban kako bi se stvorila segmentirana slika. Ovako mali broj pristupa slici povlači jednu vrlo značajnu karakteristiku algoritma, a to je relativno mala ovisnost brzine algoritma o veličini slike koja se segmentira, što čini algoritam vrlo pogodnim i kod segmentacije većih slika.



Slika 4: Vremenske performanse segmentacije slike u prostoru slike



Slika 5: Vremenske performanse segmentacije slike u prostoru boja

Sa slika 4 i 5 vidi se omjer performansi segmentacije u prostoru slike i segmentacije u prostoru boje. Oba grafa su dobivena segmentacijom iste slike u različitim rezolucijama od maksimalno 1.764 megapiksela do minimalno 0.064 megapiksela. U prostoru slike ovo smanjenje rezolucije od oko 28 puta dovelo je do ubrzanja segmentacije za 43 puta, dok je kod segmentacije u prostoru boja ubrzanje bilo oko 8 puta. Ovaj primjer potvrđuje da je segmentacija u prostoru boja daleko efikasnija, a razlika u performansama između te dvije metode raste proporcionalno sa veličinom slike (u konkretnom primjeru segmentacija u sustavu boja je oko 3 puta brža za najmanju rezoluciju, a oko 16 puta brža za najveću rezoluciju).

2.3. Izbor početnih srednjih vrijednosti segmenata

Kako je već spomenuto u opisu originalnog algoritma K srednjih vrijednosti, izbor početnih srednjih vrijednosti ima presudan utjecaj na kvalitetu segmentacije. Prilikom izbora srednjih vrijednosti javljaju se 2 osnovna problema, a to su broj segmenata na koji će slika biti podijeljena i početne srednje vrijednosti svakog od tih segmenata.

U osnovnoj inačici algoritma broj segmenata je parametar kojeg određuje korisnik prije pokretanja algoritma segmentacije ili se može koristiti predefinirani broj segmenata. Prvi slučaj čini algoritam vrlo nepogodnim za bilo kakvu primjenu algoritma u sustavu koji zahtijeva da se segmentacija obavlja automatski kao dio nekog drugog postupka, te ovisi o znanju i iskustvu korisnika koji mora znati adekvatno procijeniti broj segmenata koji bi odgovarao pojedinoj slici. S druge strane predefinirani broj segmenata omogućava automatizaciju segmentacije, ali postižu se vrlo loši rezultati segmentacije ako broj segmenata nije odgovarajući za sliku koju se obrađuje.

Slika 6: Primjer loše segmentacije sa fiksnim brojem (5) srednjih vrijednosti

Sa slike 6 se jasno vidi da je premali unaprijed definiran broj segmenata doveo do značajnog gubitka informacija na slici. Posebno valja istaknuti kompletan gubitak vrlo izražene crvene boje okruglog prometnog znaka, što je posebno značajno u kontekstu ovog rada budući da ovakva segmentacija čini boju apsolutno beznačajnim kriterijem za detekciju i identifikaciju prometnog znaka.

Kako bi se ipak postigla automatizacija procesa segmentacije, uz zadovoljavajući izbor broj segmenata i njihovih početnih srednjih vrijednosti, potrebno je primijeniti neku heurističku metodu koja će na temelju podataka koje

slika sadrži moći uspješno procijeniti odgovarajuće početne parametre procesa segmentacije.

Algoritam koji se koristi za određivanje srednjih vrijednosti u ovoj implementaciji sastoji se od sljedećih koraka:

1. Napravi se histogram boja tako da se za svaku boju na slici odredi broj piksela te boje
2. Kao prva srednja vrijednost odabire se ona boja koja je najzastupljenija na slici
3. Sljedeća boja, koja se odabire kao srednja vrijednost, je ona boja koja ima najveću minimalnu udaljenost od svih postojećih srednjih vrijednosti, a postoji barem jedan piksel te boje
4. Odabir srednjih vrijednosti završava kad je ustanovljena najveća minimalna udaljenost za sljedeću potencijalnu srednju vrijednost manja od praga koji je zadan kao parametar algoritma

Ovakav način odabira srednjih vrijednosti ima višestruku prednost naspram originalnog algoritma. Prva očigledna prednost je da se treba zadati samo jedan parametar, a to je prag za najveću minimalnu udaljenost za sljedeću srednju vrijednost. U konkretnoj implementaciji kako bi se izbjegla skupa operacija korjenovanja izračunava se kvadrat udaljenosti u RGB prostoru, pa se prema tome i za pragove trebaju također uzimati kvadrati željenih minimalnih udaljenosti za novu srednju vrijednost. Tim se parametrom određuje koliko će „gruba“ biti segmentacija, što je prag niži to će biti više segmenata, a samim time će boje na segmentiranoj slici vjerodostojnije odražavati boje originalne slike. Druga prednost ovakvog postupka je da će broj segmenata direktno ovisiti o varijetetu boje na slici, tj. slika sa velikim brojem različitih boja će imati više segmenata, čime se bitno smanjuje vjerojatnost da se neka boja izgubi prilikom segmentacije. Ovako odabrane početne srednje vrijednosti segmenata su ravnomjerno raspoređene među bojama koje se nalaze na slici, što pogoduje bržem konvergiranju algoritma K srednjih vrijednosti.

Još jedna dobra strana ovog načina izbora početnih srednjih vrijednosti je da se stvara histogram boja koji će se koristiti i u samoj segmentaciji čime se uštedjelo na vremenu.

Slika 7: Segmentacija slike sa različitom vrijednostima praga
(gore lijevo: original, gore desno: prag 200, dolje lijevo: prag 500, dolje desno: prag 1000)

3. Primjena segmentacije slike na temelju boje u detekciji prometnih znakova

Primjena boja u detekciji prometnih znakova je direktna posljedica činjenice da su boje na prometnim znakovima dobro definirane i vrlo upadljive, te se u većini slučaja značajno razlikuju od boja okoline koja okružuje taj znak. Najjednostavniji pristup bio bi jednostavno uspoređivati boje slike sa bojama prometnih znakova. Međutim problem nastaje kada se takav sustav primjenjuje u realnoj situaciji gdje je čest slučaj da su, zbog loše kvalitete slike i uvjeta u kojima je slika snimljena, odstupanja boja znakova na slici od pravih boja tih znakova vrlo velika. Zbog toga je potrebno stvoriti bazu informacija o bojama koje znakovi poprimaju na skupu primjera iz stvarnog života, kako bi se nove slike prometnih znakova mogle uspoređivati sa podacima iz baze jer postoji velika mogućnost da, ako je baza dovoljno velika, u njoj postoje podaci dobiveni na temelju slika koja su snimljene u sličnim uvjetima kao slika koju analiziramo.

Slika 8: Primjer različitosti idealnog prometnog znaka sa pravom slikom

Slika 8 dobro ilustrira opisani koncept različitosti slike znaka iz stvarnog života naspram idealne slike. Posebice je značajno uočiti razliku u crvenom rubu znaka, koji je u idealnoj slici vrlo upadljive crvene boje, dok na realnoj slici rub teži ka daleko tamnijoj, gotovo sivo-smeđoj boji.

3.1. Korištena baza slika

Za izradu sustava za detekciju znakova na temelju boje korištena je baza slika iz projekta Mastif (Mapping and Assessing the State of Traffic Infrastructure). Baza se sastoji od skupa slika dobivenih iz video zapisa snimljenih iz automobila na kojima su označene pozicije prometnih znakova. Slike uglavnom prikazuju okrugle i trokutaste znakove, te žute table. Također za testiranje ispravnosti sustava baza sadrži i skup pozadina koje ne sadrže prometne znakove kako bi se mogla testirati otpornost sustava na lažne pozitivne detekcije (false positive).

Detalji o konkretnom broju različitih uzoraka pojedine vrste u bazi bit će navedeni kasnije prilikom izrade konkretnih statistika temeljenih na njima.

3.2. Stvaranje histograma boja skupa znakova i način primjene na detekciju znakova

Prvi korak u primjeni rezultata segmentacije slike u detekciji prometnih znakova je stvaranje histograma koji sadrži statistiku o svim srednjim vrijednostima segmenata koje su se pojavljivale na segmentiranim slikama prometnih znakova, te zapisivanje podataka o učestalosti pojavljivanja pojedine boje. Također treba isti takav histogram izraditi za pozadine kako bi se mogla uspoređivati sličnost dobivene slike i sa znakom i sa pozadinom.

Postoji više statističkih informacija koje se mogu analizirati prilikom izrade takvog histograma. Najjednostavnija među njima je udio pojedine boje u ukupnom skupu vrijednosti koje su dobivene kao rezultat obrade skupa znakova za učenje.

$$udio(boja_i) = \frac{brojPiksela(boja_i)}{\sum_j brojPiksela(boja_j)}$$

Ovakva statistika se pokazala neefikasnom jer je postotni udio svake boje vrlo mali budući da se na različitim slikama pojavljuje puno vrlo sličnih nijansi svake boje, pa bi primjena takvih histograma boja na pojedine slike zahtijevala skaliranje takvih vrijednosti, što nije praktično i efikasno. Drugi nedostatak takvog histograma je da veće slike znakova (koje imaju više piksela) imaju puno veću težinu u formiranju histograma, pa se time dodatno pogoršava detekcija malih znakova, koji su često lošije kvalitete, a samim time problematičniji za detektiranje.

Zato se za izradu histograma koji su se koristili u detekciji, umjesto apsolutnog udjela boje u cijelom histogramu, koristio prosječni relativni udio boje na onim slikama na kojima se ta boja pojavljuje. Ovakva informacija se onda može direktno koristiti za usporedbu sa pojedinim slikama kod testiranja, a budući da se radi o relativnim udjelima i mali i veliki znakovi jednako sudjeluju u stvaranju histograma.

Također jedna modifikacija ovako dobivenih histograma, koja je korištena prilikom testiranja, je i varijanta koja svaku vrijednost boje računa kao prosjek svih vrijednosti svoje 3x3x3 okoline u RGB prostoru koje postoje u histogramu. Tim se

postupkom dobiva „razmazaniji“ histogram koji ima veću toleranciju na male razlike u nijansama boja.

$$udio(R_i, G_i, B_i) = \frac{1}{n} \sum_{r=R_i-1}^{R_i+1} \sum_{r=G_i-1}^{G_i+1} \sum_{r=B_i-1}^{B_i+1} udio(r, g, b)$$

Gdje *n* predstavlja broj vrijednosti u okolini te boje.

Ovako dobiveni histogrami se potom koriste prilikom detekcije znakova na sljedeći način:

1. Segmentira se zadana slika i izračunaju srednje vrijednosti segmenata na toj slici
2. Izračuna se suma kvadrata razlika u udjelu svake dobivene vrijednosti na slici sa udjelom koji je za tu boju pohranjen u histogramu znakova i u histogramu pozadina. Ako ne postoji ta boja u histogramu za razliku se koristi najveća moguća razlika u udjelu a to je 1.

$$\Delta znakovi = \sum_k \left[udioSlika(segment_k) - udioStatistikaZnakova(segment_k) \right]^2$$

$$\Delta pozadine = \sum_k \left[udioSlika(segment_k) - udioStatistikaPozadine(segment_k) \right]^2$$

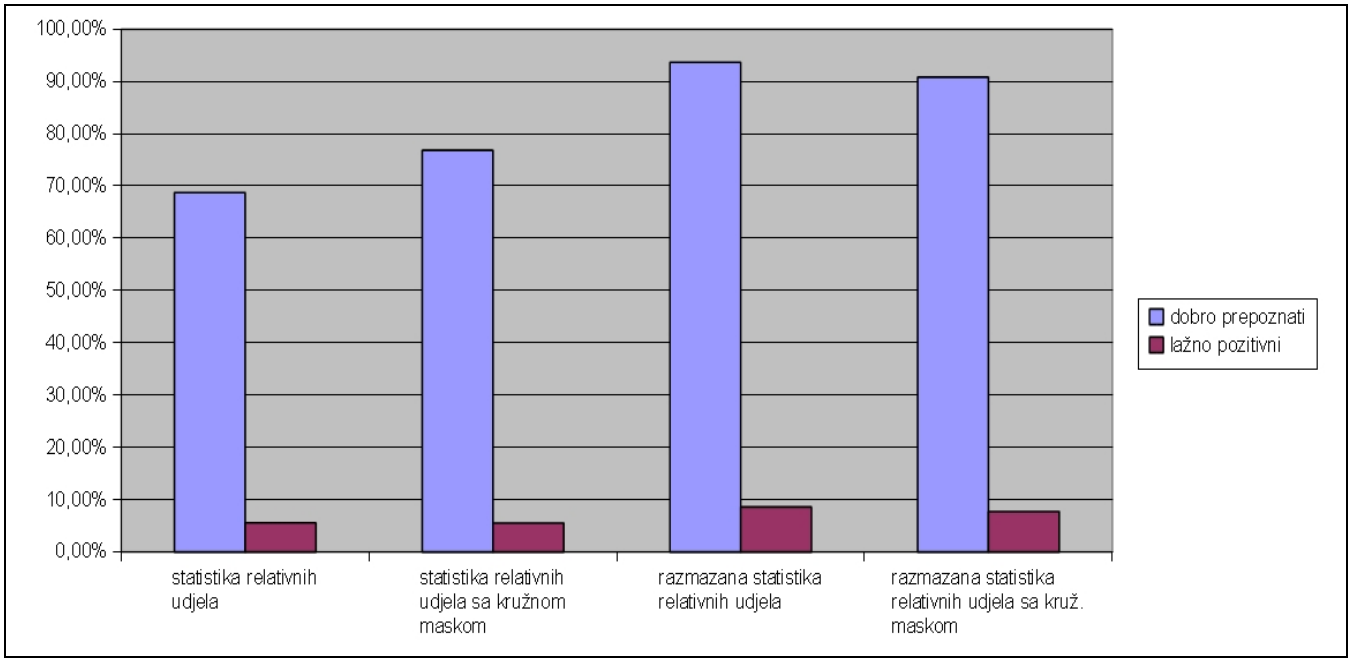
3. Ako je ukupna suma razlika između slike i histograma znakova manja od ukupne sume razlika slike i histograma pozadina onda se slika klasificira kao znak, ako ne onda se klasificira kao pozadina.

$$jeZnak = \begin{cases} true, & \Delta \text{ znakovi} < \Delta \text{ pozadine} \\ false, & \text{inače} \end{cases}$$

3.3. Rezultati testiranja za okrugle znakove

Učenje za okrugle znakove obavljeno je na skupu od 753 slike rezolucije 720x576, koje sadrže oko 800 označenih prometnih znakova različitih veličina. Na temelju tog skupa dobio se histogram koji sadrži ukupno 4125 različitih vrijednosti boja s odgovarajućim udjelima.

Histogram pozadina stvoren je tako da su se iz skupa od 713 slika sa pozadinama nasumično izrezivali dijelovi različitih veličina (kvadrati sa duljinom stranice između 15 i 50 piksela) koji su se potom segmentirali i na temelju njih je izrađena statistika pozadina. Odabran je ovakav pristup jer je to varijanta koja daje slične uzorke krivom označavanju znakova na slici, a omogućava da se jedna slika više puta iskoristi za izradu statistike. Ukupan broj dobivenih vrijednosti u histogramu pozadina iznosi 6273.



Slika 9: Rezultati testiranja za okrugle znakove

Sa slike vidimo da rezultati testiranja nad okruglim znakovima u velikoj mjeri ovise o tipu histograma koji se koristi. Obični histogram relativnih udjela boja na slikama ne daje dobre rezultate; broj uspješno prepoznatih znakova je samo 68.74%, dok je udio lažnih detekcija prihvatljivih 5.56%. Razlog ovako loših performansi valja tražiti u činjenici da je učenje okruglih znakova učinjeno na svega oko 750 primjera, dok se prilikom učenja pozadina sa svake slike uzelo između 5 i 15 prozorčića, što daje barem 5000 uzoraka, čime je statistika pozadina

znatno bogatija informacijama, a samim time vjerojatnost da se neka boja nađe na tako dobivenom histogramu je veća.

Opravdanost ovakvog razmatranja je potvrđena rezultatima razmazanog histograma, kod kojeg se vidi dramatičan porast u uspješnom prepoznavanju znakova (93.64% uspješno prepoznatih, uz porast lažnih detekcija za samo 3%). Razlog tome je upravo činjenica da je siromašniji originalni histogram okruglih znakova jako profitirao od takvog „proširivanja“ vrijednosti, jer su se time pokrile male razlike u nijansama boja koje originalni histogram nije obuhvaćao.

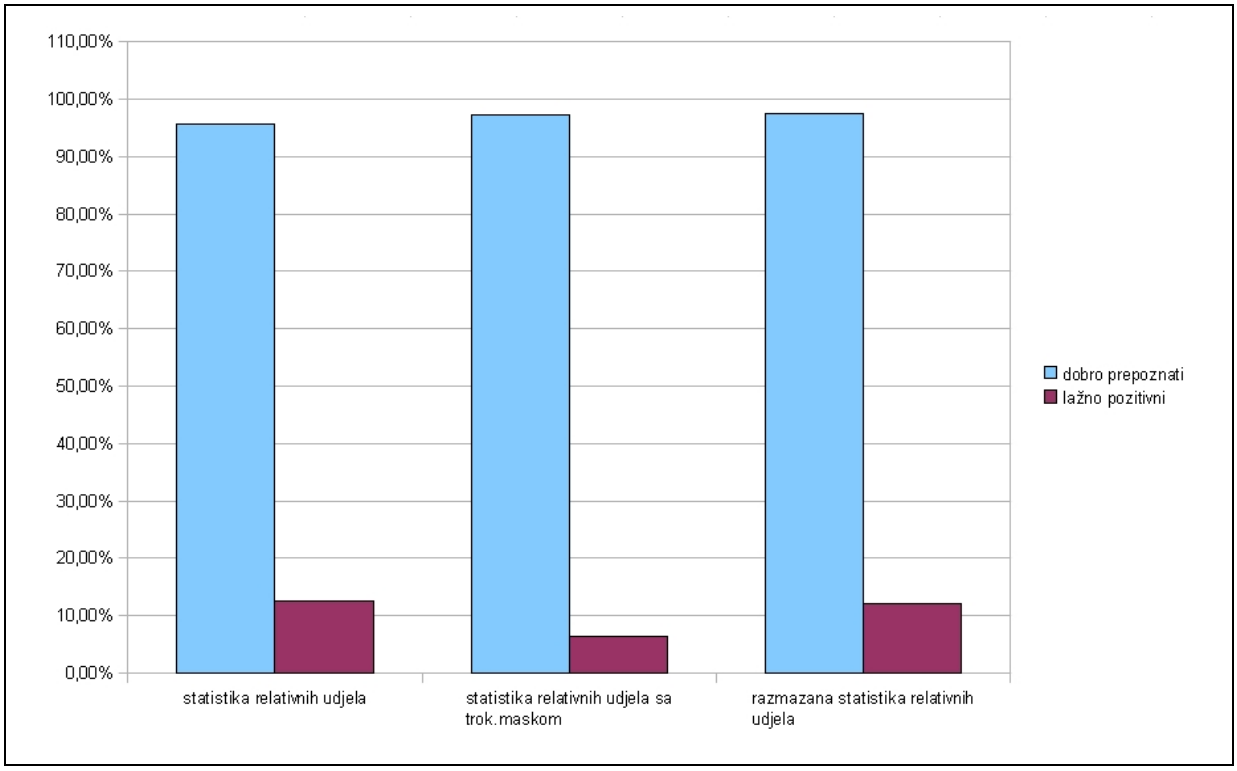
Poboljšanje performansi detekcije znakova pokušalo se dobiti i primjenom kružne maske na sliku prije segmentacije, tako da se segmentira samo dio slike koji obuhvaća znak.

Slika 10: Primjeri uporabe kružne maske kod segmentacije

Ovom se maskom iz histograma znakova eliminira dio boja koje pripadaju pozadini oko znaka, čime bi se trebala poboljšati uspješnost detekcije. Kod obične razdiobe primjena maske je rezultirala povećanjem uspješne detekcije koja s maskom iznosi 76.83% (poboljšanje za 8%) dok je broj lažnih detekcija ostao skoro nepromijenjen. Neočekivano, kad se na histogram dobiven maskom pokuša primijeniti isti postupak razmazivanja, uspješnost detekcije je blago pala (na 90.83%), kao što se i osjetno smanjio broj lažnih detekcija. Jedini mogući uzrok ovakvom (iako blagom) padu performansi može biti nepreciznost u označavanju znakova što dovodi do toga da je ili dio znaka odrezan uporabom maske ili da je nemaskirani dio obuhvatio dio pozadine znaka, a razmazivanje je u konkretnom slučaju samo naglasilo te nepravilnosti.

3.4. Rezultati testiranja za trokutaste znakove

Učenje za trokutaste znakove provedeno je na skupu od 2100 slika (iste veličine kao i slike na kojima je naučen histogram okruglih znakova) koje sadrže oko 2300 označenih trokutastih znakova. Tako dobiveni histogram sadrži ukupno 9054 različitih vrijednosti. Za statistiku pozadina korištena je ista statistika kao i za krugove, jer tip znaka ne igra nikakvu ulogu u tome kakva se pozadina nalazi oko njega.



Slika 11: Rezultati testiranja za trokutaste znakove

Rezultati testiranja nad trokutastim znakovima pokazuju neke interesantne razlike u odnosu na rezultate dobivene nad okruglim znakovima. Prva stvar koja se zamjećuje je velik broj lažnih detekcija (12.6%), koji je kod trokuta gotovo duplo veći nego kod krugova. Takav rezultat na neki način je i očekivan, budući da dio površine pravokutnika koji okružuje znak, koji u idealnom slučaju zauzima trokutasti znak je 50%, dok je kod okruglog znaka taj postotak oko 78.5%. Zbog toga udio pozadine koja ulazi u statistiku trokutastih znakova je puno veći nego kod kružnih znakova, pa je automatski i količina lažnih detekcija daleko veća.

Slika 12: Primjer dijela okvira kojeg zauzima pozadina kod okruglog i trokutastog znaka

Slika 12, na kojoj je pozadina obojana u žuto, dobro vizualno ilustrira taj omjer znaka i pozadine unutar okvira, s time da je često okvir neprecizno postavljen oko znaka, pa je taj omjer i veći u korist pozadine. To dokazuje i činjenica da je uporabom trokutaste maske, koja obuhvaća samo dio slike koji sadrži trokut, još malo porastao broj dobro prepoznatih znakova, a kako se i očekivalo drastično se smanjio postotak lažnih detekcija, koji sada iznosi 6.4%, što je na razini rezultata dobivenih sa kružnim znakovima kad se i kod njih koristila maska.

Slika 13: Primjeri uporabe trokutaste maske kod segmentacije

Drugi važan zaključak koji proizlazi iz statistike trokuta je daleko veći broj dobro prepoznatih (true positive) znakova kod ne razmazanog histograma nego kod krugova, gotovo 28% više. Taj podatak je direktna posljedica činjenice da je baza znakova korištena za učenje trokuta daleko veća od baze znakova za krugove, što je rezultiralo histogramom koji ima više od duplo različitih vrijednosti u sebi, čime je pokriven daleko veći broj sličnih nijansi.

Dodatna potvrda tome dolazi iz rezultata razmazanog histograma za trokutaste znakove. Porast performansi je ovdje vrlo mali, gotovo neprimjetan, upravo zbog toga što je histogram trokutastih znakova bio dovoljno dobar i prije razmazivanja, pa nije posebno profitirao od tog postupka. Dokaz tome je da je histogram trokuta prije razmazivanja imao 2.2 puta više različitih vrijednosti od histograma krugova, a nakon razmazivanja taj se omjer smanjio na samo 1.38 u korist histograma trokuta.

Zaključak

Segmentacija slike je vrlo važan korak u bilo kojem sustavu koji se koristi u detekciji i raspoznavanju objekata, posebice zato jer pomaže u isticanju značajnih karakteristika traženih objekata, te smanjuje šum u podacima koje slika sadrži. Zbog toga dobra segmentacija slike može biti od iznimnog značaja za poboljšanje performansi drugih algoritama koji se primjenjuju na segmentiranoj slici. Međutim ujedno je važno da postupak segmentacije ne bude prevelik teret za performanse cijelog sustava, posebice ako postoji potreba za obradom velike količine informacija ili ako je riječ o sustavu koji radi u realnom vremenu. Zato je vrlo važno kvalitetno izraditi sustav za segmentaciju slike, i po pitanju kvalitete rezultata, i po pitanju brzine izvođenja.

Implementacija koja je izrađena u sklopu ovog završnog rada temelji se na algoritmu K srednjih vrijednosti, koji se pokazao dosta jednostavnim, što je omogućilo razvoj sustava koji provodi vrlo brzu segmentaciju, ali je unatoč svojoj jednostavnosti dovoljno robustan i daje zadovoljavajuće rezultate za naše potrebe.

Odabir boje kao jedinog kriterija za segmentaciju pokazalo se kao dosta problematično, te je bilo potrebno primjenjivati više korekcija i nadogradnji na rezultate dobivene takvom segmentacijom, da bi oni bili primjenjivi kao kriterij za detekciju prometnih znakova. Uzroci tome su detaljnije objašnjeni u radu, ali valja istaknuti daleko najutjecajniji među njima, a to je veliko odstupanje u bojama kod slika dobivenih u realnim uvjetima naspram idealnih slika. Takvo se odstupanje uspjelo barem djelomično korigirati izradom velikih histograma boja znakova, koji obuhvaćaju i boje koje su prisutne na znakovima snimljenim u lošim uvjetima, pa se i takve boje u velikom dijelu slučaja mogu uspješno prepoznati.

Ukupni rezultati dobiveni testiranjem sustava na okruglim i trokutastim znakovima, uz pomoć različito oblikovanih histograma, dalo je obećavajuće rezultate. Kod većine testiranja uspješnost prepoznavanja prometnog znaka na slici iznosi preko 90%, a kod trokutastih znakova raste i preko 95%. Jedina iznimka tome je testiranje sa ne razmazanim histogramima okruglih znakova, ali kako je utvrđeno, glavni razlog tome je relativno mali skup okruglih znakova na temelju kojih je dobiven pripadni histogram, a ne problem vezan uz realizaciju i pouzdanost samog sustava.

Primjena ovakvog sustava za samostalnu detekciju znakova nije preporučljiva jer je segmentacija kao postupak jednostavno prezahtjevna da se njome pretražuje cijela slika, a i broj lažnih detekcija je ipak prevelik da bi takav sustav bio posve pouzdan. Međutim moguća je primjena ovakvog sustava kao potpore drugom sustavu koji je specijaliziran za detekciju prometnih znakova ili drugih objekata na slici. Dobar primjer takvog specijaliziranog sustava za detekciju je sustav temeljen na algoritmu Viole i Jonesa, u čijoj sam izradi sudjelovao tijekom predmeta Projekt iz programske potpore. Takav sustav za svoje potrebe prvo pretvara sliku u crno-bijelu varijantu, čime se boja u potpunosti zanemaruje kao kriterij za detekciju. Dodatak ovakve provjere koja se temelji na bojama mogla bi biti od jako velike koristi za eliminaciju lažnih detekcija kod takvog sustava, jer često lažni pozitivni rezultati koje daje algoritam Viole i Jonesa se po bojama drastično razlikuju od prometnih znakova, a implementacija tog dodatnog kriterija samo na rezultatima Viola-Jones algoritma ne bi imala velik utjecaj na njegove performanse.

Izrada završnog rada pokazala se vrlo zanimljivom, posebice iz razloga što sam imao dovoljno slobode da samostalno donosim odluke vezane uz smjer u kojem će se rad razvijati, zbog čega sam zahvalan mentoru. Implementacija mi je također omogućila da steknem nova znanja iz područja programiranja, posebice što se tiče izrade aplikacija u skriptnim jezicima (konkretno Perlu) i integracije takvih rješenja sa programima napisanim u tipičnom sustavnom jeziku kao što je C++, u kojem je izrađen sam postupak segmentacije. Također rad na ovom zadatku, gdje ne postoji jednoznačno rješenje, te gdje postoji mnogo mogućnosti koje treba istražiti, dalo mi je uvid u jedan drugačiji pristup rješavanju takvih problema, za koji smatram da će biti vrlo koristan u budućnosti.

Literatura

[1] Shapiro and Stockman, Computer Vision*,* Prentice Hall, 2001.

[2] Tonković, Danijel, Sustav za segmentaciju slike pomakom prema srednjoj vrijednosti, diplomski rad, FER, Zagreb, 2006.

[3] Lloyd, Stuart P. , "Least squares quantization in PCM", IEEE Transactions on Information Theory **28** (2): 129–137, 1982.

[4] K-means clustering, 6. lipnja 2010. , *http://en.wikipedia.org/wiki/K-means_clustering,* 20. ožujka 2010.

[5] Lloyd's algorithm, 23. ožujka 2010. , *http://en.wikipedia.org/wiki/Lloyd's_algorithm,* 10. lipnja 2010.

[6] RGB color space, 10. lipnja 2010., *http://en.wikipedia.org/wiki/RGB_color_model,* 10.travnja 2010.

[7] Colantoni, Philippe, Color space transformations, 2004., *http://colantoni.nerim.net/download/colorspacetransform-1.0.pdf*, 15. travnja 2010.

Sažetak

Sustav za segmentaciju slika prometnih znakova na temelju boje

Algoritam koji je korišten za segmentaciju slike na temelju boje je algoritam K srednjih vrijednosti, koji se temelji na grupiranju elemenata slike oko *k* srednjih vrijednosti. Radi poboljšanja efikasnosti takvog algoritma koristi se heuristika za izbor srednjih vrijednosti koja se temelji na pronalaženju točaka s najvećom minimalnom udaljenosti od postojećih srednjih vrijednosti u RGB prostoru boje, koji se općenito koristi tijekom cijelog postupka segmentacije. Izrađeni su histogrami raspodjele boja na skupovima slika sa prometnim znakovima, te su takvi histogrami potom korišteni u svrhu detekcije znakova na temelju sličnosti boja na slici i boja u histogramu. Radi poboljšanja vremenskih i kvalitativnih performansi sustava implementirane su i maske koje otklanjaju dio okvira sa znakom koji sadrži pozadinu znaka. Takav sustav je testiran, te su rezultati testiranja obrađeni i prikazani, te je na temelju njih donesen zaključak o primjenjivosti takvog sustava.

Ključne riječi

Segmentacija slike, računalni vid, algoritam K srednjih vrijednosti, segment, RGB prostor boja, prostor slike, detekcija, prometni znakovi, srednja vrijednost, početni uvjeti, histogram, apsolutni udio, relatvni udio, skup za učenje, lažna detekcija, okrugla i trokutasta maska, razmazivanje histograma

Summary

System for traffic sign image segmentation based on color

The algorithm used for image segmentation based on color is the K-means algorithm, which is based on grouping of image elements around *k* different means. In order to improve the performances of this algorithm a heuristic is used to chose the initial means and it is based on finding the points that have the maximum minimal distance from the existing means in the RGB color space, which is used during the whole segmentation process. Color distribution histograms have been created from sets of images of traffic signs and those histograms where used in the detection of traffic signs based on the similarity of the image colors with the colors in the histograms. To further improve the performance of the system, both in terms of time consumption and quality of detection, masks have been introduced which remove the background of traffic signs. The system has been tested and the results have been processed and displayed, and based on these a conclusion have been given about the applicability of such a system.

Keywords

Image segmentation, computer vision, K-means algorithm, segment, RGB color space, image space, detection, traffic signs, mean, average value, initial conditions, histogram, absolute share, relative share, training set, false positive detection, circular and triangular mask, histogram smearing