

UNIVERZITET U SARAJEVU
ELEKTROTEHNIČKI FAKULTET SARAJEVO

PROJEKTNI ZADATAK 2
-PREPOZNAVANJE OBLIKA I OBRADA SLIKE-

Članovi tima:

Bojadžić Hanna	1421/17311
Čeligija Jasmina	1537/17620
Ćenanović Mustafa	1426/17561
Kubat Ismar	1422/17148

1. Izbor modela za prepoznavanje koji odgovara problemu (grupe sa 4 člana: 2 vrste modela).

Projekat se sastoji iz dva dijela, detekcija lica te klasifikacija boje kože. Za detekciju lica testirana su dva modela, prvi, kaskadni klasifikator, te drugi *Aggregated channel features*. Za dalji rad biće odabran onaj koji pokaže bolje performanse. Za klasifikaciju kože primjenjena su tri modela: svm, knn i naivni bayesov klasifikator.

2. Izbor deskriptora koji odgovara problemu (grupe sa 4 člana: 2 vrste deskriptora) .

Za detekciju lica isprobane su 2 vrste deskriptora: HOG (Histogram of oriented gradients) i Haar. Za klasifikaciju na osnovu boje kože odabrani su deskriptori bazirani na srednjoj vrijednosti boje.

3. Izbor metoda poboljšavanja iz 1. Projektnog zadatka koje će biti primijenjene nad slikama. Rad nastaviti nad poboljšanim slikama.

Za poboljšanje su odabrane metode poboljšanja kontrast i osvjjetljenje te su oba metoda poboljšanja primjenjena nad slikama.

4. Kreiranje deskriptora svih slika (dalje se oni koriste u treniranju i detekciji) Ovo uključuje i svu neophodnu obradu slika prije samog kreiranja deskriptora (resize-anje, konvertovanje u grayscale isl.)

Za klasifikaciju boje kože kreirana su dva deskriptora na bazi boja. Jedan od njih nalazi srednju vrijednost parametara R, G i B te vraća vektor 3 vrijednosti. Ovaj deskriptor je u funkciji `rgbDesc(image)` i poziva se nad slikom. Drugi deskriptor vraća jednu vrijednost koja daje informaciju o srednjoj vrijednosti boje slike. Ova funkcija se poziva kao `rgbDesc2(image)` nad slikom. Unutar skripti za klasifikatore (primjer nekih od njih: `svmclassifier.m`, `svmclassifierWithoutOutliers.m`, `knnclass.m`) pozivaju se ovi deskriptori nad svim slikama iz training i test seta i generišu se varijable koje predstavljaju deskriptore koji se koriste u daljoj obradi.

Urađen je `resize` slika prije dalje obrade, ovo je urađeno skriptom `resize.m`.

5. Kreiranje i treniranje modela za prepoznavanje.

Unutar skripti za klasifikatore (svmclassifier.m, knnclass.m, bayesclassifier.m) kreirani su modeli za klasifikaciju te su trenirani setom slika iz training seta.

Korištena su tri modela:

Svm klasifikator koristeći matlab funkciju fitecoc:

```
classifierSvmWithOutlier = fitcecoc(trainingFeatures, trainingLabels);
```

Bayesov naivni klasifikator koristeći matlab funkciju fitcnb:

```
classifierBayesWithOutliers = fitcnb(trainingFeatures, trainingLabels);
```

Knn klasifikator koristeći matlab funkciju fitcnn:

```
classifierKnnWithOutliers = fitcnn(trainingFeatures, trainingLabels);
```

Klasifikatore kreiramo pozivom skripti: svmclassifier.m, knnclass.m i bayesclassifier.m.

6. Testirati model sa testnim podacima. Izračunati performanse modela: sp, sens, acc-

Urađena je predikcija sa testnim setom koristeći funkciju predict iz matlaba za svaki klasifikator:

```
predictedLabels2 = predict(classifierKnnWithOutliers, testFeatures);
```

Nakon toga kreiran je tabelarni prikaz rezultata predikcije za svaki klasifikator:

```
confMat = confusionmat(testLabels, predictedLabels2);
```

Nakon toga izračunate su performanse tačnost, specifičnost i senzitivnost po sljedećim formulama:

Tačnost (eng. *accuracy*) - definisanu kao: $ACC = \frac{TP + TN}{UP}$

Senzitivnost (eng. *sensitivity*) - definisanu kao: $SEN = \frac{TP}{OP}$

Specifičnost (eng. *specificity*) - definisanu kao: $SPE = \frac{TN}{ON}$

Performanse svm modela bez poboljšanja performansi:

```
acc =  
    0.9524  
  
sen =  
    0.8889  
  
sp =  
    1  
  
>>
```

Performanse bayesovog naivnog klasifikatora prije poboljšanja performansi:

```
acc =  
    0.9524  
  
sen =  
    0.8889  
  
sp =  
    1
```

Performanse knn modela prije poboljšanja performansi:

```
acc =  
    0.8571  
  
sen =  
    0.8571  
  
sp =  
    0.8571
```

Možemo primjetiti da SVM i bayesov daju bolje performanse nego knn model (prije poboljšanja).

7. Poboljšavanje performansi modela

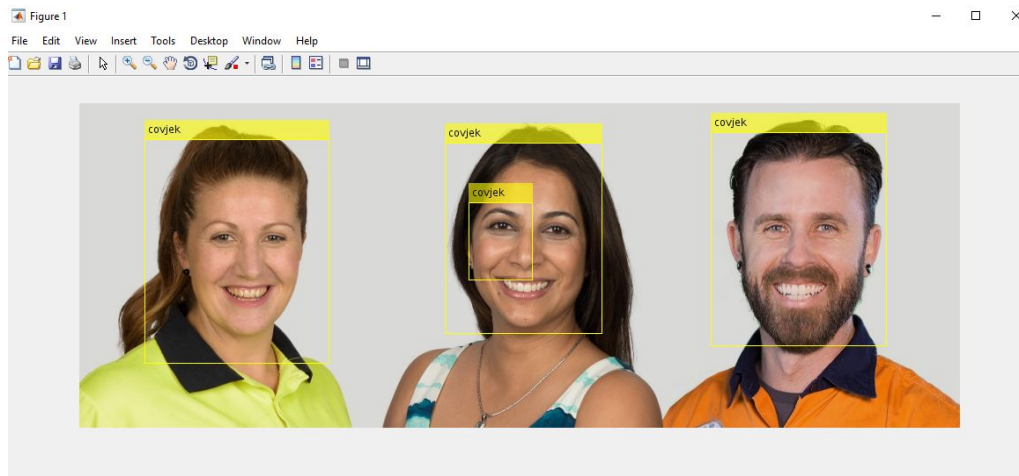
Najprije je urađeno poboljšavanje performansi modela za detekciju lica, jer se njegov izlaz koristi za dalje učenje na osnovu boja te je bitno da detektor lica radi dobro. Nakon toga odrađeno je poboljšanje modela za klasifikaciju boje kože.

Poboljšanje performansi za prvi model za detekciju lica, kaskadni klasifikator:

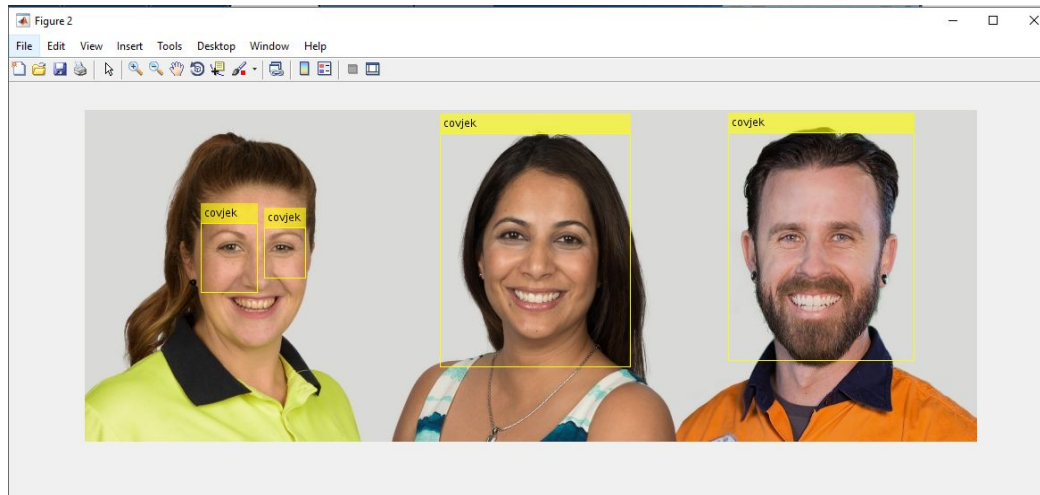
1. Mijenjanje vrijednosti parametara:

Korištena je funkcija *trainCascadeObjectDetector*. Glavni parametar koji utiče na performanse modela je *FalseAlarmRate*. Mijenjajući parametre funkcije optimizirali smo broj detektora ove funkcije za train skup u cilju bolje detekcije. Visoka vrijednost *FalseAlarmRate* dovela je do loših rezultata. Također manji broj nivoa daje lošije rezultate.

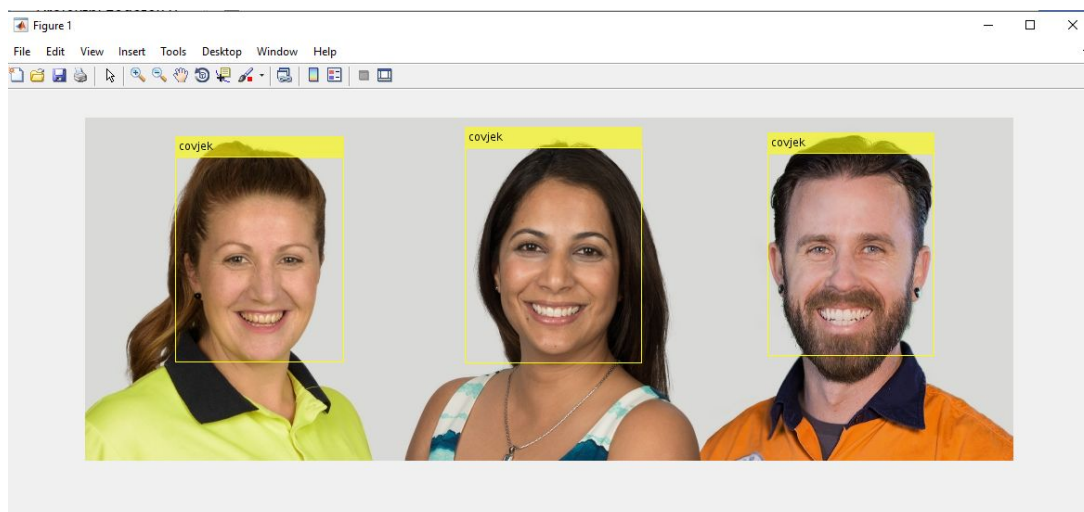
Za *FalseAlarmRate* od 0.2 primjeri reprezentativnih rezultata sa ostalim default-nim parametrima:

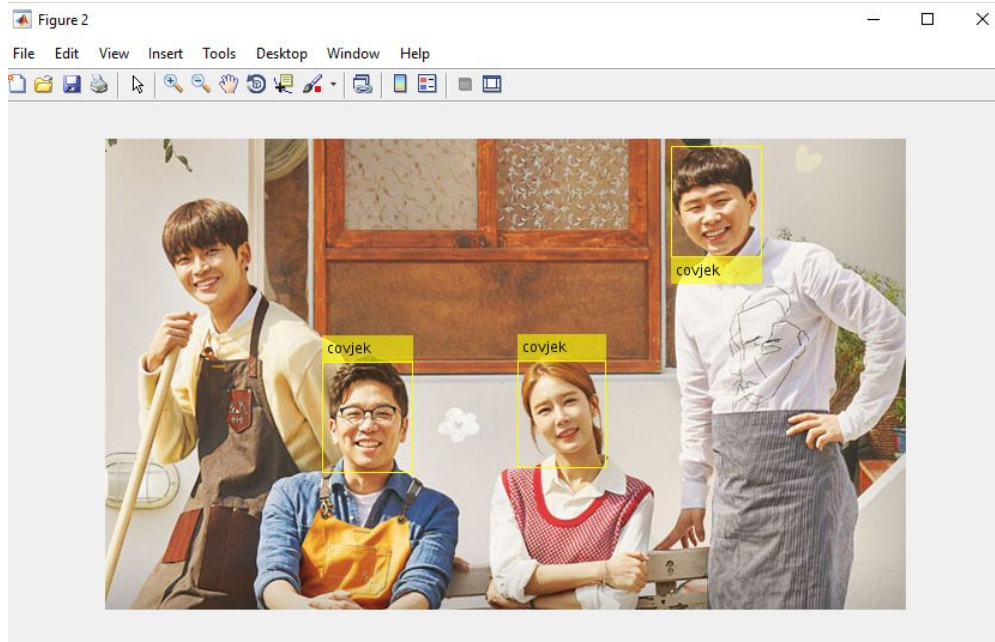


Za *FalseAlarmRate* od 0.7 i defaultne ostale parametre:



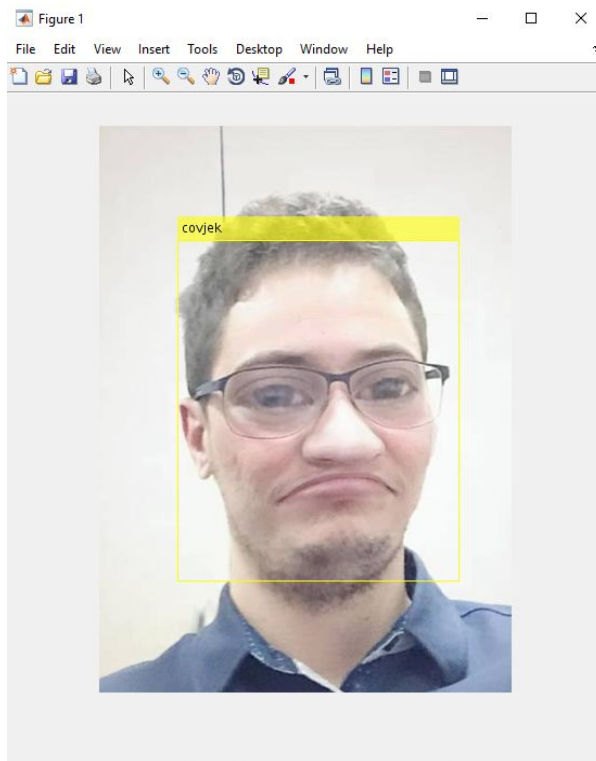
Nakon testiranja za vrijednosti 0.3, 0.4, 0.42, 0.45 i 0.35 nadjena je najbolja vrijednost za *FalseAlarmRate* od 0.35:

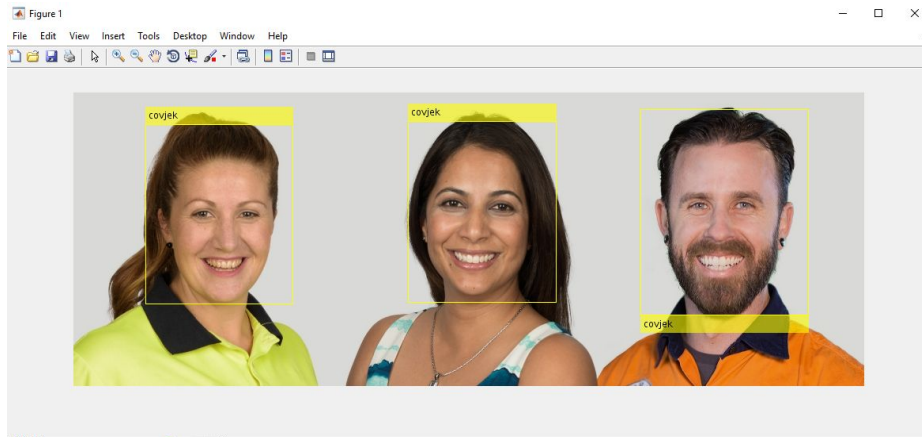




2. Promjena deskriptora:

Primjermom Haar deskriptora dobijamo sljedeće rezultate za parametar 0.35:

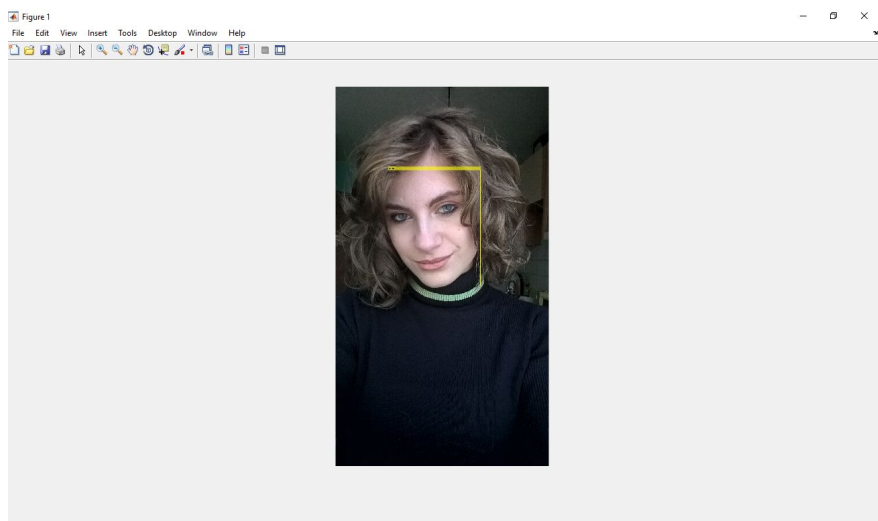
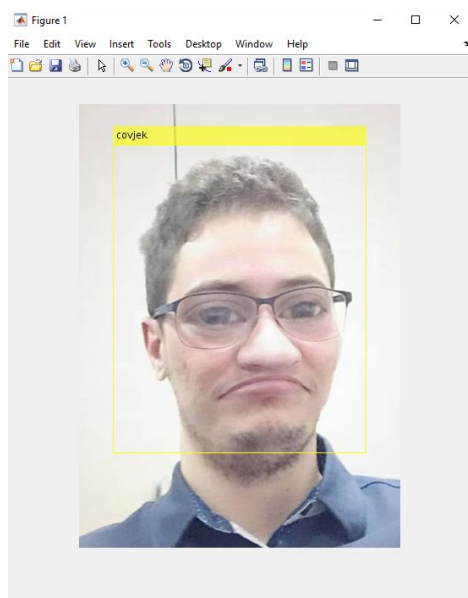




U zadnjem slučaju (i još nekim testiranim slikama) Haar deskriptor daje lošije rezultate, te smo se odlučili u nastavku za detekciju koristiti HOG deskriptor.

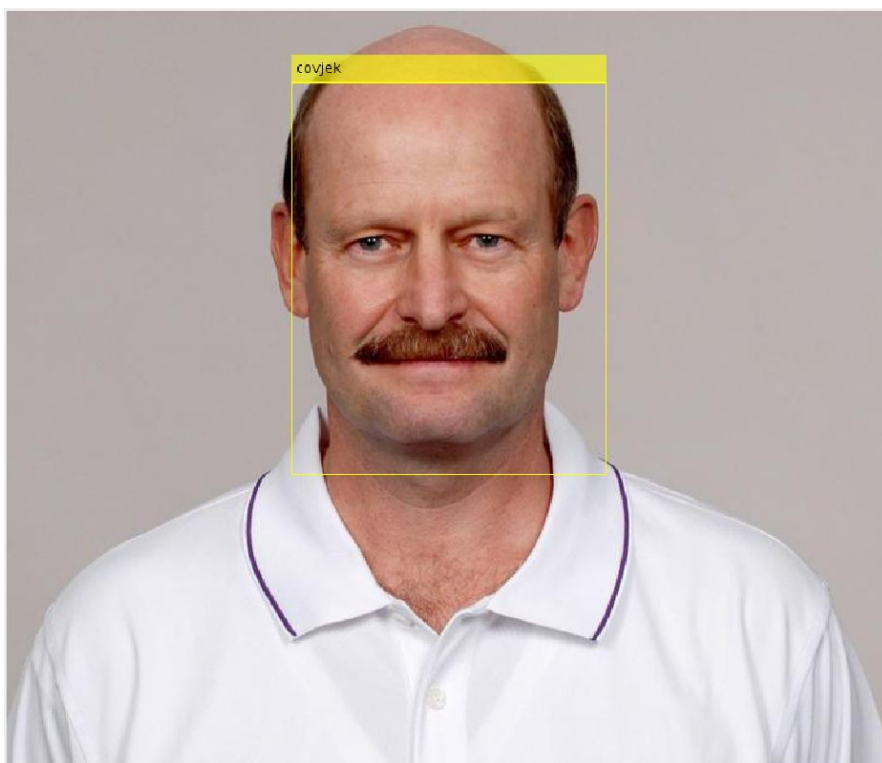
3. Povećanje dataset-a

Klasifikator će bolje trenirati sa većim dataset-om za treniranje te smo povećali dataset. U nastavku su prikazani rezultati testiranja slika za veći dataset. Možemo primjetiti da daje dosta dobre rezultate:



4. Metoda poboljšavanja - resize

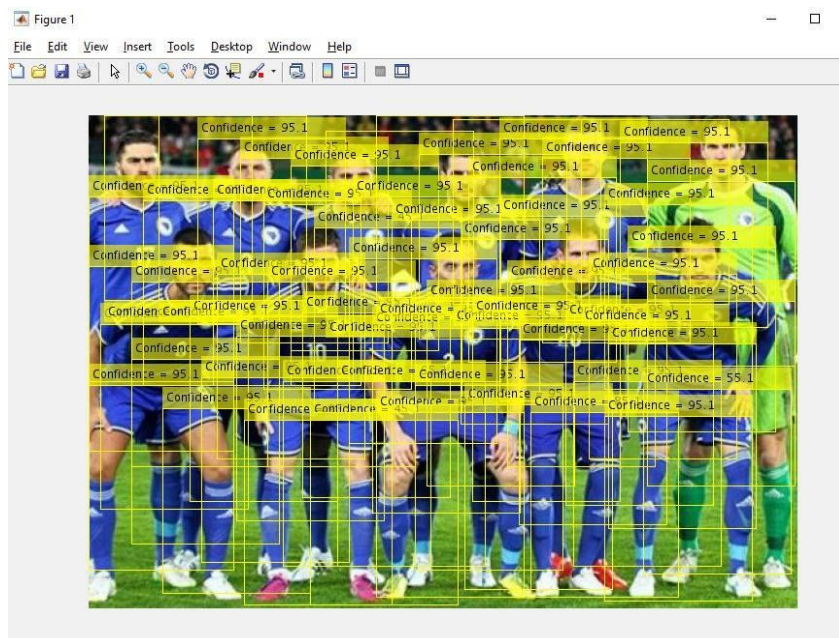
Neke slike su prevelike za ispravno treniranje i detekciju te je odrađen i resize. To je prikazano na sljedećim slikama.



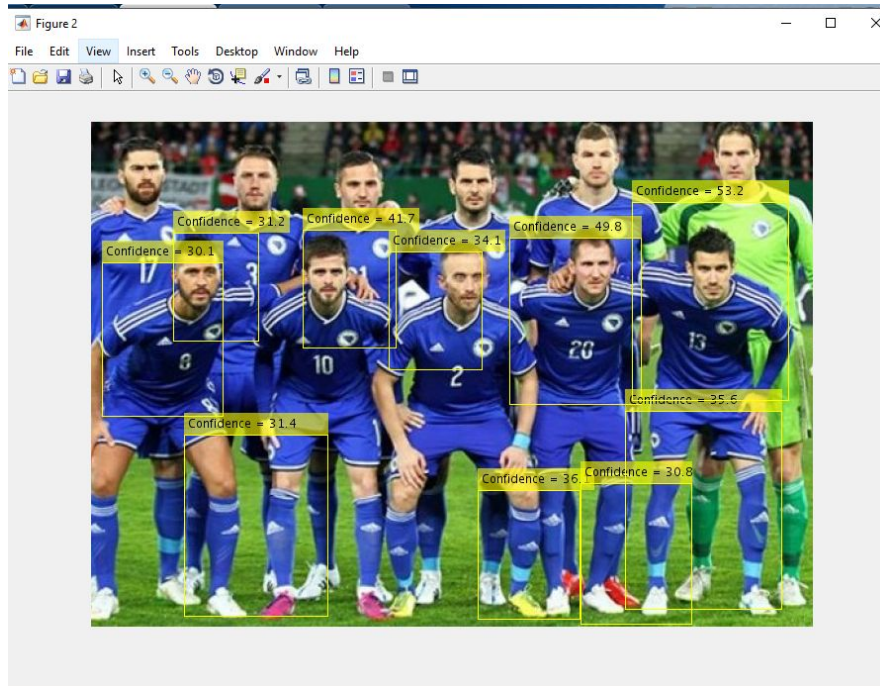
Poboljšanje performansi za drugi model za detekciju lica Aggregated Channel Features:

1. Povećanje dataset-a:

Primjer slike prije povećanja dataset-a:

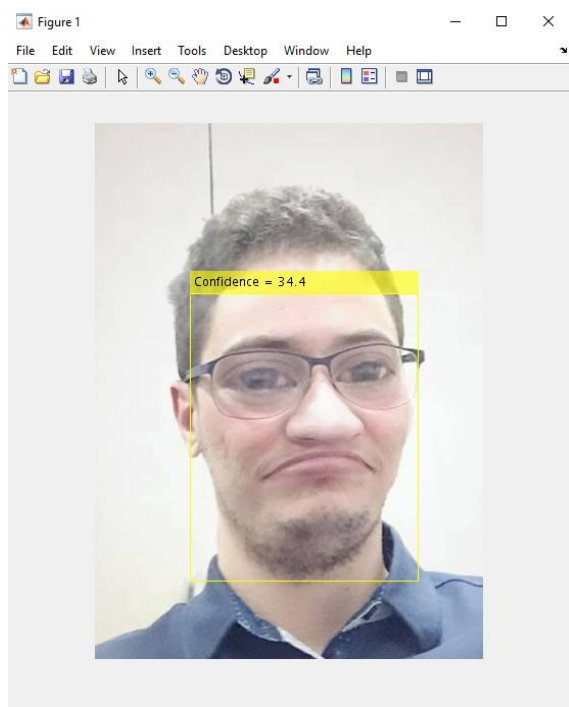
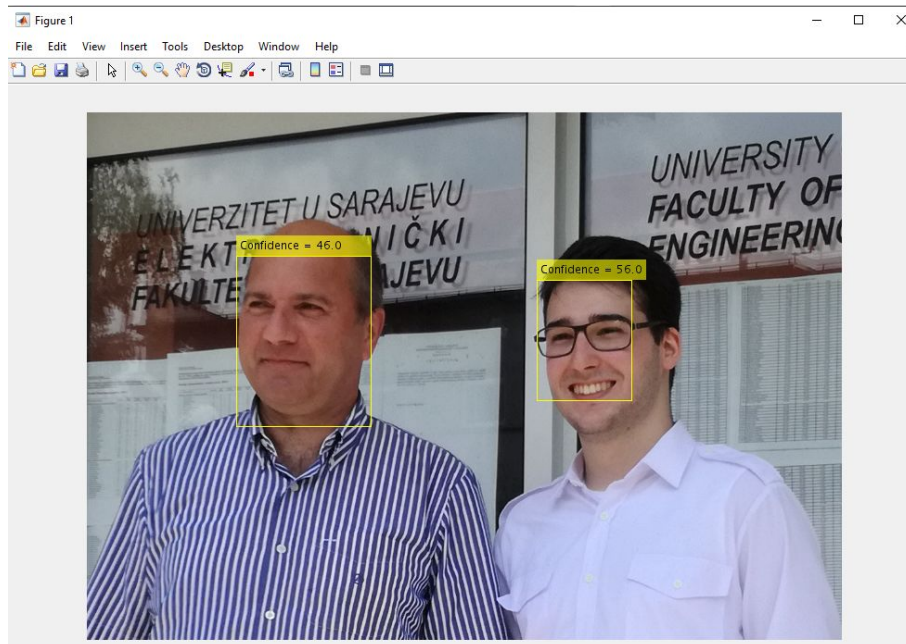


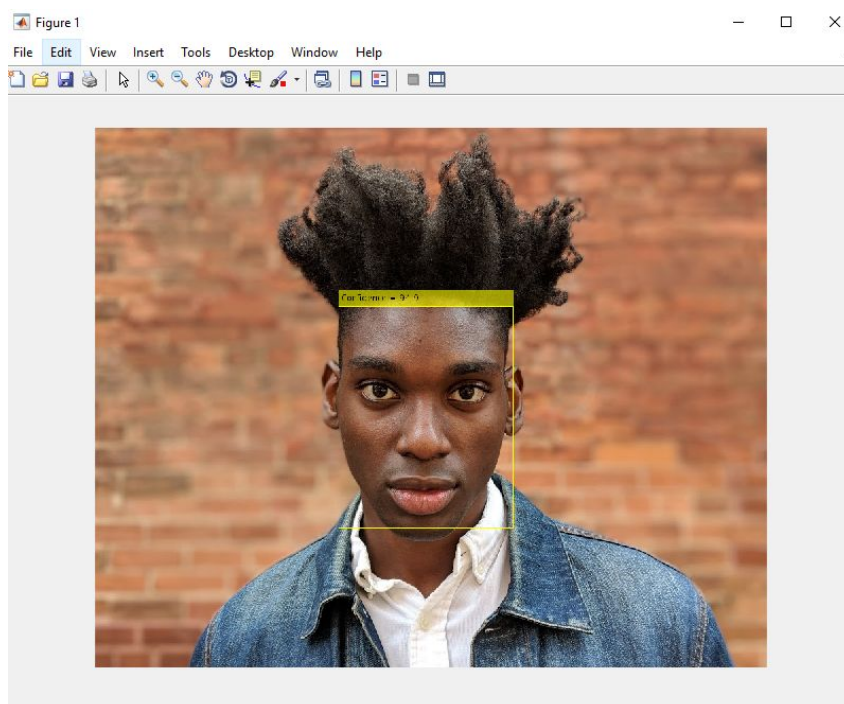
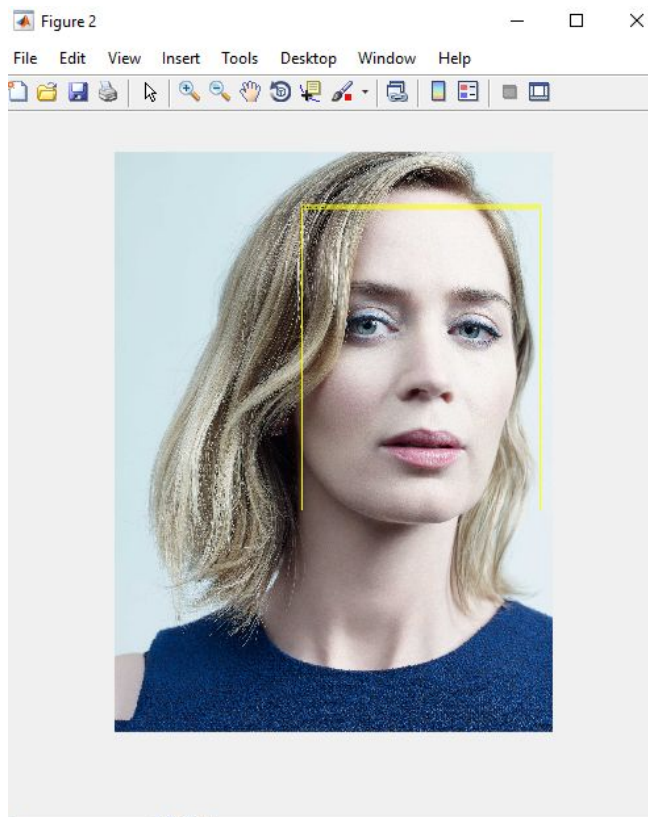
Slika nakon poboljšanja:



Vidimo da u ovom slučaju klasifikator radi mnogo bolje, no i dalje griješi. Možda ovo su mnogo bolji rezultati koji daju indicije da će raditi mnogo bolje sa većim dataset-om.

Na drugim slikama detekcija:





Možemo primjetiti da ovaj model za prepoznavanje mnogo bolje radi za slike na kojima je jedno lice. Prvi model bolje radi za slike sa više lica, te bi njega koristili u te svrhe,

dok model 2 daje bolje rezultate za slike na kojima je jedno lice te slike visoke rezulucije.

Poboljšavanje modela za klasifikaciju kože:

Klasifikatori za boju kože rade jako dobro i bez poboljšanja te sa njima nije urađeno mnogo. Ono što je bitno je da je klasifikacija, i treniranje i testiranje rađeno nad licima izdvojenim modelom za detekciju kože. Ovo osigurava da će biti uzete samo relevantne osobine.

- Izdvajanje outliera

Za svaki od tri modela dodan je dio koji izdvaja outliere. Tačnije unutar deskriptora vrijednost koja drastično odstupa od drugih zamijenjena je srednjom vrijedosti.

To je urađeno na sljedeći način:

```
trainingFeatures = filloutliers(trainingFeatures,'next','median','ThresholdFactor',2)
```

Performanse SVM modela nakon izdvajanja outliera:

```
acc =  
    1  
  
sen =  
    1  
  
sp =  
    1
```

Performanse bayesovog modela nakon izdvajanja outliera:

```
acc =  
    1  
  
sen =  
    1  
  
sp =  
    1
```

Performanse knn modela nakon izdvajanja outliera:

```
acc =  
0.9048
```

```
sen =  
1
```

```
sp =  
0.8667
```

SVM i Bayesov klasifikator nakon izdvajanja outliera daju izuzetno dobre rezultate za klasifikaciju kože. Ovaj dio nalazi se u skriptama `svmClassifierWithoutOutliers.m`, `bayesClassifierWithoutOutliers.m` i `knnClassifierWithoutClassifiers.m`.

- Različita podjela na train i test skup

Izvršena je drugačija preraspodjela slika na test i train skup. Napravljeni su novi folderi te pozvane sve skripte za ove foldere. Ovo je u skriptama `svmClassifierDifferentDataset.m`, `bayesClassifierDifferentDataset.m` i `knnClassifierDifferentDataset.m`.

Performanse SVM klasifikatora nakon promjene dataset-a:

```
acc =  
0.8500
```

```
sen =  
0.7778
```

```
sp =  
0.9091
```

Performanse Bayesovog klasifikatora nakon promjene dataset-a:

```
acc =  
    0.9500
```

```
sen =  
    1
```

```
sp =  
    0.9231
```

Performanse knn klasifikatora nakon promjene dataset-a:

```
acc =  
    0.9500
```

```
sen =  
    0.8889
```

```
sp =  
    1
```

Za promijenjenu raspodjelu na train/test set knn klasifikator pokazuje bolje performanse, dok bayesov i SVM daju lošije rezultate, tako bi od ovih klasifikatora dalje koristili samo knn klasifikator.

8. Export modela za prepoznavanje

Za eksport je napravljena skripta `exportModel.m` unutar koje su spašeni svi modeli. Klasifikatori se spašavaju kao `.mat` datoteke. Prije pokretanja ove funkcije potrebno je istrenirati klasifikatore unutar matlab okruženja pozivom prethodno navedenih skripti. Spašavanje je odrađeno funkcijom `save()` a model se može importovati funkcijom `load`.

Link na bitbucket projekat:

<https://bitbucket.org/hannabojadzic/poos>