FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

**Projekt iz predmeta Raspoznavanje uzoraka**

**Ak. god. 2012/13**

**Detekcija i „meko“ raspoznavanje na temelju siluete osoba**

Autori:  
Igor Jerković  
Petar Alilović

Matija Bagić

Tena Dominko

Daniel Ćuk  
Viktor Skolan

Martin Šošić

Matija Šošić

**siječanj, 2013.**

# Sadržaj:

[Sadržaj: I](#_Toc346532239)

[1. Projektni zadatak 1](#_Toc346532240)

[1.1 Opis projektnog zadatka 1](#_Toc346532241)

[1.2 Pregled i opis srodnih rješenja 1](#_Toc346532242)

[1.3 Konceptualno rješenje zadatka 2](#_Toc346532243)

[2. Postupak rješavanja zadatka 3](#_Toc346532244)

[2.1 Dobivanje crno-bijele siluete 3](#_Toc346532245)

[2.1.1 Oduzimanje pozadine 4](#_Toc346532246)

[2.2 Računanje Granlundovih koeficijenata 7](#_Toc346532247)

[2.3 Klasificiranje na osnovu proporcija tijela 8](#_Toc346532248)

[2.4 PCA 12](#_Toc346532249)

[2.5 Template matching 15](#_Toc346532250)

[3. Ispitivanje rješenja 16](#_Toc346532251)

[3.1 Ispitna baza 16](#_Toc346532252)

[3.2 Rezultati učenja i ispitivanja 16](#_Toc346532253)

[3.2.1 Rezultati - Body proportions matching 16](#_Toc346532254)

[3.2.2 Rezultati - Template matching 19](#_Toc346532255)

[3.2.3 Rezultati - PCA 19](#_Toc346532256)

[3.3 Analiza rezultata 20](#_Toc346532257)

[3.3.1 Body proportions matching 20](#_Toc346532258)

[3.3.2 PCA 21](#_Toc346532259)

[3.3.3 Template matching 21](#_Toc346532260)

[4. Opis programske implementacije rješenja 22](#_Toc346532261)

[5. Zaključak 24](#_Toc346532262)

[6. Literatura 26](#_Toc346532263)

# Projektni zadatak

## Opis projektnog zadatka

Prvi dio projektnog zadatka je detekcija osobe i izlučivanje njezine siluete iz slike. Detekcija osobe je dosta zahtjevan zadatak s obzirom da je ulazni podatak statična slika osobe, a ne video sekvenca kretanja osobe. Dodatno, osoba može biti tako odjevena da bude više ili manje uklopljena u pozadinu, što naravno otežava proces same detekcije. Različite mogućnosti izgleda pozadine zahtijevaju dovoljno dobre načine obradbe ulazne slike radi što boljeg dohvaćanja siluete osobe. Silueta osobe je na slici prezentirana skupom bijelih piksela, dok je pozadina crne boje. Silueta odražava figuru osobe i njezine posebnosti vezane uz građu tijela i frizure (posebno se očituje ako osoba ima dugu raspuštenu kosu). Sva ostala svojstva su izgubljena. Upravo zbog te činjenice da se barata sa siluetama koje su binarne (crno - bijele) slike potrebno je dosta eksperimentiranja oko pronalaska algoritma koji će uspjeti obuhvatiti najbolje značajke iz silueta.

Nakon što je dohvaćena silueta osobe prelazi se na fazu raspoznavanja. „Meko“ raspoznavanje odnosi se na pojam nepotpunog, odnosno nepreciznog raspoznavanja. Nepreciznost raspoznavanja nastaje zbog vrlo malo značajki koje su posebne samo za tu osobu, kao što ih ima na primjer kod otiska dlana, slike lica, otiska prstiju i ostalih biometrijskih značajki, koje jedinstveno određuju tu osobu. Kao što je i ljudskom oku vrlo teško raspoznati osobu na ulici kad vidi samo njenu sjenu, tako je i računalu vrlo teško odrediti kojoj osobi pripada ta silueta. Iz navedenih razloga treba problemu pristupiti sa stajališta da je rezultat identifikacije osobe s obzirom na njenu siluetu vrlo zahtjevan problem kojem treba pristupiti iz više aspekata i na više načina.

Što se samog izbora algoritma tiče tu nema nikakvog ograničenja što je u slučaju ovog projekta povoljna situacija jer omogućava isprobavanje različitih metoda i razvoj drugačijih algoritama koji će možda unaprijediti neke postojeće algoritme ili, još bolje, implementirati će se novi algoritmi koji s obzirom na ulazne slike silueta osoba izluče najvažnije značajke koje će omogućiti što bolje raspoznavanje i identifikaciju osobe.

## Pregled i opis srodnih rješenja

Iscrpan pregled srodne literature s predloženim rješenjima. Opis postojećih ispitnih baza (linkovi na javno dostupne baze).

## Konceptualno rješenje zadatka

Opisati niz algoritama i koncepata koji će se koristiti u rješavanju i to redom kojim će se koristiti. Nabrojiti ulazne podatke u niz i sve međurezultate do rješenja.

Za rješavanje zadatka koristi se baza slika koja obuhvaća barem 5 slika svake od 24 osoba u bazi. Prvi korak je detekcija osobe na slici, to jest odvajanje pozadine. Algoritam prima dvije slike (sliku pozadine i sliku iste te pozadine sa osobom na slici) te stvara novu binarnu sliku na kojoj pikseli s vrijednošću 0 predstavljaju pozadinu (crno područje), a pikseli s vrijednošću 1 osobu (bijelo područje). Dobivene slike spremaju se u bazu i koriste u daljnjim koracima.

Sljedeći koraci su izlučivanje i odabir značajki za što se koristi više različitih tehnika. Jedna od njih je računanje Granlundovih koeficijenata. Iz binarne slike dobivene u prethodnom koraku izdvoji se kontura koja predstavlja osobu na slici. Za dobivenu konturu računaju se Granlundovi koeficijenti koji predstavljaju vektor značajki. Značajke se dalje prosljeđuju na dva načina: kao mapa svih vektora značajki koji se sastoje od izračunatih koeficijenata i kao matrica nastala korištenjem algoritma PCA na mapu vektora značajki.

Igorov HW matching

Viktorov template matching

Nakon izlučivanja značajki slijedi učenje klasifikatora. Na metodama za koje je to moguće smo isprobali razne klasifikatore (kNN, Random Forest, SVM, Bayes).

Dobiveni klasifikator zatim testiramo na skupu za testiranje kako bismo pronašli najbolji klasifikator.

Dobivanje silueta

Izlučivanje

značajki

Učenje klasifikatora

Slika : Učenje klasifikatora

Dobivanje siluete

Izlučivanje

značajki

Klasifikacija

# Postupak rješavanja zadatka

Slika : Klasifikacija nove slike

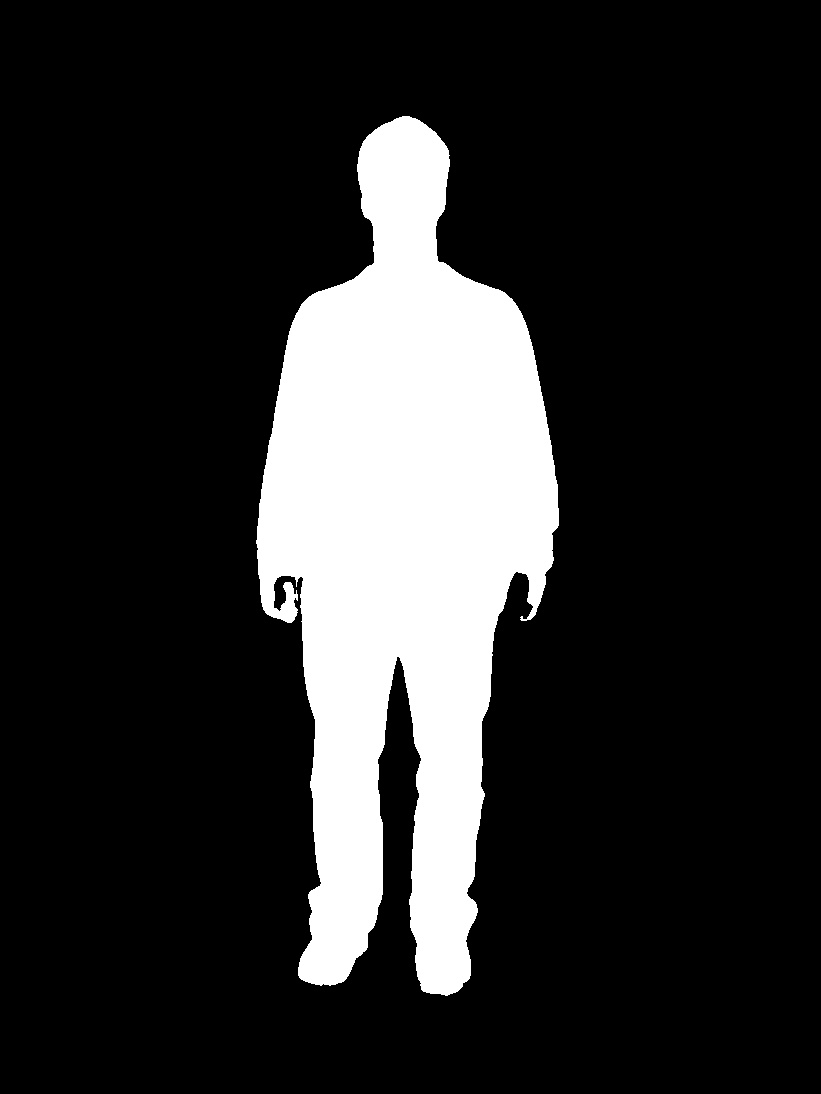
Navesti numerirani slijed koraka rješavanja. Npr.: 1. Dobivanje binarne slike iz slike u boji, 2. Segmentacija objekata na slici, 3. Nalaženje rubova u slici ...

## Dobivanje crno-bijele siluete

* **Ulazi:**
  + Slika pozadine (dalje u tekstu „slika B“)
  + Slika iste te pozadine sa osobom na slici (dalje u tekstu slika „F“)

|  |  |
| --- | --- |
| **PC200025.jpg**  Slika : Slika B | **PC200026.jpg**  Slika : Slika F |

* **Izlaz:**
  + Crno-bijela binarna slika na kojoj je silueta osobe bijele boje a pozadina je crne boje



Slika : Dobivena silueta

U ovom koraku iz slike na kojoj se nalazi osoba dobivamo njenu siluetu.   
Pri tome radi jednostavnosti pretpostavljamo da je sustav statičan tj. da nam je poznata pozadina. Kada tako pojednostavimo sustav onda siluetu osobe možemo dobiti oduzimanjem pozadine (engl. background subtraction).

### Oduzimanje pozadine

Osnovna ideja je vrlo jednostavna: ako imamo sliku pozadine te imamo sliku nekog objekta ispred te pozadine, onda taj objekt možemo izdvojiti sa slike tako da izdvojimo sve piksele u kojima se slike razlikuju. To se uobičajeno radi oduzimanjem slike pozadine od slike s objektom od kuda je i došao naziv postupka.

Nažalost ideja je jednostavna samo dok se radi o idealnim uvjetima. U realnim uvjetima pikseli pozadine na dvije slike neće nikada biti sasvim identični zbog više razloga kao što su sjena koju baca objekt, različito osvjetljenje, sičušni pomak kamere/fotoaparata.

Kako bismo riješili taj problem modificirati ćemo postupak tako da postane neosjetljiv na šum, manje promjene u osvjetljenju te mali pomak kamere/fotoaparata.

#### Kalibriranje slike

Kako bismo riješili problem pomaka kamere/fotoaparata uvesti ćemo dodatni korak koji zovemo kalibriranje slike.

Pretpostavimo da je slika F naspram slike B pomaknuta za određeni pomak (dx, dy) gdje su dx i dy cijeli brojevi. Ako nam je taj pomak poznat onda možemo sliku F translatirati za (-dx, -dy) čime dobivamo kalibriranu sliku F s obzirom na sliku B. Naš postupak tako postaje neosjetljiv na pomake kamere/fotoaparata.

Problem koji preostaje riješiti je pronalazak iznosa pomaka (dx, dy). Unatoč mogućim razlikama u osvjetljenju među slikama i šumu očekujemo da će razlika između dvije slike biti minimalna kada su slike točno kalibrirane tj. kada sliku F translatiramo upravo za (-dx, -dy). Sada smo pronalaženje pomaka (dx, dy) sveli na pronalazak argumenata (dx, dy) koji minimiziraju razliku između translatirane slike F i slike B. Pronalazak najboljih (dx, dy) smo realizirali tako da računamo razliku između translatirane slike F i slike B za svaki par (dx, dy) gdje dx i dy iteriramo od -10 do 10 uz korak od 2 (ukupno 100 kombinacija) te biramo onaj par (dx, dy) koji daje najmanju razliku. Zatim na isti način još radimo finu kalibraciju u krugu od jednog piksela od pronađenog najboljeg para (dx, dy) te je to konačni pomak.

Treba napomenuti da razliku između translatirane slike F i slike B računamo tako da zbrojimo razlike za svaki par odgovarajućih piksela. Važno je primjetiti da se translacijom slike F pojavljuju nedefinirani pikseli te moramo odlučiti kako ćemo ih uspoređivati sa pikselima sa slike B. Bilo bi loše kada bismo pretpostavili da su jednaki pikselima iz slike B jer bi onda optimalan pomak bio upravo jednak (sirina\_slike, visina\_slike). Pokazalo se dobrim tu razliku definirati na neku veliku vrijednost jer se tako kažnjavaju velike translacije te izbjegavaju one koje ne doprinose bitno smanjenju razlike.

Nakon što smo pronašli pomak (dx, dy) radimo translaciju slike F za (-dx, -dy). Postavlja se pitanje na koju vrijednost postaviti nedefinirane piksele nastale translacijom slike F. Pošto ionako pretpostavljamo da smo sliku F uspješno kalibrirali sa slikom B onda te nedefinirane piksele postavljamo na iste vrijednosti kao što su odgovarajući pikseli na slici B.

Opisani postupak se pokazao uspješnim no zbog računske složenosti smo ga ograničili da radi samo za pomake do 10 piksela (što se pokazalo dovoljnim).

#### Zaglađivanje slika

Kako bismo donekle usrednjili vrijednosti piksela i ublažili šumove radimo zaglađivanje (engl. „smoothing“) slike po blokovima. Za to koristimo Gaussovu funkciju . Prvo zagladimo posebno sliku F i sliku B gdje koristimo blokove veličine k1, zatim radimo razliku slika te potom ponovno zagladimo tu razliku gdje koristimo blokove veličine k2. Kao najbolje vrijednosti pokazale su se k1 = 3 i k2 = 7.

#### Računanje razlike slika

Računanjem razlike između slika F i B dobivamo novu sliku R. Dobivena slika R je crno-bijela što znači da joj svaki piksel poprima vrijednosti između 0 i 255.

Vrijednost piksela pR(x,y) slike R računamo ovako: pR(x,y) = max(255, |pF,red(x,y)-pB,red(x,y)| + |pF,green(x,y)-pB,green(x,y)| + |pF,blue(x,y)-pB,blue(x,y)|). Na dobivenoj slici R sada su pikseli pozadine tamniji a pikseli objekta svjetliji.

Osim ove metode probali smo i sličnu metodu koja je dala vrlo sličan ali ipak nešto lošiji rezultat: pR(x,y) = max(|pF,red(x,y)-pB,red(x,y)|, |pF,green(x,y)-pB,green(x,y)|, |pF,blue(x,y)-pB,blue(x,y)|).

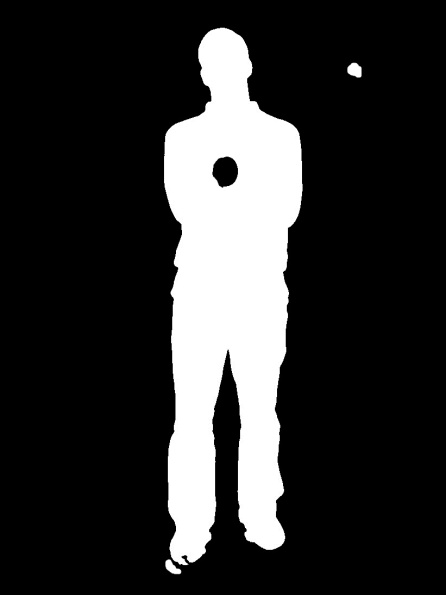
#### Binarizacija slike

Nakon što smo dobili sliku R gdje pikseli poprimaju vrijednosti od 0 do 255 želimo iz nje dobiti binarnu sliku S gdje pikseli siluete poprimaju vrijednost 1 a pikseli pozadine vrijednost 0. Sliku S dobivamo tako da odabiremo granicu g iznad koje sve piksele postavljamo na 1 a ispod na 0. Tu granicu nazivamo granicom binarizacije. S obzirom na to da je moguće da se slike F i B razlikuju u količini osvjetljenja moguće je da vrijednosti piksela pozadine na slici R nisu vrlo bliske nuli već nešto veće. Kako god, pretpostavljamo da je promjena osvjetljenja, ako je do nje došlo, otprilike jednaka na cijeloj slici što znači da bi pikseli pozadine na slici R trebali imati međusobno bliske vrijednosti. Također pretpostavljamo da je veći broj slike pokriven pozadinom nego objektom. Poduprti tim pretpostavkama možemo reći da je prosječna vrijednost piksela pozadine na slici R jednaka medijanu svih piksela slike R, i tu ćemo vrijednost koristiti za izračun granice binarizacije slike. Mogli bismo direktno koristiti medijan kao granicu no tada bismo puno piksela pozadine karakterizirali kao piksele siluete jer pikseli pozadine variraju oko medijana. Zato moramo uvesti dodatnu toleranciju pa granicu binarizacije g računamo kao g = medijan(R) + thresh gdje je thresh parametar tolerancije. Kao optimalna vrijednost tog parametra pokazalo se thresh = 20. Postupkom binarizacije sa granicom g = medijan(R) + thresh dobivamo binarnu sliku S koja sadrži siluetu.

#### Pročišćavanje siluete

Na slici S pojavljuju se dvije vrste pojava koje želimo izbjeći: rupe unutar siluete i otoci.

Otok je skup piksela siluete koji čine cjelinu ali su odvojeni od glavne cjeline piksela siluete. Otoci mogu nastati odvajanjem dijelova siluete ili pogrešnim prepoznavanjem dijela pozadine.



Slika : Primjer slike sa rupama i otocima

Prvo uklanjamo rupe unutar siluete i to tako da nađemo konture siluete te zatim sve piksele unutar kontura označimo kao piksele siluete.

Zatim uklanjamo otoke tako da identificiramo sve odvojene cjeline piksela siluete te uklonimo sve osim najveće jer za najveću pretpostavljamo da je silueta. Za identificiranje odvojenih cjelina piksela i računanje njihove veličine koristimo popunjavanje poplavom (engl. flood fill).

Kao konačni rezultat dobivamo binarnu sliku S gdje su pikseli siluete označeni sa 1 a pikseli pozadine sa 0. Zajamčeno je da svi pikseli siluete čine jednu cjelinu i da nema rupa unutar siluete.

Opisani postupak pokazao se efikasnim na slikama slikanim u zatvorenom prostoru sa dvo-dijelnom pozadinom, dok je za slike na otvorenom sa kompleksnom pozadinom davao nešto lošije rezultate. Velike probleme također uzrokuju osobe čija je odjeća slična boji pozadine. Naime zbog zaglađivanja i sličnih mjera kojima postupak postaje robustan na šum i promjene u osvjetljenju postupak također teže raspoznaje manje razlike boje pa se često dogodi da odjeću sličnu pozadini označi kao pozadinu pri čemu dobivene siluete postaju u velikoj mjeri neiskoristive.

## Računanje Granlundovih koeficijenata

Budući da je izlaz prethodnog koraka binarna slika, sljedeći korak je stvaranje konture oko bijelog područja na slici, to jest osobe. Kontura se zapisuje jednodimenzionalnom kompleksnom periodičnom funkcijom koja crpi vrijednosti iz skupa kompleksnih brojeva:

gdje je točka na konturi, a broj točaka konture.

Funkciju moguće je izraziti Fourierovom transformacijom. Koeficijenti Fourierove diskretne transformacije su tada

a funkcija je jednaka

Budući da su Fourierovi koeficijenti zavisni, ne samo o obliku konture, već i o izboru početne točke, translaciji, rotaciji i povećanju konture, G. H. Granlund je u svom članku [1] definirao funkcije Fourierovih koeficijenata koje eliminiraju sve te utjecaje:

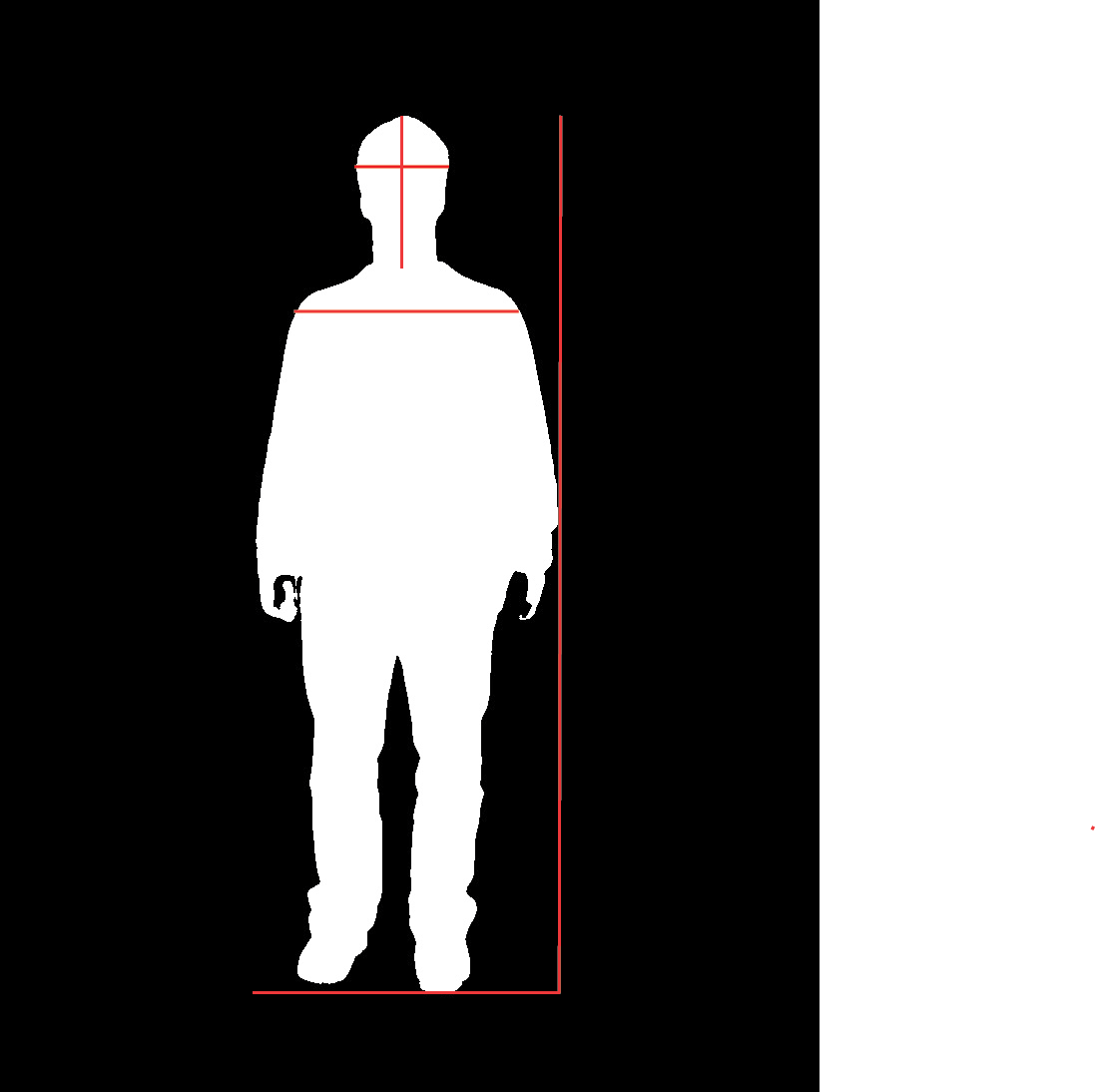
gdje , a

Granlundovi koeficijenti zavise o obliku konture.

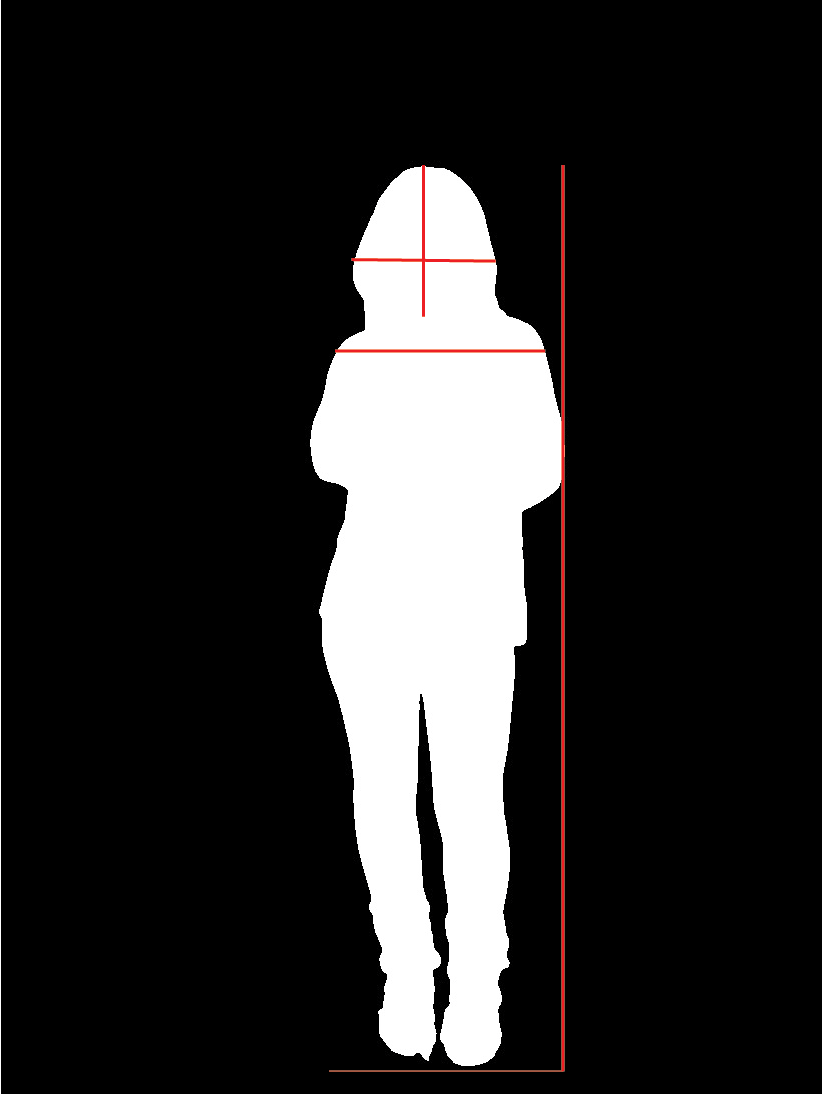
U fazi učenja računaju se Granlundovi koeficijenti za sve slike iz seta za učenje. Korišteni Granlundovi koeficijenti su , , , , , , , , , , , , , , , , , , , . Izračunati koeficijenti predstavljaju vektor značajki za svaku sliku i spremaju se u mapu.

## Klasificiranje na osnovu proporcija tijela

Glavne značajke koje se mogu izlučiti iz silueta su svakako visina i širina siluete. Jedna od najvažnijih značajki je visina jer je njome vrlo lako razlikovati dvije siluete, već i za relativno male razlike značajki. No, osim same visine i širine siluete, izlučujemo još i visinu glave, širinu glave, te širinu ramena. Naš sustav radi neovisno o udaljenosti kamere od osobe pri snimanju, zato jer se sve vrijednosti značaji mogu podijeliti sa visinom, te tako se barata sa omjerima koji su neovisni o udaljenosti kamere ili fotoaparata.



Slika 7. Glavne značajke na muškoj silueti



Slika 8. Glavne značajke na ženskoj silueti

Crvenom bojom su označene značajke izlučene iz siluete, to su visina siluete, širina siluete, visina i širina glave, te širina ramena. Na gornje dvije slike vise se razlika uočljiva čak i golim okom da postoje znatne razlike u određenim značajkama ovih dvaju silueta. Tu činjenicu smo iskoristili pošto je naš skup relativno malen te već i sama značajka visine siluete znatno doprinosi razlikovanju osoba, a uz ostale značajke moguće je relativno dobro klasificiranje.

Samo visinu iz siluete izlučujemo tako što nađemo najviši supac bijelih piksela, dok za širinu pronalazimo najdulji redak bijelih piksela. Glavu nalazimo krečući s vrha slike siluete dok ne naiđemo na bijele piksele, tada pratimo porast u širinama bijelih piksela dok ponovno ne nastane znatan pad pa rast, tj. dok ne naiđemo na vrat. Širinu ramena pronalazimo tako da krenemo odande gdje je proces nalaženja visine glave stao, tj od vrata. Zatim se širimo prema dolje dok promjena u širini ne postane konstantna, tj. dok ne naiđemo na dio tijela koji je ravan.

Pokazalo se da je najbolja značajka od ovih navedenih visina, vjerojatno prije svega jer se radilo na relativno malom skupu podataka, pa se visina osoba znatno razlikuje. Rezultati sigurno ne bi bili ni približno dobri samo temeljem visina i širina raznih dijelova tijela siluete mjereno na velikoj populaciji.

Osim toga testirali smo i sa ovih 5 značajki ali još podijeljinih sa visinom, tako smo dobili neovisnost o udaljenosti od kamere pri snimanju siluete a rezultati, odnosno točnosti su i dalje poprilično slično, no o tome piše više pod rezultatima.

Izlučene značajke potom koristimo za klasificiranje, te koristimo nekoliko klasifikatora za rješavanje tog problema:

- obično uspoređivanje sa kvadratnom greškom pri kažnjavanju

- Bayesova mreza

- SVM

- k najbližih susjeda

- Random Forest

## PCA

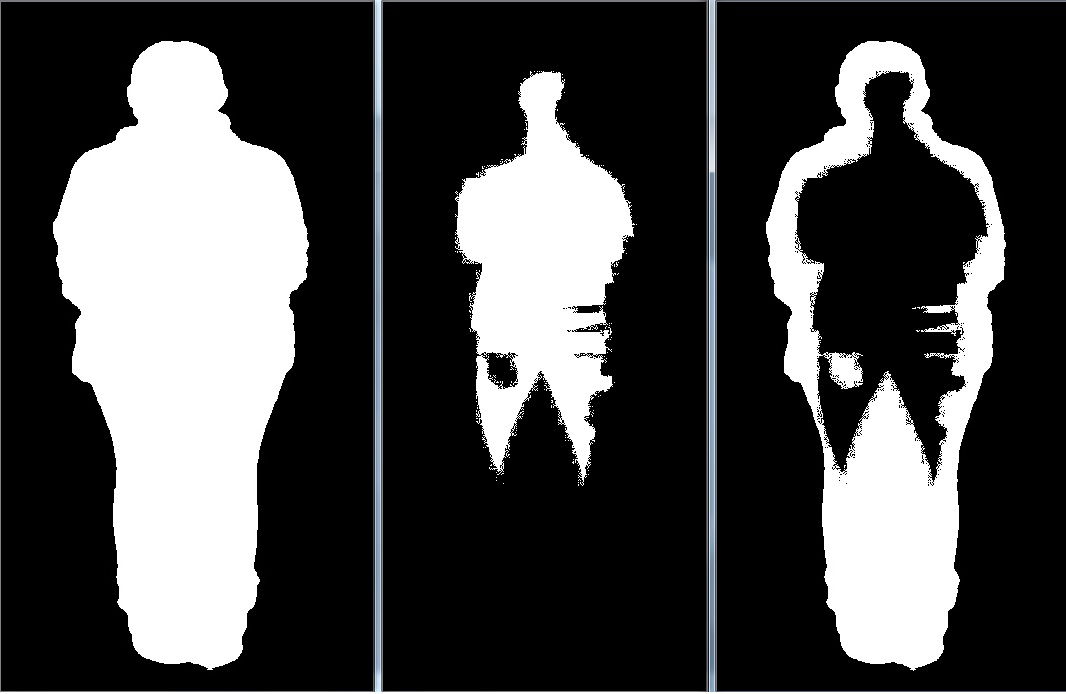
Kod slika silueta osoba koje je potrebno raspoznavati u sklopu ovog projekta može se primijetiti velika sličnost između određenih slika. Iz tog razloga metoda raspoznavanja usporedbom elementa slike po elementu slike nije odgovarajuća. Kod takvih slika se javlja šum kao posljedica nesavršenosti uređaja za snimanje, varijacija u položaju, orijentaciji osobe prilikom snimanja itd. Korištenjem metoda analize glavnih komponenti (*principal component analysis-* PCA) izbjegao bi se utjecaj ranije navedenih problema. PCA metoda se temelji na smanjenju informacije koje su sadržana u slici, a da se pri tome zadrže informacije o najvažnijim značajkama. PCA se koristi nad jednodimenzionalnim podacima, a budući da slika sadrži informacije u dvije dimenzije, potrebno ju je prikazati pomoću vektor-stupca, što za sliku dimenzija NxN znači prikaz značajki u N2-dimenzionalnom prostoru. Metodom PCA traži se prostor (skup ortogonalnih vektora) manjih dimenzija od originalnog prostora u kojem će vektori slika biti postavljeni tako da varijacija podataka bude maksimalna. Matematički rečeno, prikaz slike u novom prostoru dobiva se množenjem ortogonalne matrice **V** i matrice slike **X** prema formuli:

(1)

Unutar matrice **Y** se odabiru samo oni vektori u čijem je smjeru varijanca najveća. Pomoću PCA iz skupa slika za treniranje moguće je dobiti skup najvažnijih značajki unutar skupa za treniranje te skup istih slika predočenih pomoću tih značajki. Također, i ulazna slika koju je potrebno klasificirati se mora transformirati u isti prostor značajki. Skup za treniranje je matrica čiji su stupci slike siluete koje su pretvorene u vektore. Da bi se zadržale sve bitne značajke po kojima se slike razlikuju, potrebno je oduzeti srednju vrijednost svih slika – „prosječnu sliku“.



Slika 9. Rezultat centriranja slike



Slika 10. Rezultat usrednjivanja slike

Nakon toga potrebno je izračunati korelacijsku matricu **CX** koje se računa prema:

 (2)

gdje **X** predstavlja matricu sa vektorima slika za treniranje od kojih je oduzeta srednja vrijednost („prosječna slika“).



U korelacijskoj matrici **C** svaki pojedini element cij predstavlja međusobne korelacije pojedinih elemenata slike vektora slike i vrijednosti tog elementa slike, dok su elementi na glavnoj dijagonali cii varijance tog elementa slike, odnosno njihovih vrijednosti. Nakon što je izračunata matrica korelacije **C** može se pronaći matrica transformacije koja će najbolje projicirati matricu **X** na novi prostor značajki.

Matrica transformacije **V** se dobiva tako da se izračunaju svojstveni vektori korelacijske matrice **C** te poredaju u stupce s obzirom na odgovarajuće svojstvene vrijednosti tako da oni svojstveni vektori s najvećom svojstvenom vrijednošću budu prvi i oni predstavljaju najvažnije značajke u slici. Svaki svojstveni vektor predstavlja jednu siluetu.

Konačno kako bi se završila faza treniranja potrebno je projicirati vektor slike za treniranje na svojstvene vektore i spremiti te projekcije kao vektor značajki. Od tih značajki odabire se prvih nekoliko koje predstavljaju one najvažnije značajke. Time se postiže da se dimenzionalnost značajki smanjila na puno manje, ovisno o tome koliko je dimenzija potrebno za što bolju klasifikaciju uzimajući u obzir i vremensko trajanje klasifikacije.

Kako bi mogli uspoređivati ulaznu sliku sa skupom slika za treniranje potrebno je ulaznu sliku (od koje je bila oduzeta srednja vrijednost svih slika) pomnožiti sa spremljenim vektorom značajki svih slika i dobiju se značajke za tu određenu sliku koje se onda uspoređuju s ostalima i određuje se najsličnija slika.

## Template matching

Ova metoda radi na temelju sličnosti dviju slika u vrijednostima slike. Budući da je naša slika siluete crno bijela to je lako izvedivo. Metoda prolazi kroz cijelu testnu sliku i uspoređuje piksele sa slikom iz skupa za učenje. Tako prolazi svim slika iz skupa za učenje te pronalazi k najsličnijih slika.

# Ispitivanje rješenja

(do 10 stranica)

## Ispitna baza

Za potrebe ovog projekta, odlučili smo izraditi vlastitu bazu slika pošto nismo uspjeli naći postojeću bazu koja bi zadovoljavala naše potrebe.

Baza sadrži 248 slika od ukupno 24 osobe. Slike su uslikane digitalnim fotoaparatom razlučivosti 3648 x 2736 piksela postavljenim na visini od 156 cm i udaljenim 319 cm od osobe. Slike su naknadno reducirane na 30% originalne veličine i takve spremljene u bazu, što rezultira slikama razlučivosti od 821 x 1094 piksela. Takva razlučivost pokazala se dovoljno dobrom za izlučivanje kvalitetne siluete, također i značajno smanjuje ukupnu veličinu baze.

Prilikom izrade baze bio nam je cilj imati slike više različitih poza svake osobe. Svaka osoba slikana je frontalno. Također, prije svake slike osoba je zamoljena da ponovno uđe u kadar kako bismo osigurali prirodnost i nepovezanost uzastopnih snimki iste osobe.

Većina slika napravljena je u zatvorenom prostoru, a dodatno smo za još osam osoba (članovi tima) napravili dio slika na otvorenom kako bismo testirali uspješnost sustava. Ipak, pokazalo se kako tako dobivene siluete nisu zadovoljavajuće kvalitete i stoga te slike nisu korištene u učenju ni ispitivanju sustava. Slike su ostavljene u bazi za svrhe budućeg rada.

Osim samih slika, baza se sastoji i od silueta izlučenih ih iz njih. Iz svake slike je jednom izlučena silueta, i sustav dalje cijelo vrijeme radi samo sa siluetama. Iz nekih slika nije bilo moguće izlučiti siluetu zadovoljavajuće kvalitete, stoga takve slike nisu korištene.

## Rezultati učenja i ispitivanja

### Rezultati - Granlundovi koeficijenti

Uspoređivanje bez korištenja PCA:

Točnost (1 najbolji): 11/56 = **0.196429**

Točnost (2 najbolja): 18/56 = **0.321429**

Točnost (3 najbolja): 24/56 = **0.428571**

Točnost (4 najbolja): 27/56 = **0.482143**

Točnost (5 najboljih): 34/56 = **0.607143**

Točnost (6 najboljih): 36/56 = **0.642857**

Točnost (10 najboljih): 46/56 = **0.821429**

Točnost (15 najboljih): 56/56 = **1**

Uspoređivanje s korištenjem PCA:

Točnost (1 najbolji): 15/56 = **0.267857**

Točnost (2 najbolja): 19/56 = **0.339286**

Točnost (3 najbolja): 26/56 = **0.464286**

Točnost (4 najbolja): 31/56 = **0.553571**

Točnost (5 najboljih): 35/56 = **0.625**

Točnost (6 najboljih): 36/56 = **0.642857**

Točnost (10 najboljih): 47/56 = **0.839286**

Točnost (16 najboljih): 56/56 = **1**

### Rezultati - Body proportions matching

Uspoređivanje uz kvadratnu grešku kod kažnjavanja:

Correctness (only the best): 35/56 = **0.625**

Correctness (in **2** best): 41/56 = **0.732143**

Correctness (in **3** best): 49/56 = **0.875**

Correctness (in **4** best): 53/56 = **0.946429**

Correctness (in **5** best): 53/56 = **0.946429**

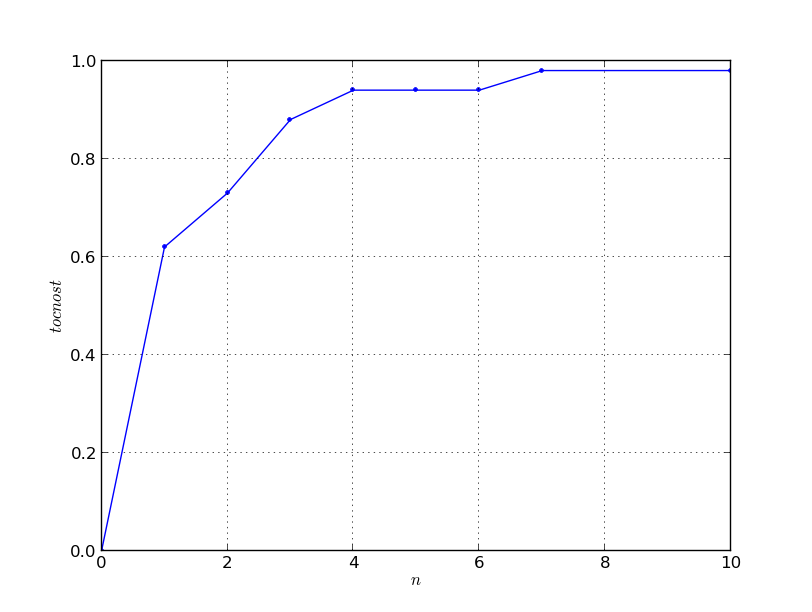
Correctness (in **6** best): 53/56 = **0.946429**

Correctness (in **7** best): 55/56 = **0.982143**

Correctness (in **8** best): 55/56 = **0.982143**

Correctness (in **9** best): 55/56 = **0.982143**

Correctness (in **10** best): 55/56 = **0.982143**



Slika 11. Točnost u ovisnosti o broju najsličnijih slika

-ukoliko sve značajke podijelimo sa visinom, a i dalje koristimo kvadratnu grešku:

Correctness (only the best): 30/56 = **0.535714**

Correctness (in **2** best): 42/56 = **0.75**

Correctness (in **3** best): 45/56 = **0.803571**

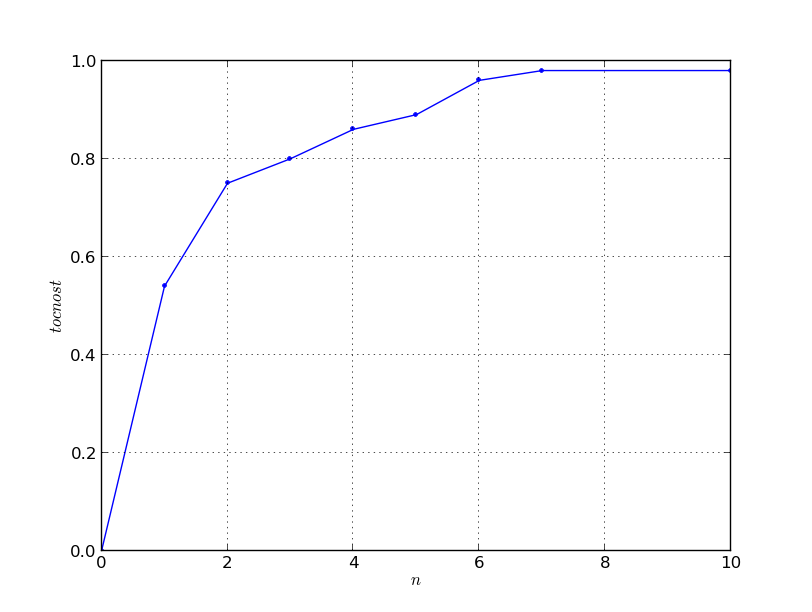
Correctness (in **4** best): 48/56 = **0.857143**

Correctness (in **5** best): 50/56 = **0.892857**

Correctness (in **6** best): 54/56 = **0.964286**

Correctness (in **7** best): 55/56 = **0.982143**

Correctness (in **10** best): 55/56 = **0.982143**



Slika 12. Točnost u ovisnosti o broju najsličnijih slika

K - najbližih susjeda

Correctness (only the best): 30/56 = **0.535714**

Bayes

Correctness (only the best): 4/56 = **0.0714286**

SVM

Correctness (only the best): 4/56 = **0.0714286**

Random Forest

Correctness (only the best): 28/56 = **0.5**

### Rezultati - Template matching

Correctness (only the best): 25/56 = **0.446429**

Correctness (in **2** best): 28/56 = **0.5**

Correctness (in **3** best): 34/56 = **0.607143**

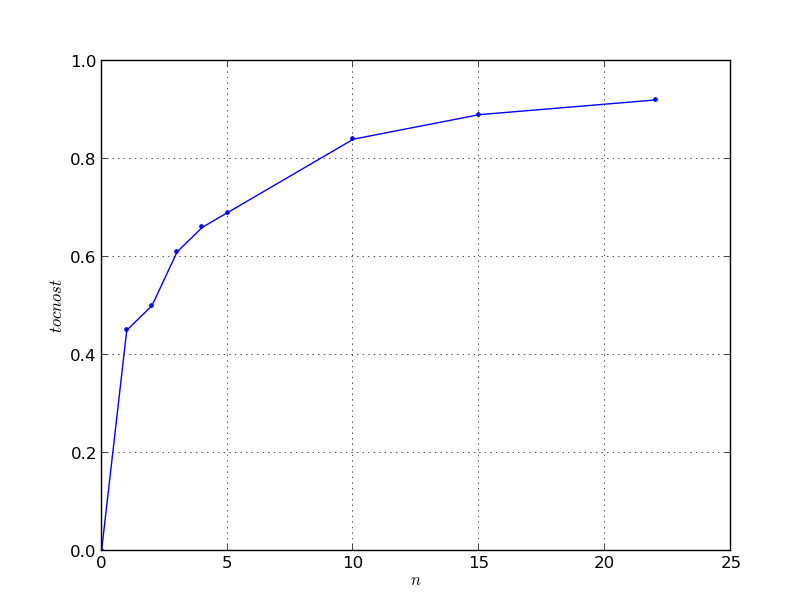
Correctness (in **4** best): 37/56 = **0.660714**

Correctness (in **5** best): 39/56 = **0.696429**

Correctness (in **10** best): 47/56 = **0.839286**

Correctness (in **15** best): 50/56 = **0.892857**

Correctness (in **22** best): 52/56 = **0.928571**



Slika 13. Točnost u ovisnosti o broju najsličnijih slika

### Rezultati - PCA

Rađeno je testiranje na skupu od 44 slika za učenje (4 slike po osobi) i 11 slika za testiranje (1 slika po osobi). Presjek tih skupova je naravno prazan skup. Implementacija za svaku testnu sliku izbacuje najsličniju u skupu slika (silueta) za testiranje. Dobivena uspješnost je 63%.

## Analiza rezultata

Ukupan skup podataka, odnosno slike iz baze, podijeljen je na skup za učenje i skup za ispitivanje. Ukupno je 56 slika odabrano za testiranje, tako da je za svaku osobu odabran reprezentativan skup od dvije do četiri slike.

Rezultat testiranja koji nas zanima jest točnost koja se može izraziti kao udio točnih klasifikacija u ukupnom broju klasifikacija. S obzirom na zahtjevnu prirodu problema, dopuštamo da se pojedina klasifikacija proglasi točnom ako se klasa kojoj pripada zadani primjer nalazi u prvih n klasa koje prema izlazu klasifikatora imaju najveću izglednost.

Tako možemo pratiti točnost klasifikatora u ovisnosti o parametru n, koji određuje koliko ćemo najizglednijih klasa uzimati u obzir pri određivanju točnosti.

### Granlundovi koeficijenti

Klasifikacija gledajući samo najbolju sliku je 19.64% bez PCA i 26.79% uz PCA, što je s obzirom na ostale isprobane metode loš rezultat. Glavni problem ovog algoritma je što se apsolutno oslanja na oblik tijela, a u bazi postoji više svaka osoba ima više različitih položaja tijela. Zbog toga algoritam uglavnom spaja iste položaje tijela, a ne iste osobe. Granlundovi koeficijenti bili bi dobro rješenje kad bi se baza sastojala od slika osoba u istom položaju. Također, vidi se da korištenje PCA pomaže, ali ne značajno. Metoda je još uvijek lošija od ostalih.

### Body proportions matching

Dakle, najbolji rezultat daje obično uspoređivanje značajki uz kažnjavanje kvadratnim odstupanjem, približno jednake rezultate daje metoda k najbližih susjeda. Metode kao što su SVM ili Bayes imaju dosta nisku točnost, to možemo objasniti činjenicom da u skupu podataka nema normalne razdiobe jer je skup premali. Te metode bi vjerojatno bile znatno uspješnije da su testirane na slučajnom skupu uzoraka od nekoliko tisuća ljudi, a s druge strane na tako velikom skupu podataka, metode običnog uspoređivanja i kNN bi imali jako loše rezultate. Pošto je skup podataka, tj silueta dovoljno mali, značajke kao što su visina i širina siluete te dimenzije pojedinih dijelova tijela su već dovoljne za relativno upješno klasificiranje odnosno meko raspoznavanje osobe na temelju siluete. Činjenica leži u tome da već i mala odstupanja u visini bivaju dobro razlikovana te metode usporedbi značajki daju dobre rezultate. Tako metoda kažnjavanje kvadratnom greškom daje točnost od 62.5% koristeći samo pet značajki ( visina siluete, širina siluete, visina glave, širina glave i širina ramena ), a ako gledamo odgovara li određena silueta u 3 najbolje tada točnost raste na 87.5%. Za četiri najbolja raste na dosta sigurnih 94.64%. Metoda k najbližih susjeda i Random Forest imaju nešto veću preciznost od 50%, dok SVM i Bayesov klasifikator imaju točnost manju od 10%.

Jako je zanimljiva činjenica da ako sve značajke podijelimo sa visinom dobivamo gotovo jednako dobre rezultate kao i ako ne koristimo omjere, što je izvrsno jer je tako naš sustav zapravo postao neovisan o udaljenosti s koje je slika odnosno silueta snimljena. Vidimo da za točno klasificiranje uz dijeljenje sa značajkom visine siluete imamo točnost od 53.57% (ovo je i dalje više od svih ostalih metoda, osim kNN s kojim je približno jednak), dok bez dijeljenja ta točnost iznosi 62.5%. No, već za k = 2, odnosno k>=3 rezultati postaju jako sličnih preciznosti preko 90% što je odlično svojstvo sustava.

### PCA

Rezultati su relativno dobri (63 %). Veća točnost bi se mogla dobiti predočavanjem većeg broja slika (silueta) po osobi odnosno povećavanjem skupa za učenje. Povećavanje skupa za učenje predstavlja problem jer nije moguće dobiti veliki skup slika za jednu osobu jer se gledaju samo siluete te ponavljanjem slične pozicije ne dobivamo ništa (dobivamo vrlo sličnu siluetu)

### Template matching

Dobiveni rezultati su za samo jednu najbolju sliku 44% što nije puno. Ali gledajući brzinu i jednostavnost metode to je dovoljno dobar rezultat. Ako gledamo postotak uspješnosti metode u 3 najbolje slike dobivamo uspješnost od 60% što je puno bolje, ali još uvijek nedovoljno dobro.

# Opis programske implementacije rješenja

(do 5 stranice)

Programska implementacija napisana je u programskom jeziku C++ uz korištenje OpenCV knjižnice, popularnog javno dostupnog skupa metoda i algoritama za računalni vid i strojno učenje.

Kod je objektno strukturiran, a osnovu implementacije čini abstraktni razred *Classifier* koji sadrži virtualne metode *learn*() i *classify()*, te već implementiranu javnu metodu *test()*.

Pri implementaciji konkretnog klasifikatora dovoljno je samo naslijediti razred *Classifier* i implementirati dvije spomenute virtualne metode. Pozivom metode *test()* izvršava se automatsko testiranje sustava, čime je osigurano jedinstveno testiranje svih isprobanih algoritama.

Metoda *learn()* prima skup za učenje kao parametar i služi kako bi algoritam stvorio strukture podataka potrebne za klasifikaciju. Ovdje se također odvija izlučivanje značajki iz silueta.

Metoda *classify()* prima primjer (siluetu) koju je potrebno klasificirati, a vraća skup klasa za koje je najizglednije da im primjer pripada, zajedno s numeričkom interpretacijom izglednosti za svaku klasu.

Metoda *test()* poziva prvo *learn()* koji inicijalizira klasifikator, a zatim za svaki primjer iz skupa za učenje zove *classify()*. Na kraju se generira HTML izvještaj koji sadrži matricu zabune, rezultat svake pojedine klasifikacije i točnost klasifikatora.

Izlučivanje značajki kao što su visina i širina siluete, visina i širina glave, te širina ramena dešavaju se u klasi HWMatching koja nasljeđuje klasu Classifier. Klasa Classifier ima već ugrađene metode za automatsko testiranje i iscrtavanje rezultata što znatno olakšava testiranje raznih klasifikatora. Prosljeđivanjem raznih konfiguracijskih varijabli moguće je namjestit kojim klasifikatorom želimo klasificirati značajke iz skupa uzoraka. Moguće je odraditi klasifikacijom Bayesovim klasifikatorom, kNN klasifikatorom, SVM klasifikatorom, Random Forest klasifikatorom ili običnim uspređivanjem kvadratne greške značajki koje se na kraju pokazalo i najtočnijim.

Implementacija Template matching-a kreće sa centriranjem slike. Pronalazi se centar siluete te područje slike na kojem je silueta sa uvjetom da je to područje što ispunnjenije. Nadalje se gleda samo područje slike koje je dobiveno prethodnim korakom, a ne cijela slika. Takve slike se stavljaju u vektor te se prilikom klasifikacije novih slika te nove slike uspoređuju sa slikama u vektoru te se izbacuje k najbližih slika. Uspoređivanje slika se radi brojanjem jednakih piksela kod obje slike.

Implementacija PCA učitava skup slika za učenje u vektor. Nakon toga zahtijeva upis broja komponenti (k) koliko ih želimo da PCA koristi. Na temelju vektora slika za učenje se izračunavaju jedinstvene vrijednosti te se dobiva vektor k najznačajnijih značajki. Na temelju tih k značajki računa se euklidska udaljenost testne slike sa slikama za učenje te se dobiva najsličnija slika.

# Zaključak

Prilikom implementiranja i testiranja moglo se primijetiti zanimljive rezultate i promjene kod isprobavanja različitih metoda. U ovom projektu isprobano je dosta metoda i probano dvije metode za testiranje, posebno je bilo testiranje za PCA metodu raspoznavanja slika, a posebno za ostale metode zbog jednostavnosti i efikasnosti prilikom testiranja.

U fazi učenja na slikama bilo je različitih pristupa od centriranje slike tako da se odabere samo regija od interesa oko same siluete, a višak pozadine se zanemaruje, pa izračun Granlundovih koeficijenata, određivanje širine i visine glave osobe, cijele osobe i širine ramena osobe pa do pronalaska srednje slike i prilagođavanje silueta oduzimanjem srednje slike od ostalih i ubacivanje takvih slika u PCA metodu ili ubacivanje Granlundovih koeficijenata u PCA umjesto cijele siluete. Od klasifikatora koji su korišteni su: Bayesov klasifikator, klasificiranje po principu najbližeg susjeda, NNNNN .Ako bi gledali uspješnosti po pojedinim metodama može se reći da je najuspješnija metoda bila XX i to s YY postotka uspješnosti i to ako bi se gledala samo najsličnija osoba. Ako bi se gledalo da se osoba nalazi u N prvih osoba uspješnost detekcije se povećava i dolazi čak i do X posto uspješnosti raspoznavanja. Nakon te metode po uspješnosti dolazi metoda Y s postotkom od XX posto ako se gleda samo najsličnija osoba, a ZZ posto ako se gleda da se osobo nalazi u skupu od ZZ najsličnijih osoba. Potom slijede sve ostale metode koje obuhvaćaju i PCA i uspoređivanje sličnosti piksel po piksel, Granlundove koeficijente itd. U tablici 1. Prikazane su sve metode i njihova uspješnost u raspoznavanju kada se gleda samo najsličnija osoba, odnosno kojoj klasi pripada testna silueta. Kod PCA koji uzima kao ulaz cijele slike je korišten manji broj testnih slika i zbog toga je postotak dosta viši, ali također je ovisan i o broju testnih slika i broju značajki koji se odabire. U tablici je prikaza postotak s najoptimalnijim izborom značajki.

|  |  |  |  |
| --- | --- | --- | --- |
| Metoda | Klasifikator | Broj testnih slika | Uspješnost |
|  |  | 56 |  |
|  |  | 56 |  |
|  |  | 56 |  |
|  |  | 56 |  |
| PCA/cijele siluete | Najbliži susjed | 11 | 63% |

Tablica 1. Prikaz pojedinih metoda i njihove uspješnosti u raspoznavanju

Daljnji razvoj ovog projekta i njegove implementacije obuhvaća razvoj nekog korisničkog sučelja koje će uljepšati i olakšati korištenje implementiranih metoda. Također postoje još dorade vezano uz same metode kao npr., za PCA probati implementirati da kao ulaz u PCA metodu bude broj bijelih piksela po redcima ili stupcima, pa probati pronaći još neke značajke osim samo visine i širene glave i cijele osobe i njezinih ramena, kao što je recimo površina koja zauzima silueta osobe pa dužina mogu itd. Uglavnom postoji još dosta različitih pristupa koji bi se mogli upotrijebiti i možda probati još koji klasifikator koji bi možda bolje odjeljivao uzorke koji se javljaju kod silueta.

Uglavnom identifikacija osobe na temelju samo njezine siluete je još dosta neistraženo područje i nema još pronađen najbolji način koji se odmah odrediti najsličniju osobu s vrlo visokim postotkom uspješnosti tako je to područje otvoreno za eksperimentiranje i testiranje. U ovom radu prikazano je nekoliko standardnih metoda koje su koristile uobičajene klasifikatore, a za daljni razvoj se ostavljaju mogućnosti isprobavanja novih metoda izlučivanja značajki i učenja te upotrebu različitih klasifikatora što bi trebalo još dodatno poboljšati rezultate klasificiranja.

# Literatura

1. G. H. Granlund: „Fourier Preprocessing for Hand Print Character Recogniton“, Computers, IEEE Transactions on, C-21, veljača 1972., pp 195 - 201

1. Ime i prezime autora: Naziv časopisa vol. br. godina izdanja, pp od-do (npr. pp 486-492)/knjige/članka/web resursa (s linkom i datumom pristupa web resursu)

...

**DVD/CD**

* kompletan tekst projekta
* izvorni kod projekta
* exe verzija
* readme file – upute za korištenje i pokretanje programa
* baze slika (sve koje su korištene)
* E-oblik članaka koji su korišteni za izradu projekta
* primjeri obrade