

Sveučilište u Zagrebu

FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

Projekt iz predmeta Raspoznavanje uzorka

Ak. god. 2016/17

Detekcija lica u videu

Autori:

Branimir Akmadža

Jakov Grković

Domagoj Krivošić

Luka Novak

Vilim Šoštarić

Siječanj, 2017.

Uvod

Računalni vid je područje računarstva koje se bavi ekstrakcijom korisnih informacija iz slika ili video sekvenci, te sustavima koji takvu ekstrakciju omogućavaju. Računalni vid je usko povezan s drugim područjima u računarstvu, kao npr. umjetnom inteligencijom, raspoznavanjem uzorka, strojnim učenjem, robotikom, obradom signala, obradom slika itd. Važnije teme koje istražuje računalni vid su pronalaženje dvodimenzionalnih ili trodimenzionalnih objekata, raspoznavanje objekata, detekcija događaja, rekonstrukcija scene, restauracija slika, praćenje pokreta itd. Pronalaženje (engl. detection) objekata u računalnom vidu se bavi problemom nalaženja objekata koji pripadaju nekoj zadanoj klasi u slici ili video sekvenci. Dok je ljudima vrlo lako "intuitivno" detektirati objekt neovisno o njegovoj veličini, kutu gledanja, mogućoj rotaciji ili djelomičnoj zaklonjenosti, računalima to predstavlja problem, te taj problem treba predstaviti nekim algoritmom da bi on računalu bio shvatljiv. U tom smislu se koristi raspoznavanje uzorka. To je grana strojnog učenja koja je primjenjiva na cijeli niz problema od kojih je računalni vid pri vrhu. Najčešće se pronalaženje objekata primjenjuje za pronalaženje ljudskih lica, pješaka, nekih zadanih znakova itd. Postoji razlika između detekcije objekata i raspoznavanja objekata. Raspoznavanje objekata (engl. recognition) bavi se određivanjem konkretne klase na uzorku na kojem se nalazi više sličnih klasa. Za razliku od toga, detekcija objekata se bavi određivanjem nalazi li se uopće na slici objekt neke klase koju tražimo. Postoji više metoda kojima je moguće ostvariti učinkovite sustave za detekciju objekata. Zbog razvijenih digitalnih kamera visoke rezolucije nastala je potreba za algoritmima za što bržu obradu sve masivnijih slika. Paul Viola i Michael J. Jones su predložili izuzetno brzu metodu detekcije objekata koja se temelji na kaskadi boostanih Haarovih klasifikatora. Ta metoda se zbog svoje brzine koristi za detekciju objekata u realnom vremenu. Izgradnja kaskade je određena većim brojem parametara (primjerice, tip algoritma boostanja, količina pozitivnih i negativnih primjera, korišteni tipovi Haarovih klasifikatora i sl.).

Projektni zadatak

Opis projektnog zadatka

Oblikovati sustav za automatsku detekciju lica temeljenu na fuziji značajki u video scenama. Sustav treba omogućiti robusnu detekciju lica na slikama dobivenim pri velikoj varijaciji osvjetljenja i poze lica. Usporediti dobivene rezultate s rezultatima metode Viola - Jones.

Pregled i opis srodnih rješenja

Najčešće korištena implementacija učenja kaskade boostanih Haarovih klasifikatora se nalazi u OpenCV programskoj biblioteci koja je opisana u 4. odjeljku. Većina drugih rješenja bazira se na izmjenama implementacije iz OpenCV biblioteke. Primjerice, Allusse et al. (2008) su razvili "GpuCV: GPU-accelerated Computer Vision." Radi se o implementaciji koja je napravljena po uzoru na OpenCV biblioteku, ali je prilagođena izvršavanju na grafičkim procesorima.

Slična prilagodba je opisana u (Mun, 2009a). Navedeni su rezultati prilagodbe OpenCV algoritama za izvođenje na procesorima bez jedinice za računanje s pomičnim zarezom (engl. Floating Point Unit, FPU). Primjer su ARM procesori koji su izuzetno često korišteni u ugradbenim računalima.

U (Mun, 2009b) opisan je postupak iskorištavanja MPI (Message Passing Interface) tehnologije za izradu raspodijeljenog postupka učenja kaskade. Navodi se konfiguracija učenja za koju je na jednom osobnom računalu bilo potrebno 6 dana (te 4 dana korištenjem OpenMP implementacije OpenCV-a), a raspodjeljom računanja na grozd (engl. cluster) od 11 povezanih osobnih računala, učenje kaskade je trajalo samo 21 sat.

Postupak rješavanja zadatka

Viola - Jones metoda

Opis metode Viola – Jones

Metoda koju su predložili Paul Viola i Michael J. Jones je jedna od metoda pronalaženja objekata koja koristi strojno učenje, te ujedno i prva koja postiže dobre rezultate u realnom vremenu (Viola i Jones, 2002). Primarna svrha joj je detekcija lica [2]. Algoritmi strojnog učenja korišteni u detekciji objekata uzimaju podatke o slikama koje su već prethodno obrađene i sasvim sigurno znamo sadrže li ili ne taj objekt, te na temelju naučenog znanja uspješno detektiraju objekte na novom skupu slika. Važna prednost metoda temeljenih na strojnom učenju je mogućnost primjene na razne klase objekata, samo je potrebno prikupiti podatke za učenje. Tako isti sustav možemo primijeniti za pronalaženje lica, pronalaženje prometnih znakova ili bilo koju drugu klasu objekata. Postoji više algoritama za gradnju klasifikatora kojima klasificiramo podatke. Klasifikatorski sustavi su sustavi temeljeni na skupu pravila (klasifikatora) koja određuju reakciju sustava na uvjete dane u njegovoj okolini. Ta pravila se uče na temelju primjera koji se nalaze u skupu za učenje. Detekciju objekata tako možemo gledati kao zadaću gradnje klasifikatora, pri čemu se slika ili video sekvenca na kojoj tražimo objekte, razdvaja u više manjih potprozora, i odluka se donosi za svaki potprozor na temelju toga sadrži li on traženi objekt ili ne. Budući da se to pokazalo kao proces s velikom složenošću, umjesto direktnog rada s pikselima ulazne slike, ove metode rade s određenim skupom drukčijih značajki te slike. Sustavi bazirani na tim značajkama će raditi znatno brže nego sustavi bazirani na pikselima. Vrijednosti dobivene takvim računom predstavljaju prisustvo ili odsustvo nekih karakteristika slike (kao što su rubovi, kutevi i sl.). Metoda Viole i Jonesa vrlo brzo procesira slike i postiže visoki broj detekcija, te je brzo nakon objavlјivanja postavila standarde u području detekcije lica. Istraživanjem iz 2002. godine ustanovljeno je da je metoda Viola - Jones, u usporedbi s ranijim metodama, učinkovitija najviše zbog 4 brzine pronalaženja (na računalu 700MHz Intel Pentium 3 postignuta je detekcija od 15 slika u sekundi, na slikama veličine 384×288 piksela

(Viola i Jones, 2002)). Tako veliku brzinu ova metoda postiže među ostalim zbog rada s informacijama koje se nalaze u samom crno–bijelom području slike. Moguće je kombinirati nekoliko različitih metoda, te postići još veću učinkovitost u pronalaženju objekata. Ulazni parametar za ovaj algoritam je slika bilo koje vrste, a izlazni parametar je lista pozicija lica na slici. U nastavku su objašnjena tri glavna dijela od kojih se metoda sastoji.

Možemo izdvojiti 3 glavna doprinosa metode Viole i Jonesa:

Integralna slika - Novi prikaz slike, koji omogućava detektoru koji koristimo da brzo izračuna značajke slike.

Gradnja klasifikatora - Klasifikator je izgrađen korištenjem AdaBoost algoritma za strojno učenje koji izdvaja mali broj kritičnih značajki slike od vrlo velikog broja potencijalnih značajki slike.

Metoda kombinacije klasifikatora u kaskadu - Omogućava da se brzo odbace suvišni predjeli slike, i detaljnije provede računanje nad predjelima slike koji sadrže mogući objekt koji tražimo, te time poboljšava performanse postupka.

Haarove značajke

Budući da se to pokazalo kao proces s velikom složenošću, metoda Viole i Jonesa, umjesto direktnog rada s pikselima ulazne slike, radi s određenim skupom drukčijih značajki te slike. Sustavi bazirani na tim značajkama će raditi znatno brže nego sustavi bazirani na pikselima. Haarove značajke su pravokutne značajke, i računaju se kao razlika nekoliko suma piksela unutar različitih pravokutnih područja slike. Vrijednosti dobivene takvim računom predstavljaju prisustvo ili odsustvo nekih karakteristika slike (kao što su rubovi, kutevi i sl.).

Tri su tipa značajki definiranih u metodi Viola i Jonesa:

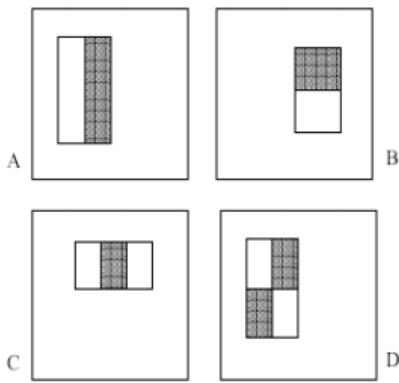
Two-rectangle: Računa razliku suma piksela u 2 pravokutna predjela istog oblika i veličine koji su spojeni jednim vertikalnim ili horizontalnim bridom

Three-rectangle: Uz 3 spojena pravokutnika istog oblika i veličine, računa razliku sume piksela 2 vanjska pravokutnika i sume piksela centralnog pravokutnika

Four-rectangle: Uz 4 spojena pravokutnika istog oblika i veličine, računa razliku suma piksela dijagonalnih parova pravokutnika

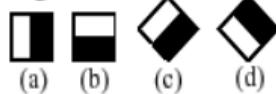
Osnovne Haarove značajke su samo two-rectangle značajke, a ostale dvije vrste su u svojoj metodi dodali Viola i Jones. Na detektoru razlučivosti 24×24 piksela, skup Haarovih značajki sadrži oko 160000 značajki, što je očigledno puno više od broja piksela slike. Tako nam Haarove značajke daju brojne informacije o slici, a zbog jednostavnog i učinkovitog načina kojim se one izračunavaju, vrlo su pogodan alat pri pronalaženju objekata.

- Original set of features:

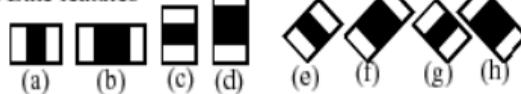


- Extended set of features:

1. Edge features



2. Line features



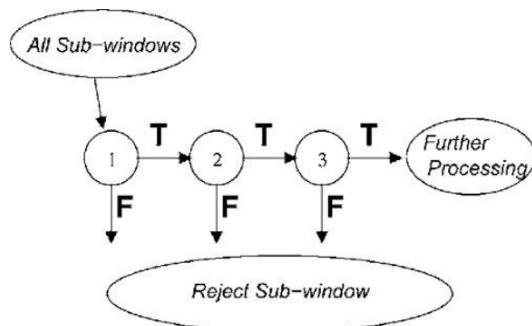
3. Center-surround features



Slika 1. Prikaz Haarovih značajki. Vrijednost značajke je razlika zbrojeva slikovnih elemenata ispod bijelih i crnih pravokutnika.

Kaskada klasifikatora

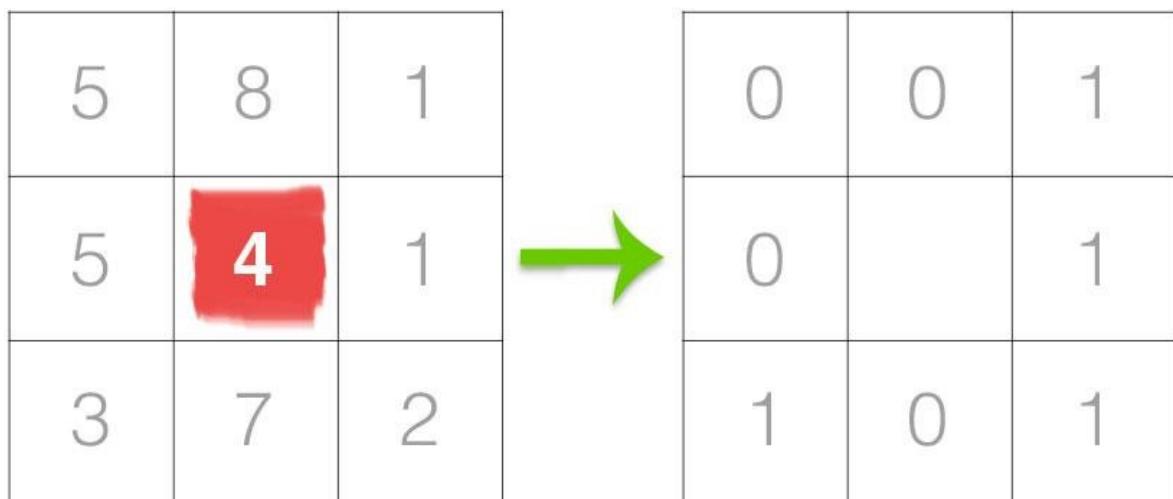
Nakon što su klasifikatori trenirani, performanse se mogu poboljšati kombiniranjem više klasifikatora. Jedan klasifikator dobiven prethodnim algoritmom učenja teško može davati rezultate u realnom vremenu pa se klasifikatori kombiniraju u kaskadu poredani po kompleksnosti, od najjednostavnijeg do kompleksnijeg. Prvi klasifikator u kaskadi dobiva na ulaz sve prozore slike, kako je prikazano na Slici 2., a svaki sljedeći klasifikator evaluira samo one primjere koje prethodni nije odbacio kao negativne. Ako je neki prozor u tom procesu odbačen od strane nekog klasifikatora, on se potpuno odbacuje i na njega se više ne primjenjuju nikakve operacije, te se pretraživanje nastavlja od idućeg prozora. Takvim postupkom kaskada povećava brzinu cijelog detektora, jer fokusira pažnju na predjele slika za koje je veća vjerojatnost da sadrže traženi objekt (za promatrani problem, lice).



Slika 2. Model kaskadnog klasifikatora

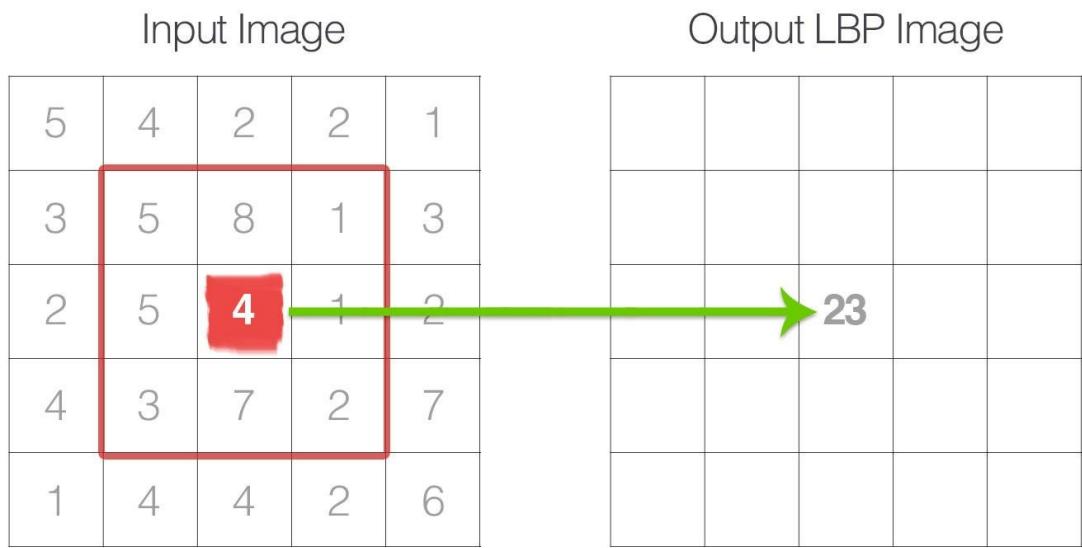
Local binary patterns

Local binary patterns (LBP) su deskriptor teksture koji je postao popularan 2002. godine znanstvenim radom “*Multiresolution Grayscale and Rotation Invariant Texture Classification with Local Binary Patterns*” (Ojala et al.), iako je ovaj koncept poznat još iz 1993. Godine [1]. LBF se temelji na lokalnoj reprezentaciji teksture usporedbom piksela s njihovim susjedima. Postupak se provodi na sivoj (greyscale) slici. Za svaki piksel u slici izaberemo usporedimo njegovu vrijednost s vrijednostima njegovih 8-susjednih piksela određenim redom (npr. u smjeru kazaljke na sat). Smjer i pozicija početnog susjeda moraju biti konzistentni za svaki piksel u slici. Ako je vrijednost susjeda veća od promatranog centralnog piksela zapisujemo 1, ako je jednaka ili manja zapisujemo 0.



Slika 3. Način rada LBP klasifikatora

Time za svaki piksel u slici dobivamo njegovu LBP vrijednost, što je niz od 8 bita koji daje 256 (2^8) mogućih vrijednosti. LBP vrijednost piksela prebacuje se u dekadsku vrijednost (0-255) i sprema se u 2D matricu tako da je pozicija LBP vrijednosti piksela jednaka poziciji piksela u originalnoj slici (Slika 4.).

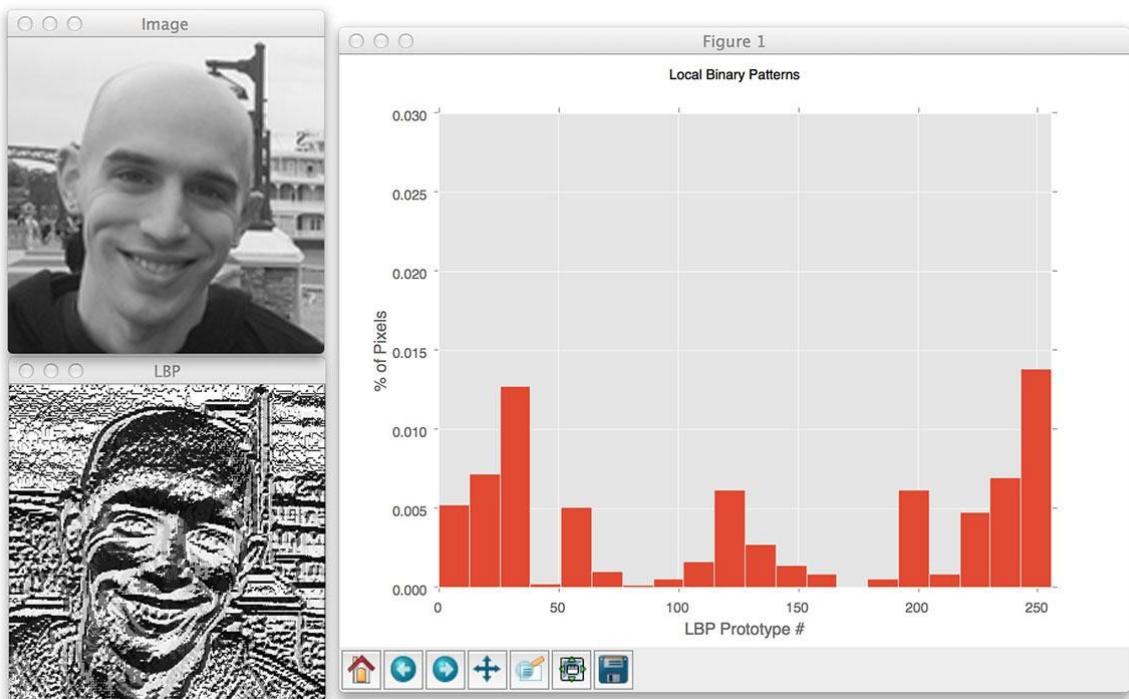


Slika 4. Način zapisa vrijednosti piksela u 2D matrici

Ovaj se postupak dobivanja LBP vrijednosti provodi za svaki piksel u ulazne slike. Tako se dobiva LBP reprezentacija ulazne slike. Zadnji korak je izračunavanje i stvaranje histograma nad izlaznom LBP matricom (Slika 5.). Pošto vrijednosti LBP kodova piksela imaju 8 bitova, dakle 256 mogućih vrijednosti, dobiva se 256-bin histogram koji služi kao završni vektor značajki (Slika 6.).



Slika 5. Usporedba originale slike i slike obrađene LBP klasifikatorom

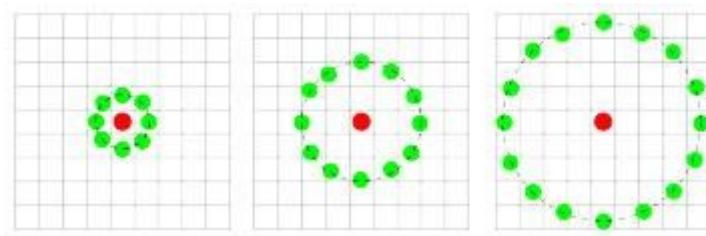


Slika 6. Vektor značajki LBP klasifikatora (256-pinski histogram)

Glavna prednost originalne LBP implementacije je to što se njom mogu uočiti najmanje detalje u slikama, no to joj je i glavni nedostatak. Koristeći isključivo 8-susjedne piksele za LBP reprezentaciju piksela nas ograničava na detalje, dok se veći uzorci ne mogu tako očitati.

Kako bi se ovaj problem riješio, Ojala et al. su predložili poboljšanu implementaciju algoritma tako da veličina razmatranog susjedstva piksela bude varijabilna. U tu svrhu, u implementaciju su dodana dva parametra:

- broj piksela p u simetričnoj kružnici oko razmatranog piksela
- radijus r kružnice po kojoj se izračunavaju LBP vrijednosti



Slika 7. Prikaz implementacije s varijabilnom analizom susjednih piksela

Potrebno je naglasiti da povećanjem ovih parametara povećava se broj razmatranih susjeda, a time i veličina LBP koda jednog piksela, što uzrokuje većom dimenzionalnošću krajnjeg histograma i vektora značajki (Slika 7.).

Ispitivanje rješenja

Ispitna baza

Za evaluaciju modela koristimo bazu AFW (annotated faces in the wild) slika s Flickr-a koju smo preuzeeli sa stranica University of California, Irvine. Slike su nasumično izabrane od slika koje sadrže barem jedno veliko lice. Konačna baza koju smo koristili za ispitivanje sadrži 205 slika s 468 lica. Slike imaju natrpane pozadine i lica koja se razlikuju u starosti, boji kože, izrazu, rotaciji, šminki te po tome nose li ljudi naočale ili ne.

Analiza rezultata

Od ukupnog broja od 468 lica na 205 slika LBP klasifikator detektirao je 224 lica s 22 lažnih detekcija (false positive), dok je Viola-Jones klasifikator s Haarovim značajkama detektirao samo 116 lica s 30 lažnih detekcija (false positive). Slabi rezultati oba klasifikatora proizlaze iz činjenice da oba jako ovise o osvjetljenju lica (detekcije su neuspjele kako na jako tamnim tako i na iznimno osvijetljenim licima) te rotacijama glave (detekcije su neuspjele za profilna lica). Značajno je izražena razlika u točnosti između dva klasifikatora s točnošću LBP klasifikatora od 47.86% i točnošću Haarovog klasifikatora sa 24.79%.

Opis programske implementacije rješenja

Za pisanje računalnog programa korišteno je programsko okruženje Eclipse i programski jezik Java. Grafički prikaz prepoznavanja lica u stvarnom vremenu rađen je u JavaFx frameworku. U projektu postoje dvije različite implementacije. Jedna se bavi prepoznavanjem lica u stvarnom vremenu (eng. Real-time recognition), a druga uzima i obrađuje skup slika.

Prepoznavanje lica u stvarnom vremenu

Program radi tako da koristi prednju kameru računala tako da u malim jedinicama vremena izdvaja dijelove videa i na njima radi detekciju lica kao da je riječ o detekciji slike. Prije same detekcije potrebno je provesti nekoliko koraka obrade slike da bi detekcija bila učinkovitija. Prvi korak je pretvorba slike u matricu. Matrica je najbolji zapis slike jer je pogodna za matematičke izračune u računalu. Idući korak je pretvorba matrice u boji u sivu matricu.

- 1.) Pretvorba rgb matrice u sivu matricu
- 2.) Izjednačavanje histograma

Koraci 1.) i 2.) prikazani su u sljedećem odsječku koda:

```
// convert the frame in gray scale
Imgproc.cvtColor(rgb, grayFrame, Imgproc.COLOR_BGR2GRAY);
// equalize the frame histogram to improve the result
Imgproc.equalizeHist(grayFrame, grayFrame);
```

Prva dva koraka provode se u svrhu poboljšanja rezultata. „rgb“ je prvočitna matrica u boji, a „grayFrame“ je siva matrica (crno-bijela). Histogram je reprezentacija raspodjele intenziteta (u ovom slučaju) sive boje za svaki piksel na matrici. Izjednačavanje histograma je metoda kojom povećava kontrast slike što je posebno važno u području slike gdje kontrast nije lokalno izražen. Efekt takve metode je da se dobije kvalitetnija raspodjela intenziteta piksela, a time i omogući kvalitetniji daljnji rad.

3.) Identifikacija lica s karakterističnim parametrima

U sljedećem odsječku koda prikazana je metoda kojom se detektiraju moguća lica na slici:

```
// detect faces
faceCascade.detectMultiScale(grayFrame, faces, 1.1, 2, 0 | Objdetect.CASCADE_SCALE_IMAGE,
    new Size(0.2 * mjestoZaSliku.getFitWidth(), 0.2 * mjestoZaSliku.getFitHeight()), new Size());
```

Metoda prima sljedeće parametre:

„grayFrame“ –siva matrica

„faces“ – varijabla u koju spremam sva pronađena lica, odnosno sve što algoritam smatra licem

vrijednost 1.1 – označava da reprezentativni okvir kojim se traži lice u algoritmu može u dimenziji odstupati 10% u veličini (što je dopušteno veće odstupanje, to je algoritam brži, ali i manje precizan. Definirana vrijednost je dobivena eksperimentalno)

vrijednost 2 – označava koliko je minimalno susjednih pravokutnika koji su kandidati za lice potrebno zapamtiti. Što je ta vrijednost veća, bolja je kvaliteta pronađenih lica, ali je i veća vjerojatnost da će se lica sa slabijim obilježjima zanemariti

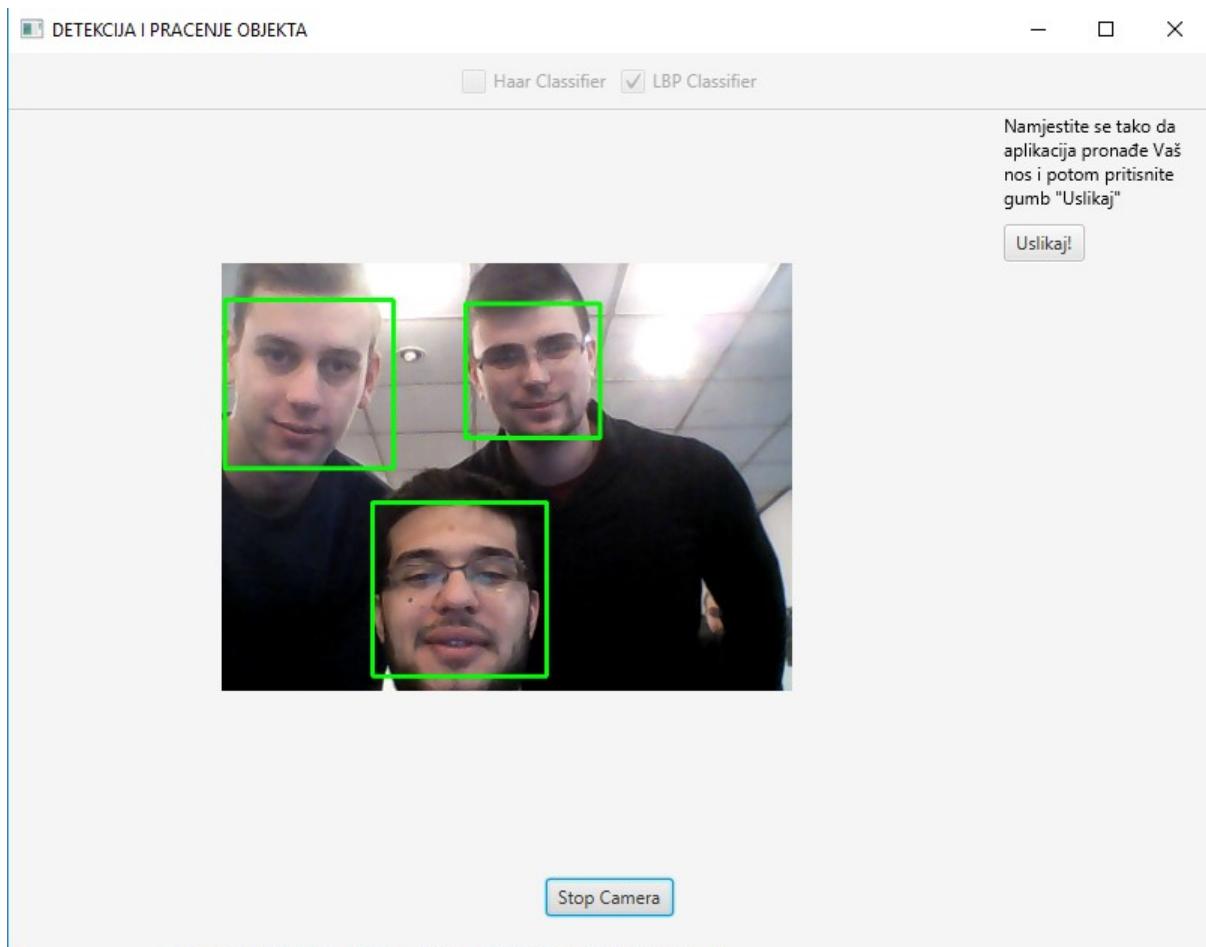
CASCADE_SCALE_IMAGE – zastavica korištena za sinkronizaciju metode s bibliotekom (nije od velike važnosti za rad)

Size (prvi) – označava minimalnu veličinu lica koja se očekuje. U primjeru je uzeto 20% širine i visine učitane slike je dobiveno kao optimalna dimenzija. Svaki objekt manji od tih dimenzija bit će ignoriran

Size (drugi) – maksimalna veličina slike koja se očekuje. Svaki objekt veći od te dimenzije bit će ignoriran. U primjeru nije ograničena maksimalna dimenzija zbog mogućnosti da lice prekriva gotovo cijelu sliku

Prepoznavanje lica u bazi testnih slika

Rad proučavanog LBP klasifikatora je proučavan na i na temelju rezultata za zadalu bazu koja se sastoji od slika na kojima su lica osoba. S obzirom na to da je naučeni klasifikator, klasifikator prednje strane lica, za bolju kvalitetu detekcije, dodan je i klasifikator lica iz profila, što je povećalo učinkovitost detekcije. Ovaj dio koda radi tako da uzima sliku po sliku iz baze za učenje, svaku posebno obradi te označi regiju koju smatra licem i pohrani novu označenu sliku u mapu gdje se nalaze rezultati. Sama obrada slike i detekcija radi jednako kao što je objašnjeno u prethodnom poglavlju, što je i vidljivo sa Slika 8. I 9..



Slika 8. Primjer detekcije u stvarnom vremenu



Slika 9. Usporedba Haarovog i LBP klasifikatora na primjeru slike iz baze

Zaključak

Na temelju rezultata u detekciji lica u stvarnom vremenu (eng. Real time detection), može se reći kako oba klasifikatora daju zadovoljavajuće rezultate zahvaljujući brzini obrade svakog elementa videa (eng. Frame). Osim brzine, i preciznost detekcije je također bila na visokoj razini. Dodatno, za dobru detekciju nisu smetali ni veličina lica (udaljenost od kamere), ni broj lica u okviru slike. Eksperimentalnim putem došli smo do zaključka da je Haarov klasifikator ipak davao više false positive detekcija lica od LBP klasifikatora. Međutim, rezultati detekcije lica iz zadane baze podataka su takvi da zahtijevaju poboljšanje klasifikatora. Taj zaključak dolazi iz činjenice da su oba klasifikatora teško pronalazila lica pri intenzivnim osvjetljenjima ili većim rotacijama glave. Ispravljanje takvih nepreciznosti bio bi temelj za daljnji razvoj i rad.

Literatura

- [1] Adrian Rosebrock, *Local Binary Patterns with Python & OpenCV*, izvor : <http://www.pyimagesearch.com/2015/12/07/local-binary-patterns-with-python-opencv/>, 2015.
- [2] YI-QING WANG, *An Analysis of the Viola-Jones Face Detection Algorithm*, izvor: <https://doi.org/10.5201/1pol.2014.104>, 2014.