

# Računalni program za automatsko određivanje godina i spola osobe na slici

---

**Kralj, Paula**

**Undergraduate thesis / Završni rad**

**2023**

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **University of Zagreb, Faculty of Mechanical Engineering and Naval Architecture / Sveučilište u Zagrebu, Fakultet strojarstva i brodogradnje**

*Permanent link / Trajna poveznica:* <https://urn.nsk.hr/urn:nbn:hr:235:203244>

*Rights / Prava:* [In copyright](#)/[Zaštićeno autorskim pravom.](#)

*Download date / Datum preuzimanja:* **2025-01-02**

*Repository / Repozitorij:*

[Repository of Faculty of Mechanical Engineering and Naval Architecture University of Zagreb](#)



SVEUČILIŠTE U ZAGREBU  
FAKULTET STROJARSTVA I BRODOGRADNJE

# ZAVRŠNI RAD

**Paula Kralj**

Zagreb, 2023.

SVEUČILIŠTE U ZAGREBU  
FAKULTET STROJARSTVA I BRODOGRADNJE

# ZAVRŠNI RAD

Mentor:

Izv. prof. dr. sc. Tomislav Stipančić, dipl. ing.

Studentica:

Paula Kralj

Zagreb, 2023.

Izjavljujem da sam ovaj rad izradila samostalno koristeći znanja stečena tijekom studija i navedenu literaturu.

Zahvaljujem izv. prof. dr. sc. Tomislavu Stipančiću na pomoći tijekom pisanja rada. Također zahvaljujem svim bliskim osobama na podršci tijekom studiranja, a posebno Ivani i Nataliji. Natalija, hvala ti do neba.

Paula Kralj



SVEUČILIŠTE U ZAGREBU  
FAKULTET STROJARSTVA I BRODOGRADNJE

Središnje povjerenstvo za završne i diplomske ispite  
Povjerenstvo za završne i diplomske ispite studija strojarstva za smjerove:  
proizvodno inženjerstvo, računalno inženjerstvo, industrijsko inženjerstvo i menadžment, inženjerstvo  
materijala i mehatronika i robotika



Sveučilište u Zagrebu	
Fakultet strojarstva i brodogradnje	
Datum	Prilog
Klasa: 602 - 04 / 23 - 6 / 1	
Ur.broj: 15 - 1703 - 23 -	

## ZAVRŠNI ZADATAK

Student: **Paula Kralj** JMBAG: **1191239743**

Naslov rada na hrvatskom jeziku: **Računalni program za automatsko određivanje godina i spola osobe na slici**

Naslov rada na engleskom jeziku: **Computer program for automatic determination of age and sex of the person in the picture**

Opis zadatka:

Računalni modeli temeljeni na algoritmima umjetne inteligencije omogućavaju automatiziranje različitih procesa koji koriste podatke, uključujući strojno: prepoznavanje, klasifikaciju, predviđanje, učenje i sl. Metode prepoznavanja su primjenjive u različitim domenama ljudskog djelovanja uključujući prepoznavanje ili detekciju lica, ljudskog djelovanja, različitih objekata i sl.

U radu je potrebno implementirati računalni model neuronske mreže za određivanje godina i spola osoba na slikama. Model mreže je potrebno naučiti koristeći prikladni skup podataka (slika). Računsko rješenje je potrebno temeljiti na *OpenCV* programskoj biblioteci otvorenog koda implementiranoj kroz *Python* programski jezik. Dobiveno softversko rješenje je potrebno eksperimentalno evaluirati uključivši ljudske subjekte u sklopu Laboratorija za projektiranje izradbenih i montažnih sustava. U radu je potrebno navesti korištenu literaturu i eventualno dobivenu pomoć.

Zadatak zadan:

30. 11. 2022.

Zadatak zadao:

Doc. dr. sc. Tomislav Stipančić

Datum predaje rada:

**1. rok:** 20. 2. 2023.  
**2. rok (izvanredni):** 10. 7. 2023.  
**3. rok:** 18. 9. 2023.

Predviđeni datumi obrane:

**1. rok:** 27. 2. - 3. 3. 2023.  
**2. rok (izvanredni):** 14. 7. 2023.  
**3. rok:** 25. 9. - 29. 9. 2023.

Predsjednik Povjerenstva:

Prof. dr. sc. Branko Bauer

## SADRŽAJ

SADRŽAJ .....	I
POPIS SLIKA .....	II
SAŽETAK.....	IV
SUMMARY .....	V
1. UVOD.....	1
2. NEURONSKE MREŽE .....	2
3. PYTHON I OPENCV .....	6
3.1. Python .....	6
3.2. OpenCV.....	7
4. SOFTVERSKO RJEŠENJE .....	9
4.1. Neuronska mreža.....	9
4.2. Priprema slike .....	9
4.3. Određivanje spola .....	11
4.4. Određivanje godina .....	12
4.5. Provjera dobivenih rezultata .....	13
5. EKSPERIMENTALNA EVALUACIJA.....	14
5.1. Ljutnja .....	14
5.2. Prijezir .....	17
5.3. Gađenje .....	19
5.4. Strah .....	21
5.5. Sreća.....	23
5.6. Neutralno.....	25
5.7. Tuga .....	27
5.8. Iznenadenje .....	29
5.9. Osvrt na dobivene rezultate.....	31
6. ZAKLJUČAK.....	32
LITERATURA.....	33
PRILOZI.....	34

**POPIS SLIKA**

Slika 1.	Biološki neuron .....	2
Slika 2.	Umjetni neuron (perceptron) .....	3
Slika 7.	Model neuronske mreže .....	4
Slika 8.	Raspodjela godina osoba na slikama korištenim u neuronskoj mreži .....	4
Slika 4.	Logo programskog jezika <i>Python</i> .....	6
Slika 5.	Python biblioteke .....	7
Slika 6.	<i>OpenCV</i> logo .....	8
Slika 9.	Haarove značajke .....	10
Slika 10.	Vjerojatnosti starosti osobe na slici .....	12
Slika 11.	Prvi primjer ljutnja .....	14
Slika 12.	Drugi primjer ljutnja .....	14
Slika 13.	Treći primjer ljutnja .....	15
Slika 14.	Četvrti primjer ljutnja .....	15
Slika 15.	Peti primjer ljutnja .....	15
Slika 16.	Šesti primjer ljutnja .....	15
Slika 17.	Sedmi primjer ljutnja .....	16
Slika 18.	Osmi primjer ljutnja .....	16
Slika 19.	Prvi primjer prijezir .....	17
Slika 20.	Drugi primjer prijezir .....	17
Slika 21.	Treći primjer prijezir .....	17
Slika 22.	Četvrti primjer prijezir .....	17
Slika 23.	Peti primjer prijezir .....	18
Slika 24.	Šesti primjer prijezir .....	18
Slika 25.	Sedmi primjer prijezir .....	18
Slika 26.	Osmi primjer prijezir .....	18
Slika 27.	Prvi primjer gađenje .....	19
Slika 28.	Drugi primjer gađenje .....	19
Slika 29.	Treći primjer gađenje .....	19
Slika 30.	Četvrti primjer gađenje .....	19
Slika 31.	Peti primjer gađenje .....	20
Slika 32.	Šesti primjer gađenje .....	20
Slika 33.	Sedmi primjer gađenje .....	20
Slika 34.	Osmi primjer gađenje .....	20
Slika 35.	Prvi primjer strah .....	21
Slika 36.	Drugi primjer strah .....	21
Slika 37.	Treći primjer strah .....	21
Slika 38.	Četvrti primjer strah .....	21
Slika 39.	Peti primjer strah .....	22
Slika 40.	Šesti primjer strah .....	22
Slika 41.	Sedmi primjer strah .....	22
Slika 42.	Osmi primjer strah .....	22
Slika 43.	Prvi primjer sreća .....	23
Slika 44.	Drugi primjer sreća .....	23
Slika 45.	Treći primjer sreća .....	23
Slika 46.	Četvrti primjer sreća .....	23
Slika 47.	Peti primjer sreća .....	24
Slika 48.	Šesti primjer sreća .....	24
Slika 49.	Sedmi primjer sreća .....	24

Slika 50.	Osmi primjer sreća .....	24
Slika 51.	Prvi primjer neutralno .....	25
Slika 52.	Drugi primjer neutralno.....	25
Slika 53.	Treći primjer neutralno.....	25
Slika 54.	Četvrti primjer neutralno .....	25
Slika 55.	Peti primjer neutralno .....	26
Slika 56.	Šesti primjer neutralno .....	26
Slika 57.	Sedmi primjer neutralno .....	26
Slika 58.	Osmi primjer neutralno .....	26
Slika 59.	Prvi primjer tuga.....	27
Slika 60.	Drugi primjer tuga .....	27
Slika 61.	Treći primjer tuga.....	27
Slika 62.	Četvrti primjer tuga .....	27
Slika 63.	Peti primjer tuga .....	28
Slika 64.	Šesti primjer tuga.....	28
Slika 65.	Sedmi primjer tuga .....	28
Slika 66.	Osmi primjer tuga.....	28
Slika 67.	Prvi primjer iznenađenje .....	29
Slika 68.	Drugi primjer iznenađenje.....	29
Slika 69.	Treći primjer iznenađenje.....	29
Slika 70.	Četvrti primjer iznenađenje .....	29
Slika 71.	Peti primjer iznenađenje.....	30
Slika 72.	Šesti primjer iznenađenje .....	30
Slika 73.	Sedmi primjer iznenađenje.....	30
Slika 74.	Osmi primjer iznenađenje .....	30
Slika 75.	Neuspješan pokušaj 1 .....	31
Slika 76.	Neuspješan pokušaj 2 .....	31
Slika 77.	Neuspješan pokušaj 3 .....	31
Slika 78.	Neuspješan pokušaj 4 .....	31



---

**SAŽETAK**

Metode strojnog prepoznavanja primjenjive su u raznim područjima ljudskog djelovanja. Tema ovog rada je izrada računalnog programa za automatsko određivanje godina i spola osobe na slici. Rješenje je bazirano na implementiranim modelima neuronskih mreža za određivanje godina i spola te je izrađeno u programskom jeziku *Python* uz korištenje programske biblioteke otvorenog koda *OpenCV*. Predloženo rješenje na kraju je eksperimentalno evaluirano.

Ključne riječi: *Python*, *OpenCV*, neuronske mreže

**SUMMARY**

Machine recognition methods are used in many areas of society. The goal of this thesis is to create a computer program that automatically determines the age and sex of the person in the image. The program is based on neural network models for determining age and sex. It was created in the *Python* programming language using *OpenCV*, an open-code computer vision programming library. The program is also experimentally evaluated.

Key words: *Python*, *OpenCV*, neural networks

## 1. UVOD

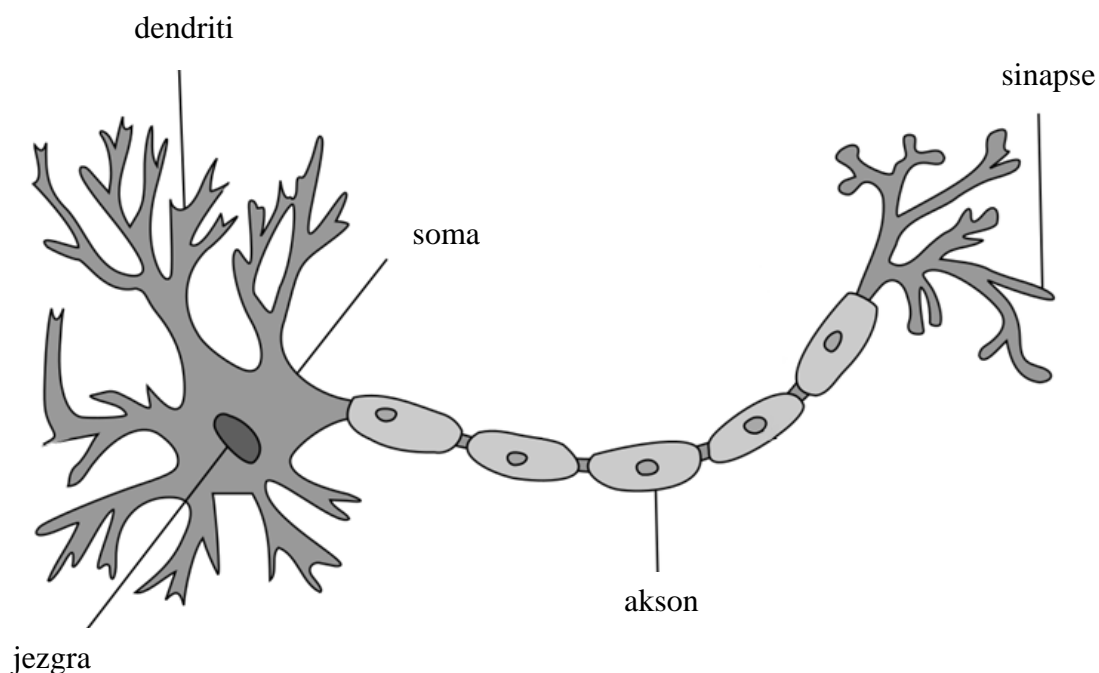
Umjetna inteligencija dio je računalnih znanosti. Označava računalne alate koji su sposobni učiti, donositi zaključke, razumjeti i analizirati govor, emocije, slike, ... [1] Umjetna se inteligencija uvelike koristi u raznim područjima. U svakodnevnom životu pojavljuje se primjerice u personaliziranom oglašavanju, pametnim uređajima i automobilima. Osim toga, koristi se u automatizaciji, poboljšavanju i ubrzavanju procesa u proizvodnji, zdravstvu, financijskom sektoru, prometu, sigurnosti te brojnim drugim aspektima društva.

Strojno učenje je grana umjetne inteligencije, a posebna grana strojnog učenja jest duboko učenje koje se primjenjuje za računalni vid, primjerice za prepoznavanje i analizu podataka sa slika i videa. Umjetne neuronske mreže su jedna od metoda strojnog učenja koja često se koristi za probleme računalnog vida. [2] U ovome radu korištena je neuronska mreža za određivanje godina i starosti osobe na učitanoj slici. Korišten je programski jezik *Python* te biblioteka otvorenog koda *OpenCV*.

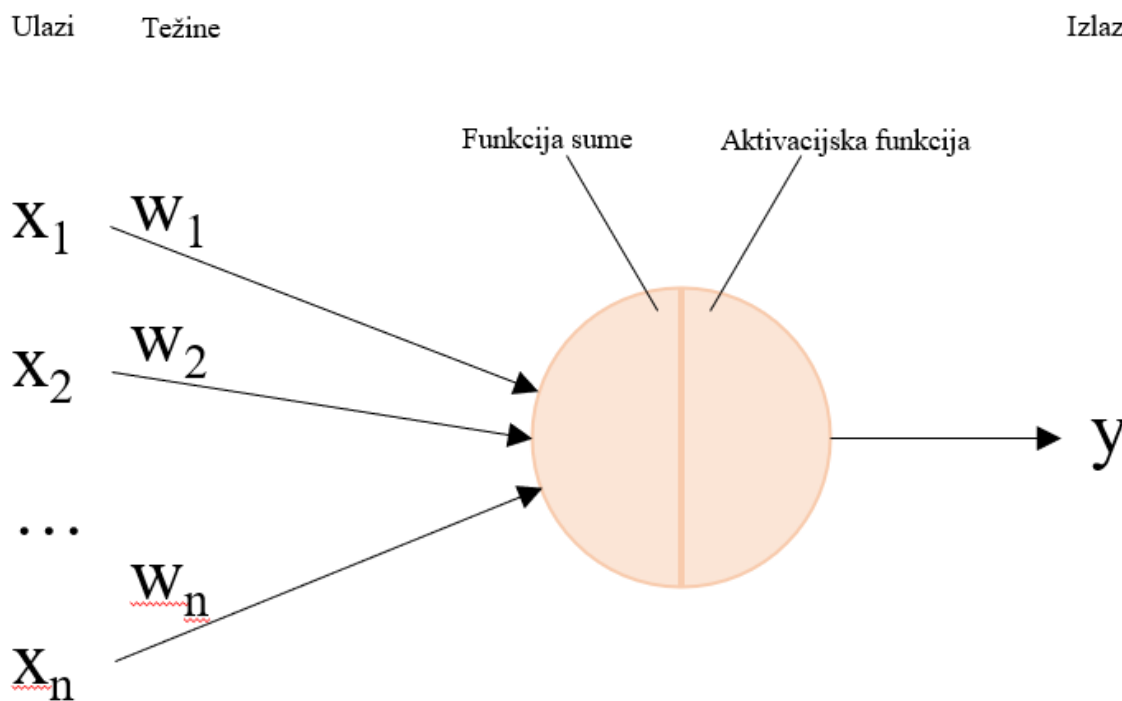
## 2. NEURONSKE MREŽE

Umjetne neuronske mreže inspirirane su biološkim neuronskim mrežama. 1890. godine William James u svojoj je knjizi napisao da aktivnost bilo koje točke mozga čovjeka predstavlja zbroj tendencija svih ostalih točaka da se prazne ili ispaljuju u nju. Time je postavio temelje za strukturu umjetnih neurona. Matematički model neuronske mreže prvi su dali McCulloch i Pitts te su koristili jednostavan model koji je nazvan perceptron. [3]

Struktura umjetnog neurona može se usporediti s biološkim neuronom. [Slika 1, Slika 2]



Slika 1. Biološki neuron [4]



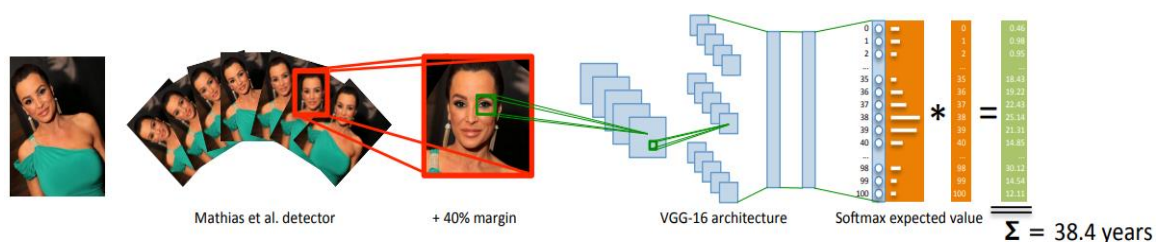
Slika 2. Umjetni neuron (perceptron)

Ulazni sloj neuronske mreže može se usporediti s dendritima, a čine ga ulazi ( $x_i$ ) otežani težinskim koeficijentima ( $w_i$ ), odnosno težinama (eng. *weights*). U „tijelu“ umjetnog neurona, koje se može usporediti sa somom biološkog neurona, nalaze se funkcija sume te aktivacijska funkcija. Funkcija sume sumira otežane ulaze prema odabranoj funkciji, dok aktivacijska funkcija određuje kako će se ulaz pretvoriti u izlaz. Izlaz ( $y$ ) može se usporediti s aksonom te se nastavlja prenositi do drugog neurona, kako u biološkoj, tako i u umjetnoj neuronskoj mreži. [3]

Za prepoznavanje obilježja na slikama prikladno je koristiti konvolucijske neuronske mreže (eng. *convolutional neural networks*) jer su one pogodne za veliku količinu kompleksnih podataka. Napravljene su po uzoru na vidni korteks te se smatraju najefikasnijima u području procesiranja slika. Kao i svaka neuronska mreža, sastoje se od ulaznog, jednog ili više skrivenih te izlaznog sloja. Za ove neuronske mreže, poseban je konvolucijski sloj. On sadrži parametre koji se uče tijekom treniranja mreže – težine i pristranosti (eng. *bias*). Takve su neuronske mreže sposobne razlikovati objekte i značajke na slici. [5]

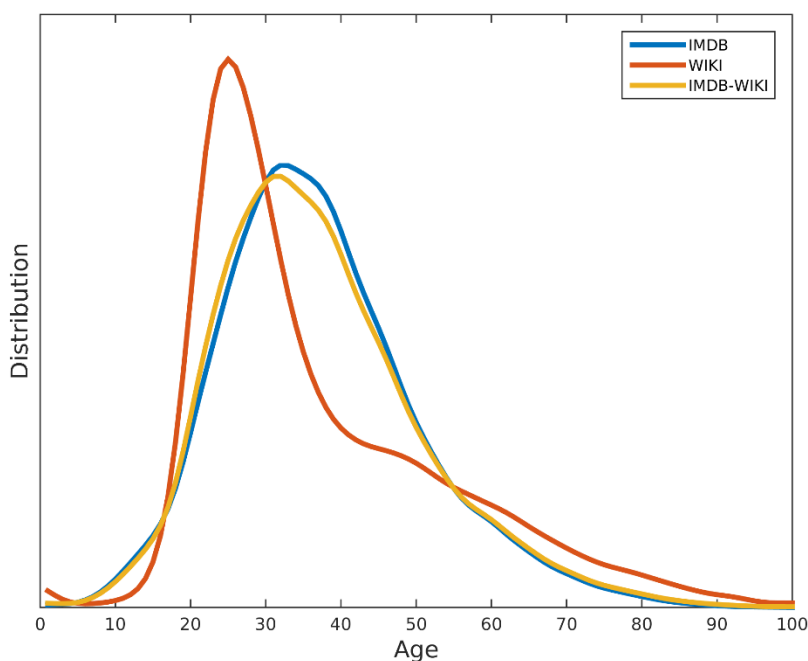
Korištena neuronska mreža prvo je bila pred-trenirana na *ImageNet* bazi podataka (vizualnoj bazi podataka dizajniranoj za korištenje u istraživanju softvera za prepoznavanje vizualnih objekata [6]), a potom je trenirana na velikom setu podataka preuzetih s *Wikipedije*

(enciklopedije slobodnog sadržaja na internetu [7]) te *IMDb-a* (*Internet Movie Database*) – najveće baze podataka o filmu na internetu koja sadrži podatke o glumcima, redateljima, scenaristima, producentima, voditeljima, ... [8] Uzete su samo one slike za koje su pronađeni potpuni podaci iz kojih se mogla odrediti starost osobe na slici (godina rođenja osobe te godina kada je slika uslikana). Sveukupno je pronađeno i korišteno 260 282 odgovarajuće slike. Algoritam na slikama prvo prepoznaje lice, zatim se slika izreže kako bi na njoj bilo prikazano samo prepoznato lice s kojega se onda izdvajaju značajke na temelju kojih se provodi određivanje godina i spola. [Slika 3]. Opisana metoda nazvana je *DEX* (eng. *Deep EXpectation Method*). [9]



**Slika 3. Model neuronske mreže [9]**

Na slici 4 prikazana je distribucija godina osoba na korištenim slikama, što može utjecati na točnost modela za određivanje željenih značajki. Ukoliko je za starost osobe na učitanju slike korištena manja količina podataka pri treniranju mreže, primjerice za osobe mlađe od 10 i starije od 90 godina, moguće je da će točnost određivanja značajki biti manja.



**Slika 4. Raspodjela godina osoba na slikama korištenim u neuronskoj mreži [10]**

Umjesto postavljanja problema određivanja godina kao regresijskog problema, korištena je klasifikacija sa 101 kategorijom, odnosno diskretnim vrijednostima godina u rasponu od 0 do 100. Potom su vrijednosti usavršene tzv. *softmax* očekivanom vrijednošću. Kao rezultat dobivaju se vjerojatnosti godina osobe na slici.

Za evaluaciju mreže, korištena je srednja apsolutna pogreška (kao prosjek apsolutnih grešaka između dobivene i stvarne godine starosti) te iznosi 3.221. Kao druga vrsta provjere, koristila se i  $\varepsilon$ -greška koja osim stvarne godine starosti uzima u obzir i procjenu nekoliko korisnika, prema formuli:

$$\varepsilon = 1 - e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

gdje su

$x$  – procijenjena godina starosti

$\mu$  – aritmetička sredina procjene korisnika

$\sigma$  – standardna devijacija procjene korisnika.

Pri evaluaciji mreže,  $\varepsilon$ -greška iznosila je 0.278. Rezultati su, kao i trenirana neuronska mreža, preuzeti iz navedenog izvora. [9]

Arhitektura korištene mreže jest VGG-16. Model sadrži 16 slojeva te postiže točnost od 92.7% na *ImageNet* bazi podataka. [11]

### 3. PYTHON I OPENCV

#### 3.1. Python

*Python* je jedan od danas najpopularnijih programskih jezika. Ima opću namjenu, jednostavnu sintaksu te je interpretirani jezik koji prevodi kod liniju po liniju. Lako je prenosiv, što znači da se može koristiti na različitim operacijskim sustavima.

Osmislio ga je Guido van Rossum te je postao dostupnim 1991. godine. Upotreba i razvoj *Pythona* brzo su se proširili po cijelom svijetu te doprinijeli njegovoj današnjoj popularnosti. Drugo je izdanje objavljeno u listopadu 2000. godine te se mu pripadaju verzije 2.x. Treće je izdanje objavljeno u prosincu 2008. godine i trenutno je posljednje izdanje s najnovijom verzijom *Python 3.11.2*, objavljenom u veljači 2023. godine. U radu je korištena verzija 3.9.13. *Python* se može koristiti za analizu podataka, strojno učenje, razvoj web aplikacija, vizualizaciju podataka, obradu slika, ... [13]



**Slika 5. Logo programskog jezika *Python* [13]**

Pogodan je za više vrsta programiranja – objektno usmjereno, strukturalno te aspektno usmjereno programiranje. [14]

U radu je korišteno objektno usmjereno programiranje (eng. *object-oriented programming*).

Biblioteka (eng. *library*) je zbirka dijelova koda ili (pot)programa koji pripadaju istoj temi. Koriste se kao pomoćni dijelovi u programu. Postoji mnogo biblioteka otvorenog koda (eng. *open source*) koje se u *Pythonu* mogu koristiti. Neke od njih su *NumPy*, *SciPy*, *PyTorch*, *Keras*, *OpenCV*, ... [13]



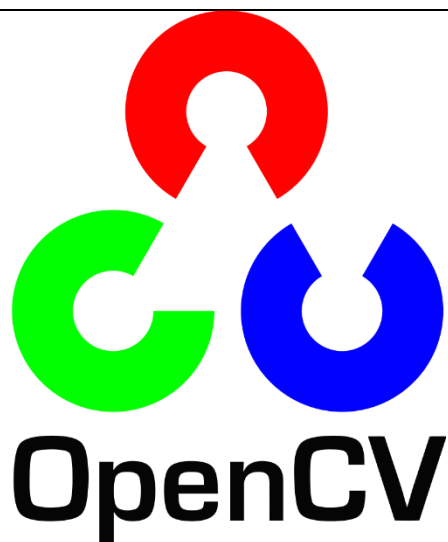


Slika 6. Python biblioteke [15]

U radu je korištena *OpenCV* biblioteka koja je opisana u sljedećem potpoglavlju.

### 3.2. *OpenCV*

*OpenCV* (eng. *Open Source Computer Vision*) je programska biblioteka otvorenog koda za računalni vid. Fokus joj je na upotrebi računalnog vida u stvarnom vremenu. Može se koristiti u više programskih jezika (*C++*, *Python*, *Java*, *MATLAB*) te podržava operacijske sustave *Windows*, *Linux*, *Android* i *Mac OS*. [15]



Slika 7. *OpenCV* logo [17]

*OpenCV* sadrži nekoliko stotina algoritama za računalni vid. Neki od njih su algoritam za analizu videa, trodimenzionalnu rekonstrukciju, prepoznavanje objekata i lica, izdvajanje značajki, ... U radu se navedena biblioteka koristi u svrhu korištenja neuronskih mreža, prepoznavanja lica i izdvajanja značajki te je korištena verzija 4.7.0.

Da bi se *OpenCV* mogao koristiti u *Pythonu*, potrebno je preuzeti i instalirati biblioteku. Prije instalacije, potrebno je ažurirati *pip*. *Pip* je sustav napisan u *Pythonu* koji se koristi za instaliranje i upravljanje softverskim paketima. [18]

Naredba za ažuriranje jest `pip install --upgrade pip`, a potom se koristi naredba za instalaciju `pip install opencv-python`. Nakon toga se biblioteka *OpenCV* može uvesti u *Python* naredbom `import cv2`.

## 4. SOFTVERSKO RJEŠENJE

Kao što je već spomenuto, na početku je potrebno uvesti *OpenCV* biblioteku naredbom *import cv2* kako bi se mogle koristiti potrebne funkcije, odnosno metode.

### 4.1. Neuronska mreža

Naredbom *dnn.readNetFromCaffe()* učitavaju se potrebni modeli neuronskih mreža. Potrebno je učitati dvije datoteke, binarnu i tekstualnu. Učitana binarna datoteka je *Caffemodel* datoteka, napravljena kao model za strojno učenje. [19] Kao tekstualna datoteka, učitana je *Prototxt* datoteka čiji je oblik napravljen kako bi se koristio uz *Caffemodel* oblik. *Caffemodel* datoteka, koju nije moguće uređivati, nastaje iz *Prototxt* datoteke koju je moguće uređivati. [20] U korištenoj metodi, *dnn* predstavlja modul dubokih neuronskih mreža (eng. *deep neural network*).

### 4.2. Priprema slike

Nadalje, u rješenju se slika s koje je potrebno prepoznati spol te godine osobe učitava naredbom *imread()* čiji je argument lokacija slike. Sve datoteke korištene u softverskom rješenju potrebno je staviti na istu lokaciju. U predloženom softverskom rješenju, korisnik upisuje samo naziv slike koju želi učitati, pod pretpostavkom da je slika stavljena na ispravnu lokaciju.

Kako bi se učitana slika pripremila za prepoznavanje spola i godina, potrebno je pronaći lice na slici te sliku pretvoriti u odgovarajući format. Korištene su naredbe:

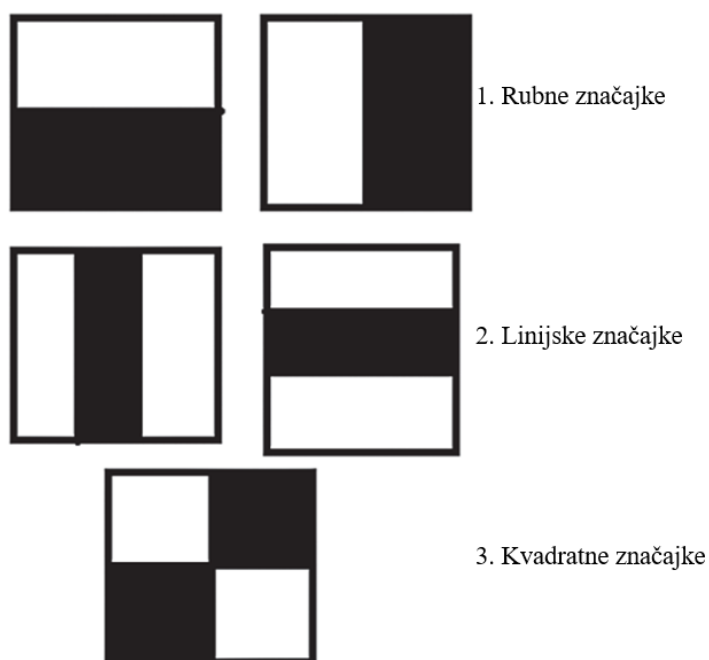
- *CascadeClassifier()* – metoda koja omogućuje učitavanje klasifikatora.

Korišten je Haarov kaskadni klasifikator, odnosno klasifikator koji radi s Haarovim značajkama. Kaskadne su funkcije općenito trenirane na tzv. pozitivnim i negativnim slikama, odnosno slikama koje sadrže objekte za koje ih se trenira prepoznati i slikama koje ne sadrže te objekte. Umjesto rada s pikselima, ovaj klasifikator radi s pravokutnim, odnosno Haarovim značajkama. [21]

Haarove su značajke:

1. Rubne
2. Linijske

## 3. Kvadratne



**Slika 8. Haarove značajke [22]**

- *detectMultiScale()* – metoda koja detektira objekte na slici. Prvi je argument učitana slika, a ostali su argumenti faktori o kojima ovisi koliko će se dobro prepoznati lice, a u konačnici i potrebne značajke s lica, te se odabir tih faktora pokazao bitnim za točnost konačnog rješenja.

Faktor skaliranja jest drugi argument. Koristi se jer je učitano sliku potrebno skalirati da bi se mogla detektirati. Korak skaliranja uvelike utječe na rezultat. Primjerice, faktor skaliranja 1.15 predstavlja korak skaliranja od 15%, odnosno slika se skalira za 15% u svakom koraku počevši od originalne veličine slike sve dok se ne dobije veličina potrebna za model. [23] Za velike iznose, detekcija će biti brža, dok je za male iznose detekcija sporija, ali točnija. Uzevši u obzir ove podatke te podatke dobivene testiranjem, u konačnici je izabran faktor iznosa 1.3 koji se pokazao najboljim za dobivanje točnih rezultata u realnom vremenu.

Treći je argument faktor koji utječe na kvalitetu detekcije. On određuje koliko susjednih pravokutnika svaki pravokutnik koji je kandidat treba imati da bi se zadržao u detektiranom području. [23] Veće vrijednosti dat će manji broj detekcija, ali će kvaliteta biti veća. Na nekoliko je primjera testirano nekoliko različitih faktora te je izabran je faktor iznosa 6 koji daje najbolje rezultate.

Kao izlaz, dobivaju se podaci o donjem lijevom kutu te širini i visini detektiranog lica.

Ti podaci koriste se kako bi se odabrao relevantan dio slike.

Odabir relevantnog dijela učitane slike koji će se koristiti za prepoznavanje navedenih značajki te transformacija slike izvodi se naredbom oblika *slika[visina, širina]*, gdje su visina i širina intervali vrijednosti unutar kojih su prethodne metode pronašle lice. Visina se unosi u obliku intervala  $[y_1, y_2]$ , odnosno u *Pythonovoj* sintaksi  $y_1:y_2$ , pri čemu  $y_1$  predstavlja y-koordinatu donjeg ruba slike te je dobivena pomoću prethodno opisane naredbe *detectMultiScale()*, a  $y_2$  je koordinata gornjeg ruba slike, koja je dobivena zbrajanjem  $y_1$  i visine također dobivene naredbom *detectMultiScale()*. Analogno se dobiva širina slike.

Zatim se koriste naredbe:

- *resize()* – kako bi se dobila veličina slike potrebna za model.

Karakteristične dimenzije za neuronske mreže su  $224 \times 224$ ,  $227 \times 227$  te  $299 \times 299$ . U ovim neuronskim mrežama, zbog VGG-16 arhitekture koristi se  $224 \times 224$ .

- *dnn.blobFromImage()* – *BLOB (Binary Large Object)* predstavlja binarni veliki objekt, odnosno zbirku binarnih podataka pohranjenih kao jednu cjelinu. [24] U ovom slučaju, predstavlja grupu povezanih piksela na slici koji imaju zajednička svojstva. Korištenim neuronskim mrežama potrebni su kao ulazi da bi se prepoznale značajke.

Sada je slika pripremljena za prepoznavanje značajki.

### 4.3. Određivanje spola

Naredbom *setInput()* učitava se modificirana slika u model neuronske mreže, dok se naredbom *forward()* dobiva informacija o željenoj vrijednosti.

Za određivanje spola, prethodne se dvije naredbe koriste za neuronsku mrežu za određivanje spola. U tom slučaju, *forward()* daje dva člana kao izlaz koji predstavljaju ženski i muški spol. Naredbom *round()* zaokružuju se vrijednosti.

Ukoliko je vrijednost prvog člana izlaza zaokružena na 1, spol je ženski, a ako je vrijednost zaokružena na 0, spol je muški. [10]

Na temelju dobivene vrijednosti, ispisuje se odgovarajuća poruka, što će biti prikazano u poglavlju 5.

#### 4.4. Određivanje godina

Za određivanje godina osobe na slici, odnosno koliko se čini da osoba na slici ima godina, metode je potrebno koristiti s odgovarajućim modelom neuronske mreže za određivanje godina.

Naredbom *setInput()* ponovno se učitava ista modificirana slika.

Naredba *forward()* daje izlaz koji se sastoji od 101 člana koji predstavljaju vjerojatnosti o godinama osobe na slici. Svaka od vjerojatnosti odgovara jednoj godini starosti iz intervala [0, 100]. Drugim riječima, naredba *forward()* daje distribuciju vjerojatnosti starosti osobe na slici. [10]

Primjer je prikazan na slici, [Slika 9].

```
[0.00025168 0.00057025 0.00053211 0.00035283 0.00051087 0.00060473
0.00063736 0.00113281 0.00132577 0.0017828 0.00201332 0.00224211
0.00152394 0.00290705 0.00379204 0.00348102 0.00281284 0.00633158
0.00699092 0.00686194 0.00861085 0.01279173 0.01902808 0.02214076
0.03365838 0.03851293 0.03638788 0.04031525 0.05091509 0.06003918
0.04476896 0.04933211 0.0702141 0.05624266 0.05560096 0.04083641
0.03374482 0.03515718 0.02645458 0.02170917 0.01970086 0.01525293
0.010204 0.01133807 0.01009409 0.00774784 0.00712454 0.00645588
0.00562183 0.00552344 0.00494607 0.00547589 0.00598202 0.00394708
0.00497741 0.00480801 0.00386516 0.00381623 0.00434713 0.00453375
0.00475811 0.00387654 0.0043774 0.00359461 0.0028028 0.00310826
0.00240439 0.0024729 0.00217438 0.00258689 0.00213443 0.00187636
0.00133324 0.00144458 0.00136857 0.00136281 0.00138459 0.0009326
0.00068275 0.00092094 0.00063079 0.00039378 0.00036807 0.00034056
0.00064489 0.00030301 0.00022684 0.0002813 0.00024942 0.00018855
0.00023211 0.00018283 0.00018506 0.00019818 0.00015799 0.00014644
0.00020528 0.00013126 0.00014823 0.0001466 0.0001604 ]
```

Slika 9. Vjerojatnosti starosti osobe na slici

Korištenjem dobivenog izlaza te jednostavne *for* petlje, u kojoj se vjerojatnost da osoba na slici ima određen broj godina množi s tom istom godinom te se dobivene vrijednosti zbrajaju, dolazi se do brojke o procijenjenoj starosti osobe.

Potom se ispisuje odgovarajuća poruka, što će također biti prikazano u poglavlju 5. Za primjer prikazan na slici [Slika 9], dobiva se da osoba na slici izgleda kao da ima 24 godine.

---

#### **4.5. Provjera dobivenih rezultata**

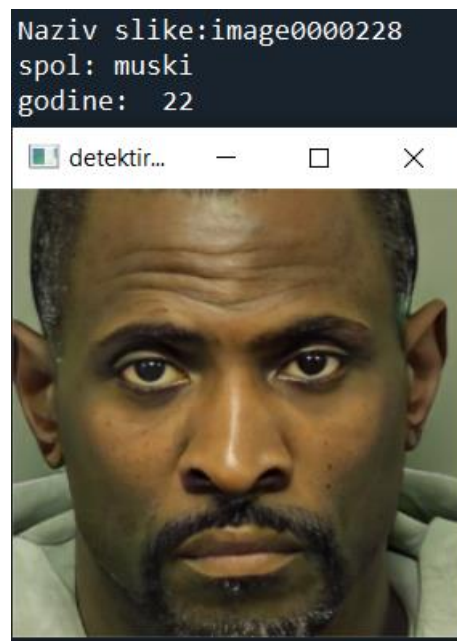
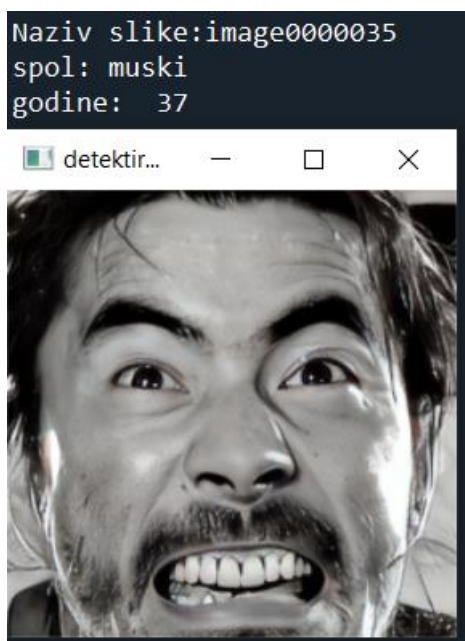
Na kraju se koristi naredba *imshow()* kojoj je kao argument stavljena obrađena slika sa koje su prepoznate značajke na temelju kojih su se odredili spol i godine. Korištenjem ove naredbe omogućuje se prikaz obrađene slike te vizualna provjera detekcije dijela slike koji se želio analizirati u svrhu određivanja značajki, što bi trebalo biti lice osobe za koju se žele odrediti spol i godine. Primjeri će biti prikazani u petom poglavlju.

## 5. EKSPERIMENTALNA EVALUACIJA

U sklopu Laboratorija za projektiranje izradbenih i montažnih sustava, za eksperimentalnu evaluaciju dobivenog softverskog rješenja odabran je *AffectNet* set – veliki set slika različitih izraza lica.

U nastavku su prikazani neki od rezultata za svaku od emocija.

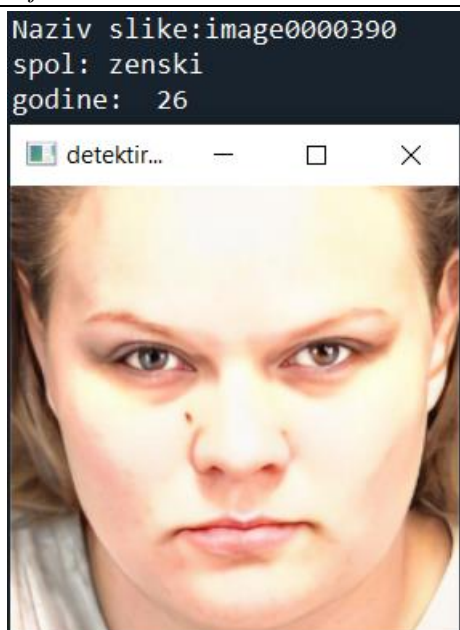
### 5.1. Ljutnja



Slika 10. Prvi primjer ljutnja

Slika 11. Drugi primjer ljutnja

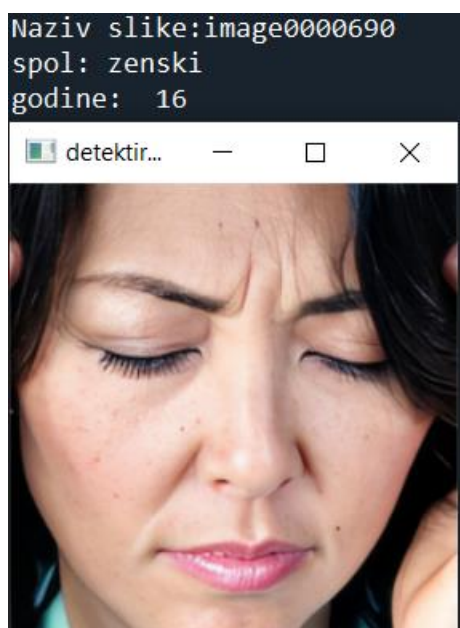




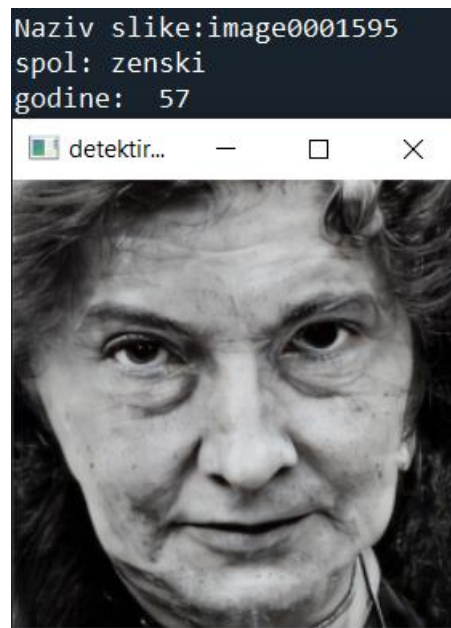
Slika 12. Treći primjer ljutnja



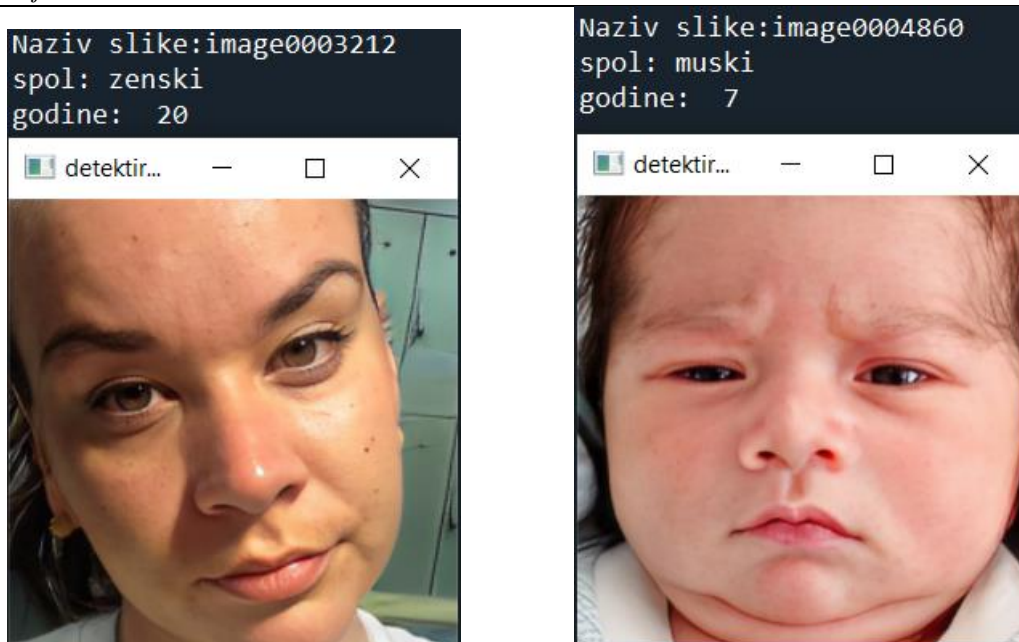
Slika 13. Četvrti primjer ljutnja



Slika 14. Peti primjer ljutnja



Slika 15. Šesti primjer ljutnja

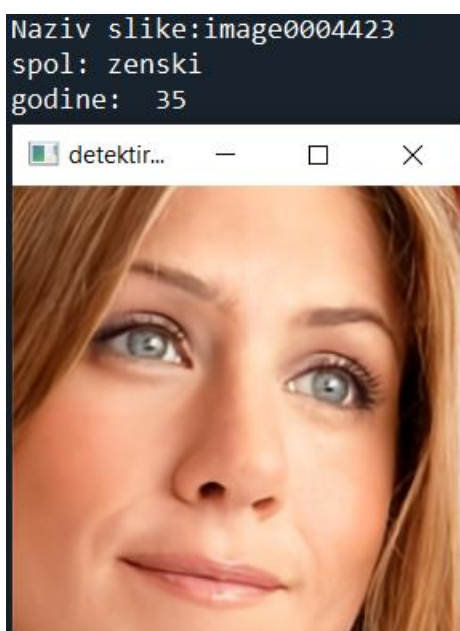


**Slika 16. Sedmi primjer ljutnja**

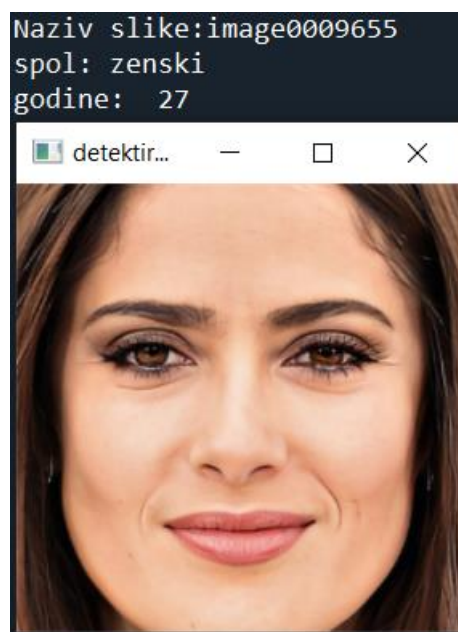
**Slika 17. Osmi primjer ljutnja**

Na osmom primjeru [Slika 17] može se jasno primijetiti spomenuti nedostatak u prethodnom poglavlju – zbog manje količine podataka za rubne intervale (manje od 10 i više od 90 godina), očekivane su veće greške u rezultatu. Na navedenom primjeru očita je veća greška između procijenjenih i stvarnih godina.

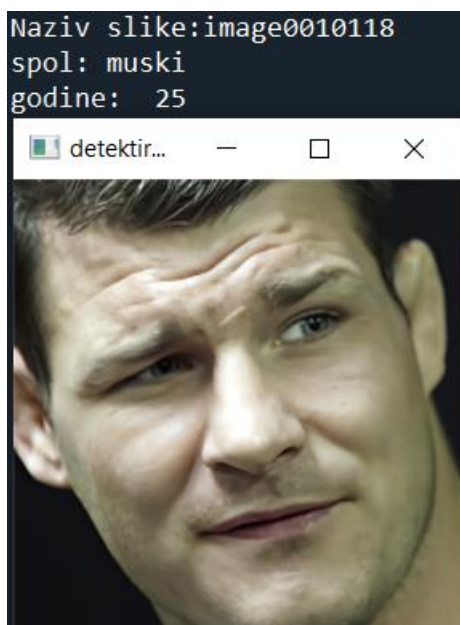
## 5.2. Prijezir



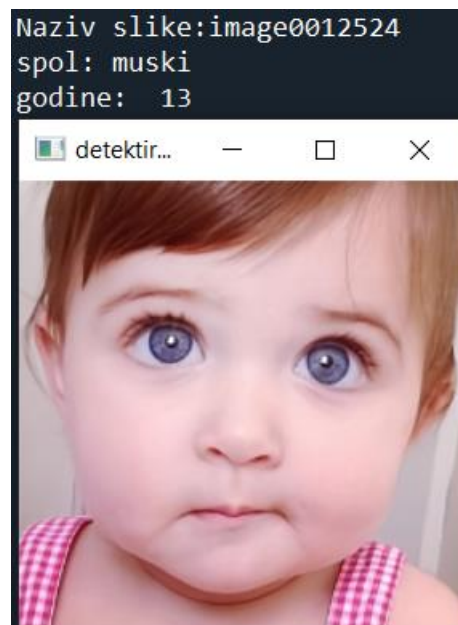
Slika 18. Prvi primjer prijezir



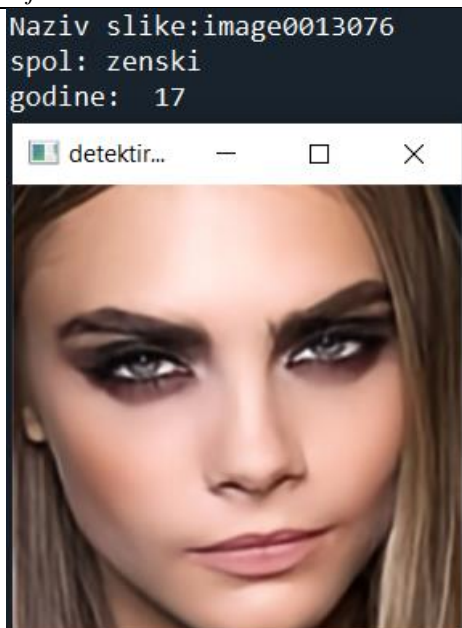
Slika 19. Drugi primjer prijezir



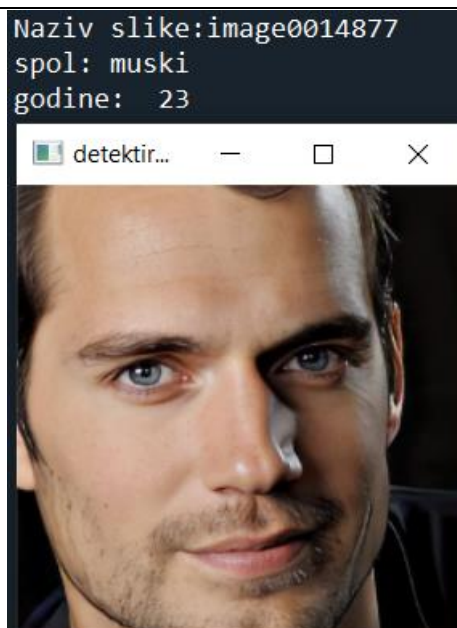
Slika 20. Treći primjer prijezir



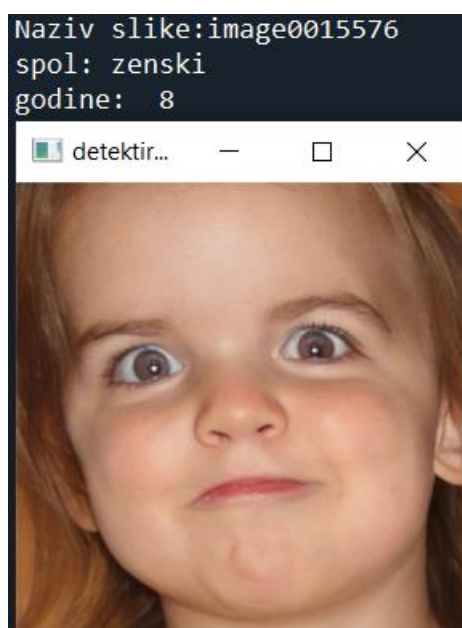
Slika 21. Četvrti primjer prijezir



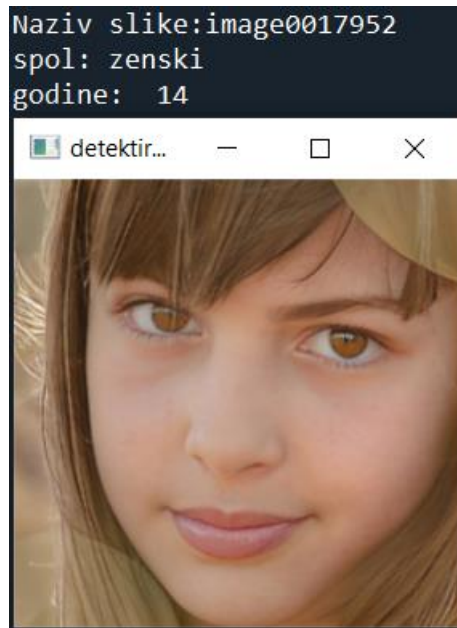
Slika 22. Peti primjer prijezir



Slika 23. Šesti primjer prijezir



Slika 24. Sedmi primjer prijezir

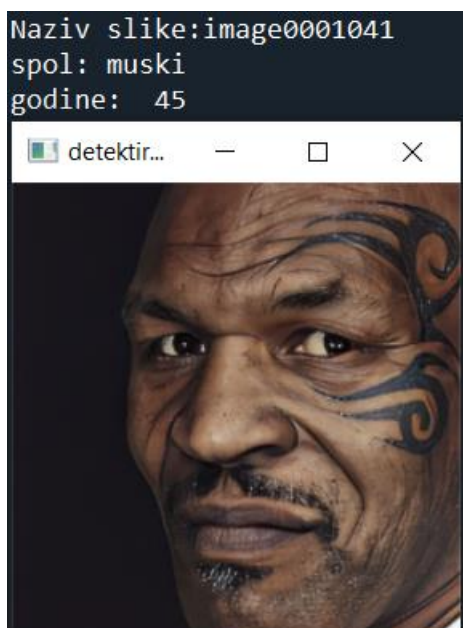


Slika 25. Osmi primjer prijezir

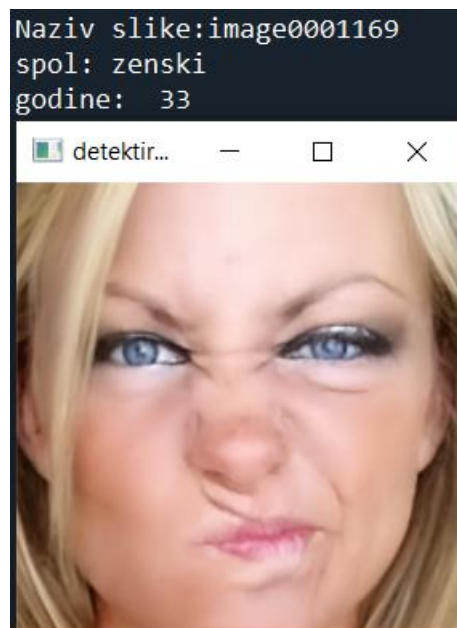
Na četvrtom primjeru [Slika 21] moguće je ponovno vidjeti već spomenuti nedostatak velike greške procijenjenih godina, ali u ovom slučaju i krivo procijenjenog spola.



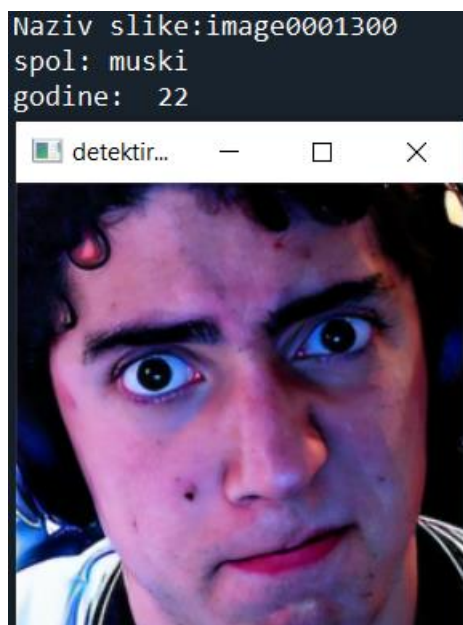
### 5.3. Gađenje



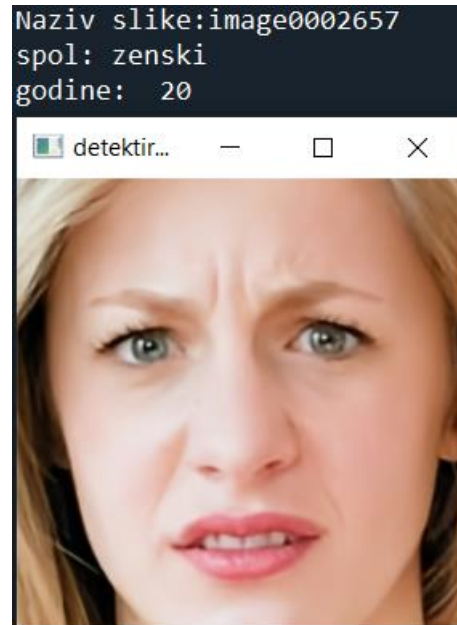
Slika 26. Prvi primjer gađenje



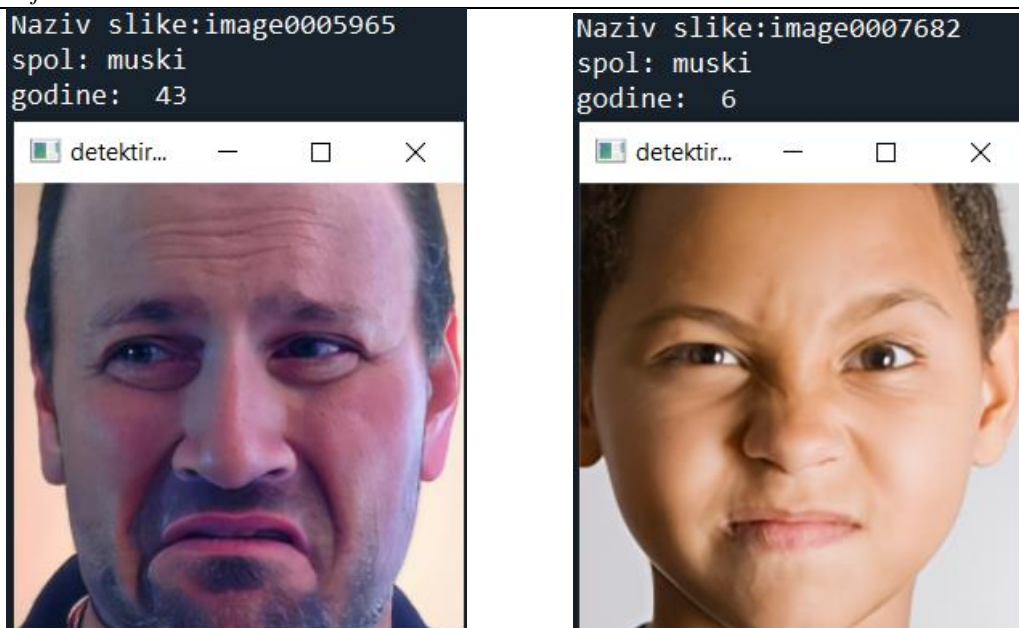
Slika 27. Drugi primjer gađenje



Slika 28. Treći primjer gađenje

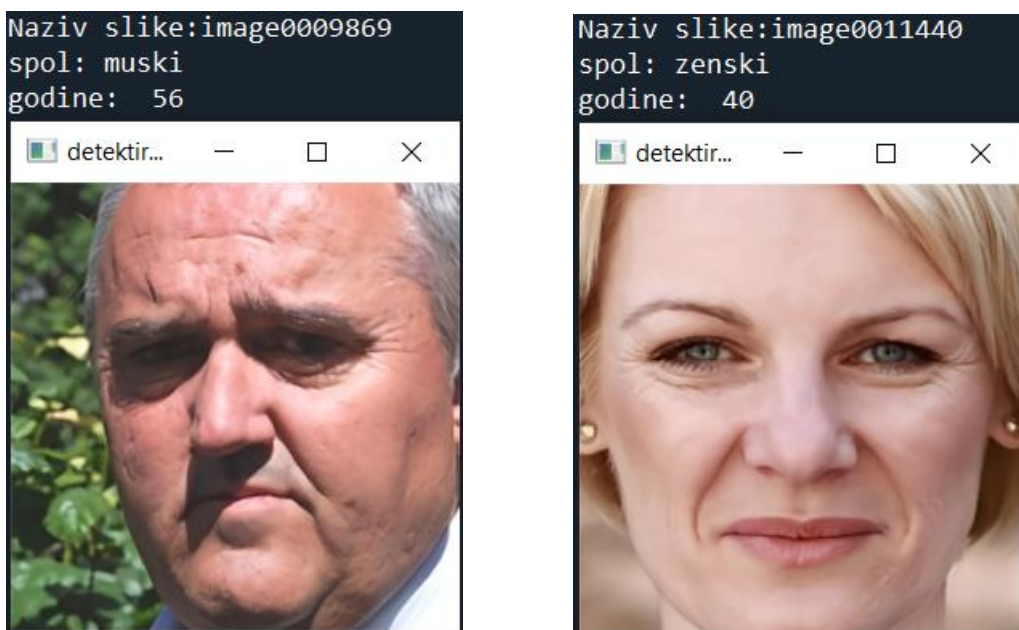


Slika 29. Četvrti primjer gađenje



Slika 30. Peti primjer gađenje

Slika 31. Šesti primjer gađenje



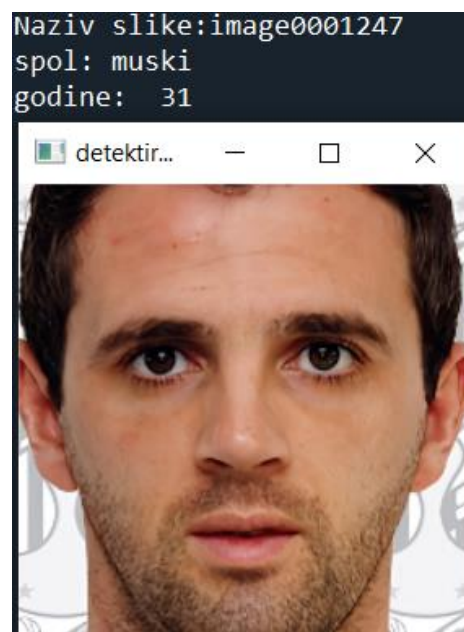
Slika 32. Sedmi primjer gađenje

Slika 33. Osmi primjer gađenje

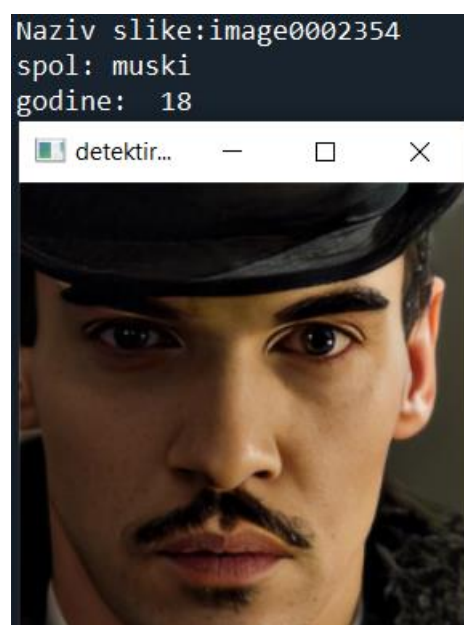
## 5.4. Strah



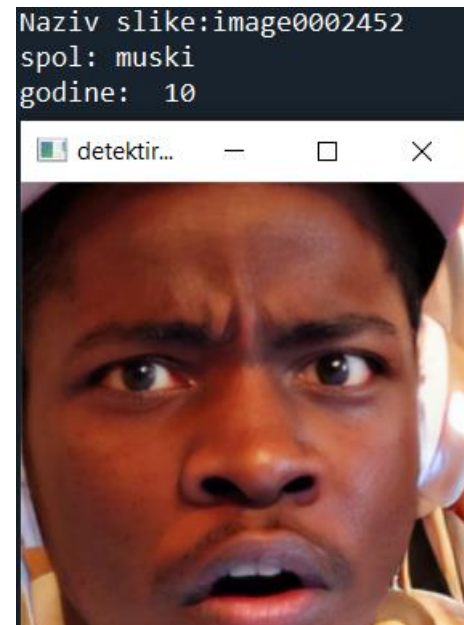
Slika 34. Prvi primjer strah



Slika 35. Drugi primjer strah



Slika 36. Treći primjer strah

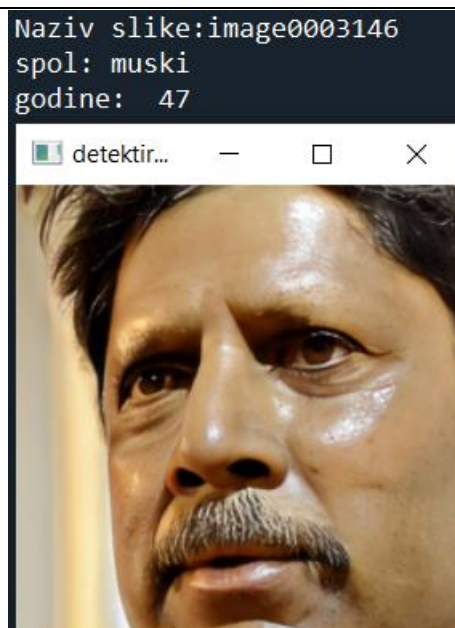


Slika 37. Četvrti primjer strah

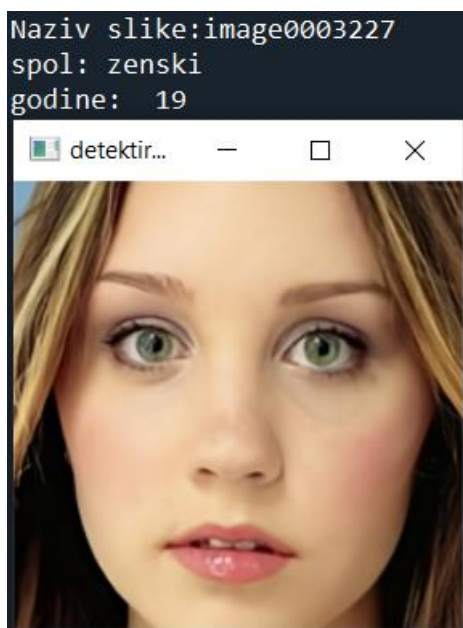




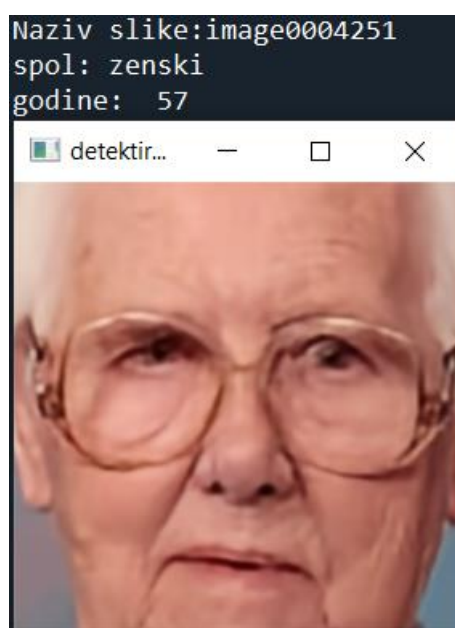
Slika 38. Peti primjer strah



Slika 39. Šesti primjer strah



Slika 40. Sedmi primjer strah



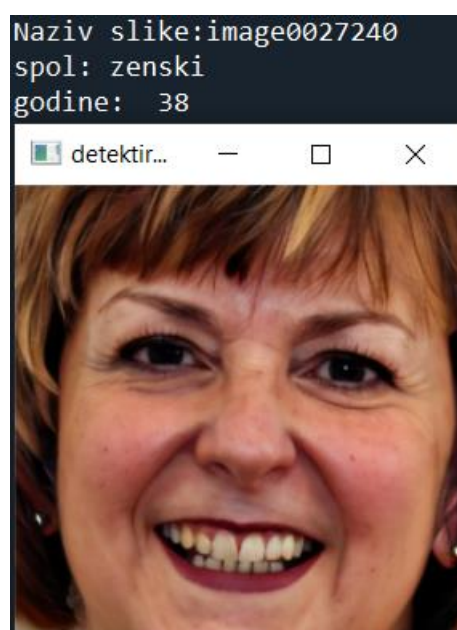
Slika 41. Osmi primjer strah



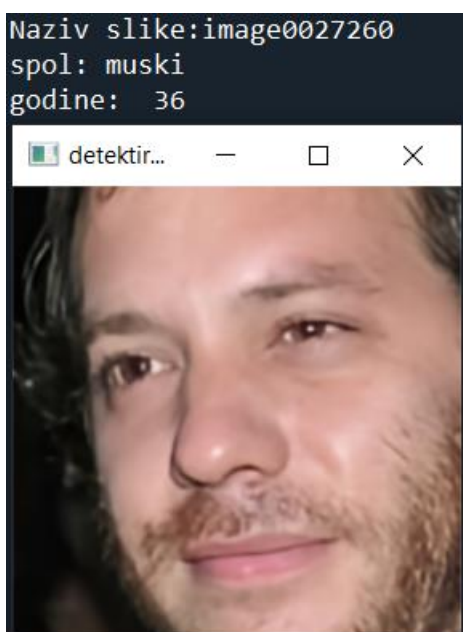
## 5.5. Sreća



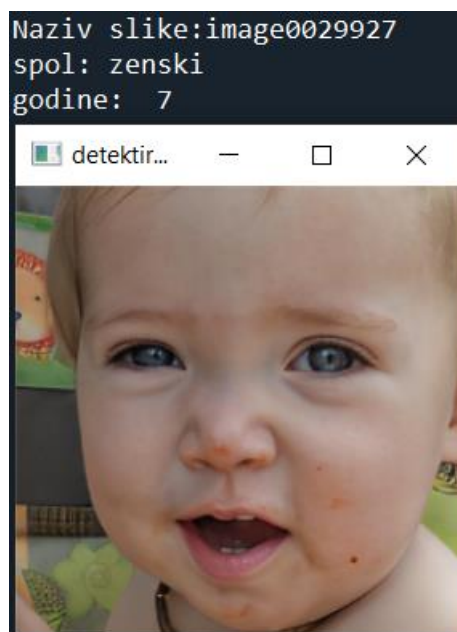
Slika 42. Prvi primjer sreća



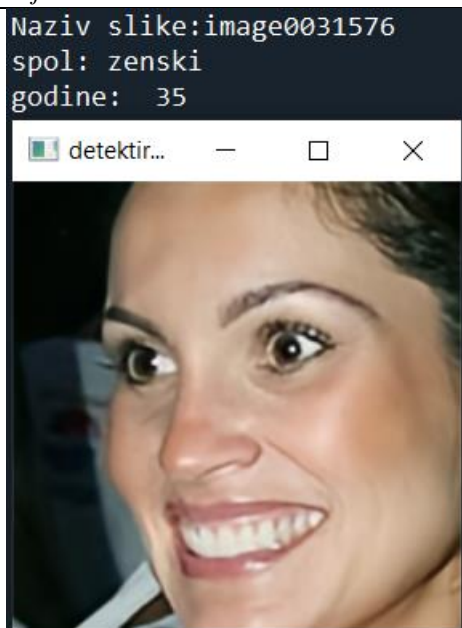
Slika 43. Drugi primjer sreća



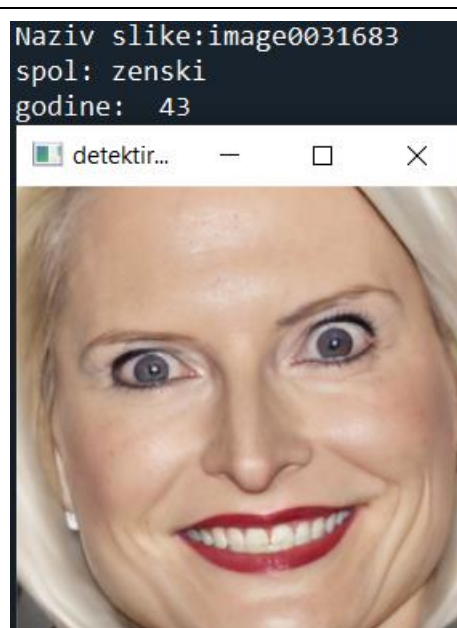
Slika 44. Treći primjer sreća



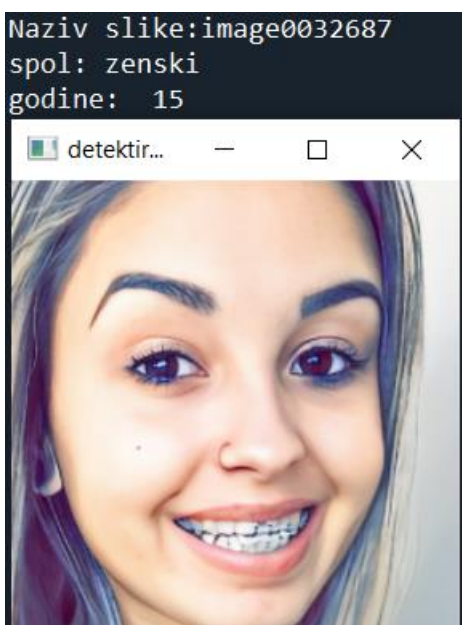
Slika 45. Četvrti primjer sreća



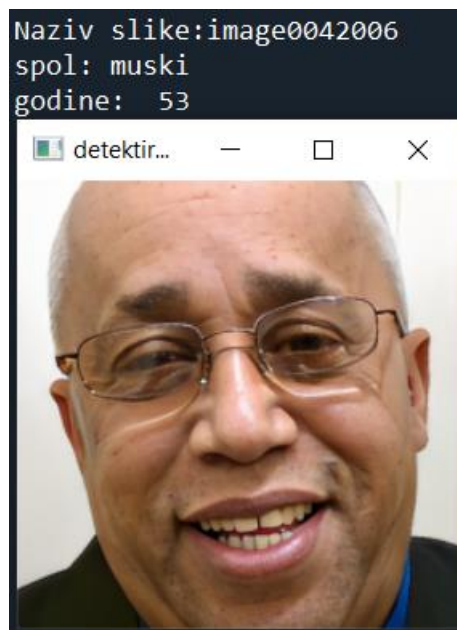
**Slika 46. Peti primjer sreća**



**Slika 47. Šesti primjer sreća**



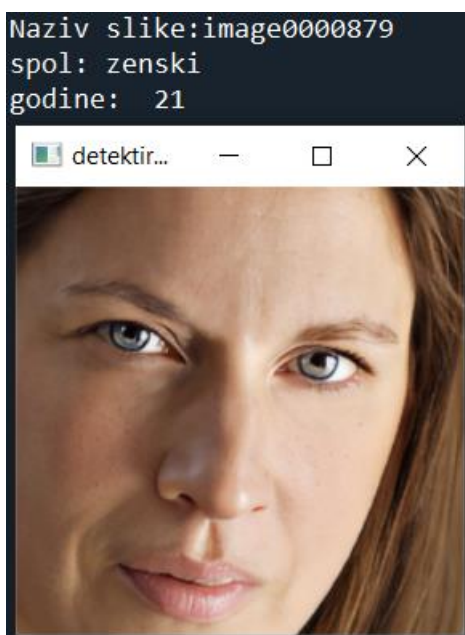
**Slika 48. Sedmi primjer sreća**



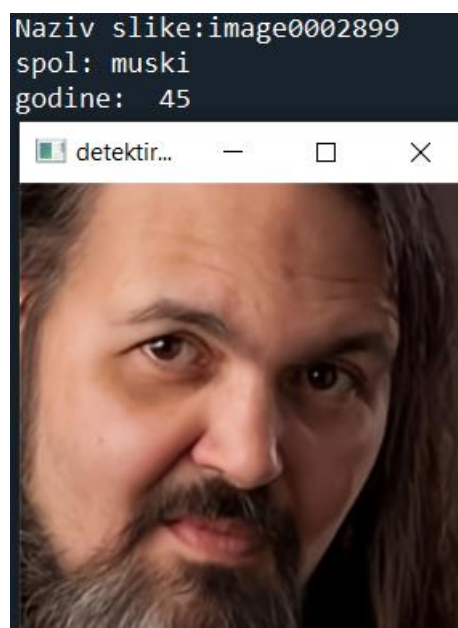
**Slika 49. Osmi primjer sreća**

Na četvrtom se primjeru [Slika 45] ponovno može vidjeti već spomenuti nedostatak veće greške procijenjenih godina.

## 5.6. Neutralno



Slika 50. Prvi primjer neutralno



Slika 51. Drugi primjer neutralno

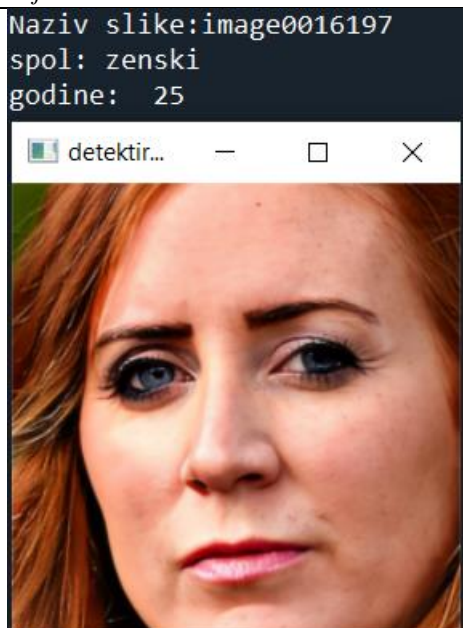


Slika 52. Treći primjer neutralno



Slika 53. Četvrti primjer neutralno

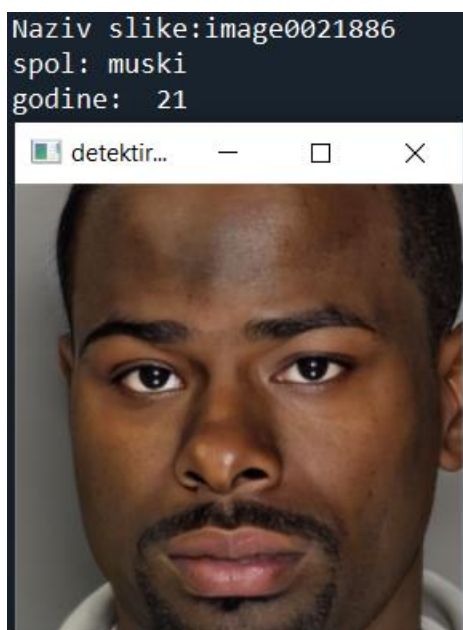




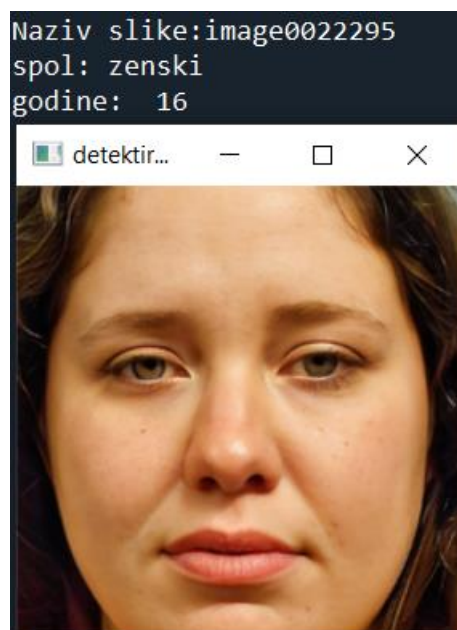
Slika 54. Peti primjer neutralno



Slika 55. Šesti primjer neutralno



Slika 56. Sedmi primjer neutralno

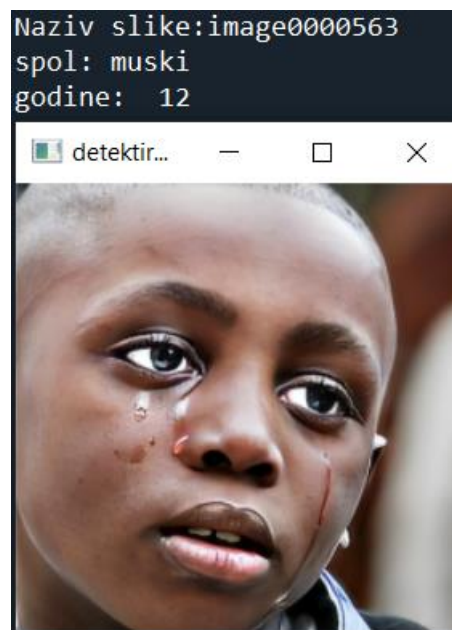


Slika 57. Osmi primjer neutralno

## 5.7. Tuga



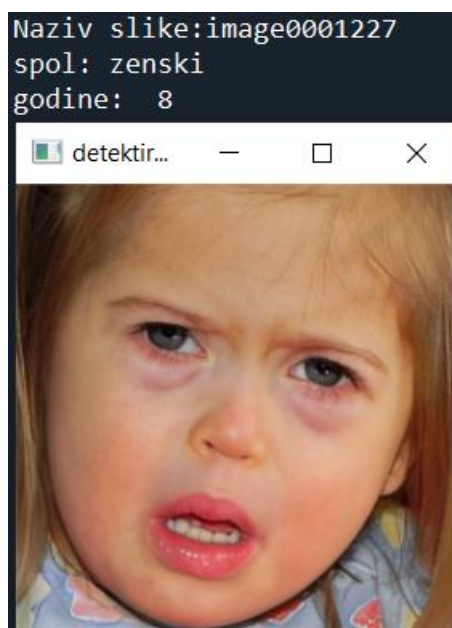
Slika 58. Prvi primjer tuga



Slika 59. Drugi primjer tuga



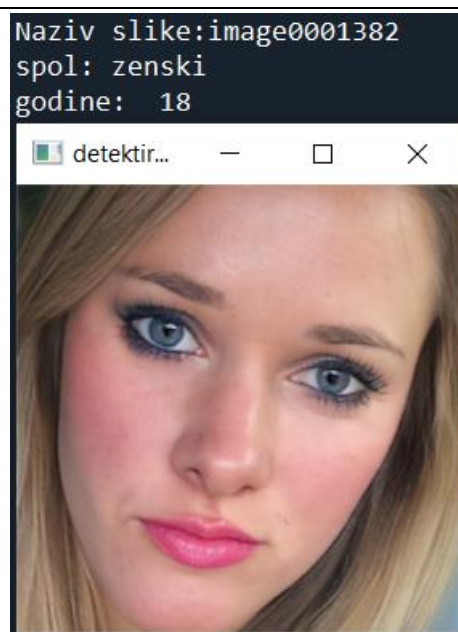
Slika 60. Treći primjer tuga



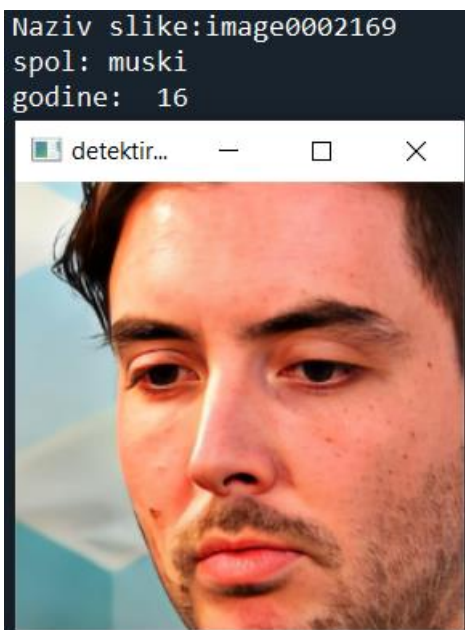
Slika 61. Četvrti primjer tuga



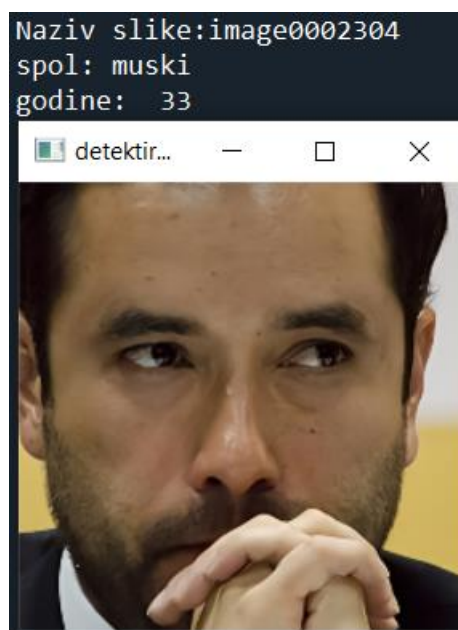
Slika 62. Peti primjer tuga



Slika 63. Šesti primjer tuga



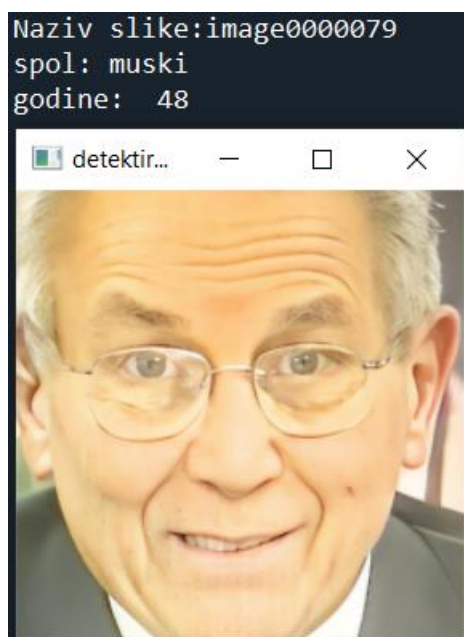
Slika 64. Sedmi primjer tuga



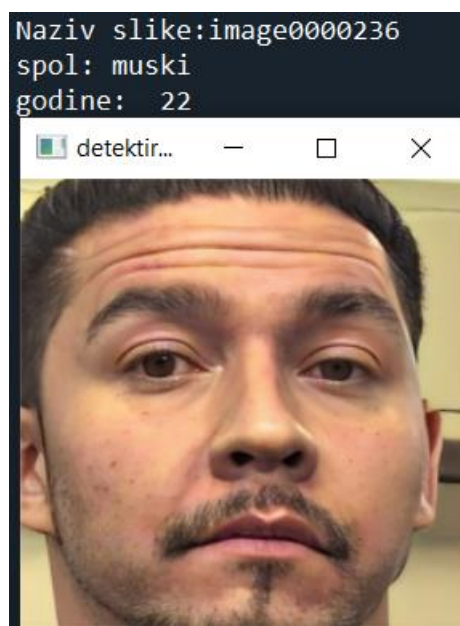
Slika 65. Osmi primjer tuga



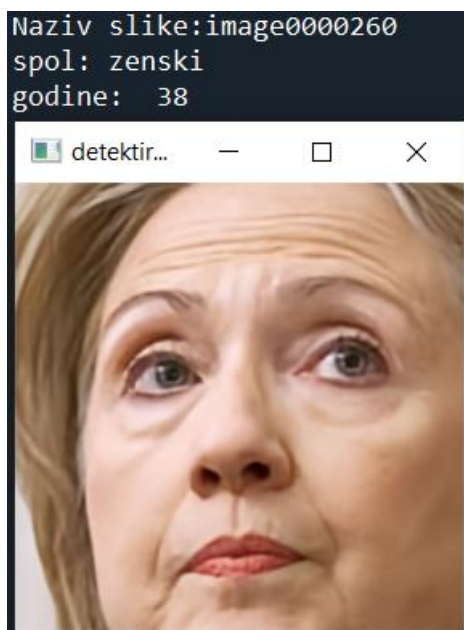
## 5.8. Iznenadenje



Slika 66. Prvi primjer iznenadenje



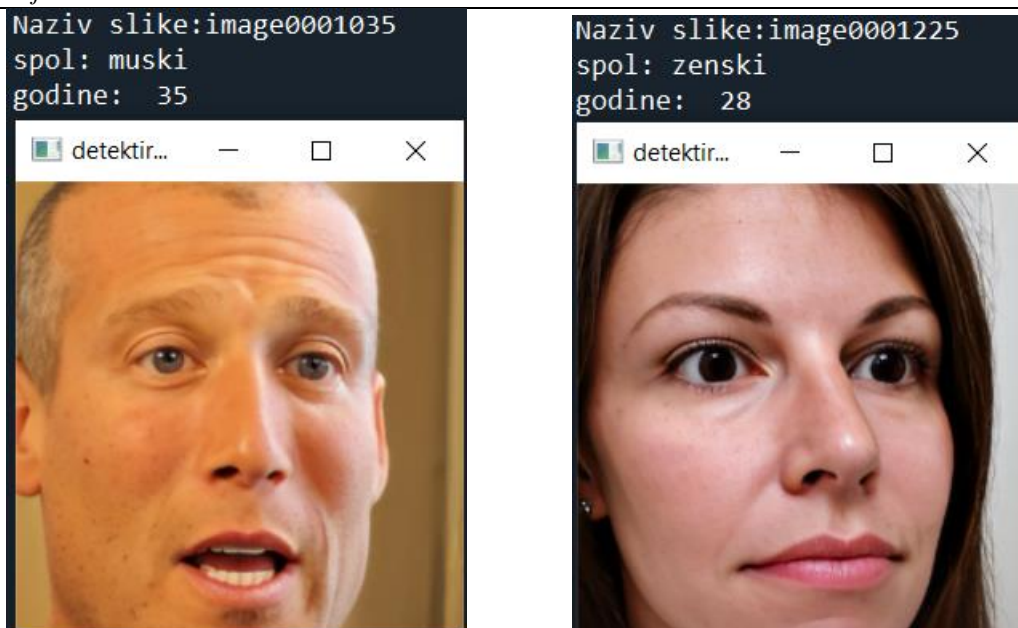
Slika 67. Drugi primjer iznenadenje



Slika 68. Treći primjer iznenadenje

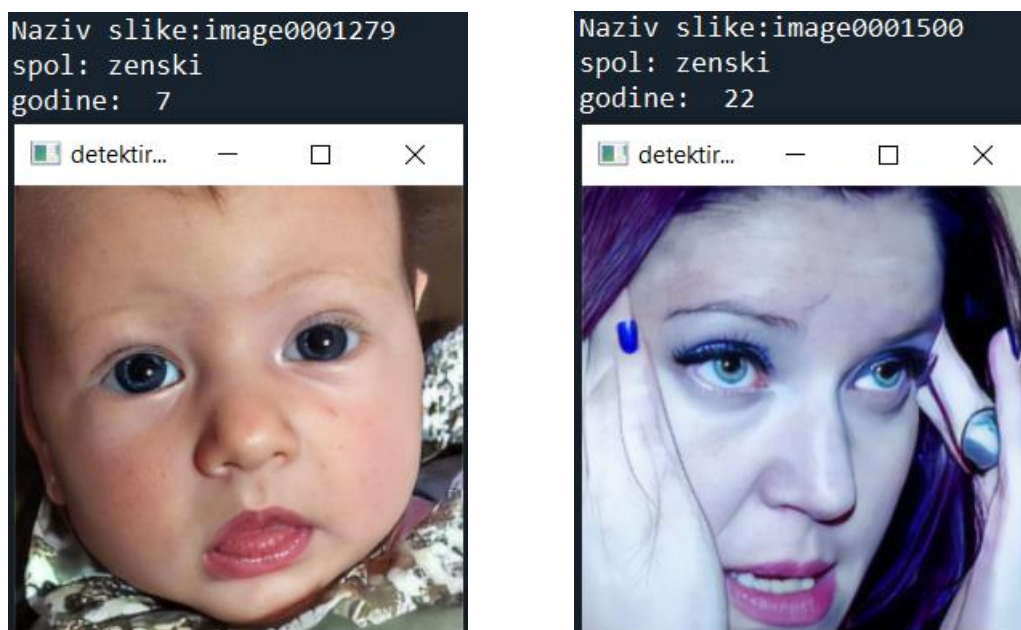


Slika 69. Četvrti primjer iznenadenje



**Slika 70. Peti primjer iznenađenje**

**Slika 71. Šesti primjer iznenađenje**



**Slika 72. Sedmi primjer iznenađenje**

**Slika 73. Osmi primjer iznenađenje**

Još jednom se može vidjeti nedostatak zbog male količine korištenih podataka za pripadajuće godine na sedmom primjeru [Slika 72], gdje je veća pogreška u procijenjenim godinama.



## 5.9. Osvrt na dobivene rezultate

Na slikama se može vidjeti da je spol osoba uglavnom dobro prepoznat. Iznimke postoje, primjerice na slici 21, što se može objasniti u radu već spomenutim nedostatkom korištene neuronske mreže – manjom količinom podataka korištenom za treniranje neuronske mreže za određene skupine godina (manje od 10 i veće od 90). Kako u predloženom setu podataka za evaluaciju nema podatka o starosti osobe na slici, nije moguće objektivno usporediti dobivena rješenja. Subjektivno se uspoređivanjem primjećuje razlika u godinama na određenim primjerima. Na slikama: Slika 11, Slika 14, Slika 36, Slika 37, Slika 50, Slika 54, Slika 64 te Slika 68 subjektivno osoba izgleda starije od dobivenog rezultata. Suprotno tome, na slikama Slika 17 i Slika 45 osobe subjektivno izgledaju mlađe od dobivenog rezultata.

U nastavku su prikazane neke od slika na kojima nije bilo moguće izdvojiti relevantno područje. Može se vidjeti da se lice nije detektiralo kada nije dovoljno prikazano na slici – ukoliko nije dovoljno okrenuto prema kameri, izlazi iz kadra ili je zaklonjeno.



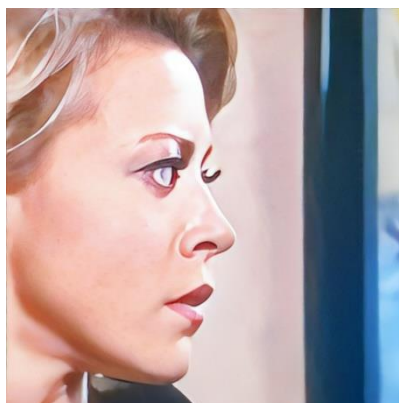
**Slika 74. Neuspješan pokušaj 1**



**Slika 75. Neuspješan pokušaj 2**



**Slika 76. Neuspješan pokušaj 3**



**Slika 77. Neuspješan pokušaj 4**

---

## **6. ZAKLJUČAK**

Implementacijom neuronskih mreža u radu je napravljen program za automatsko određivanje spola i godina osoba na slici. Prikazan je postupak dobivanja rješenja te eksperimentalna evaluacija. Eksperimentalnom evaluacijom pokazana je primjena rješenja na 8 različitih emocija: ljutnja, prijezir, gađenje, strah, sreća, neutralno, tuga i iznenađenje. Utvrđeno je da se na određenim slikama nije uspjelo prepoznati lice. Uočen je, kao što je i pretpostavljen, problem s predloženim rješenjem za slike na kojima su osobe mlađe od 10 ili starije od 90 godina zbog malog broja podataka korištenih kod treniranja korištene neuronske mreže, što daje prostora za unapređenje rješenja.

Za cjelovitu analizu rješenja bilo bi potrebno napraviti set podataka osoba koje pokazuju svih osam emocija. U tom bi se slučaju mogla napraviti analiza o procijenjenim rezultatima obzirom na različite emocije.

---

**LITERATURA**

- [1] <https://enciklopedija.hr/natuknica.aspx?ID=63150>, zadnji pristup 15.2.2022.
- [2] Stipančić, T.: predavanja iz kolegija Umjetna inteligencija, Fakultet strojarstva i brodogradnje, Zagreb, 2020.
- [3] Majetić, D.: predavanja iz kolegija Neuronske mreže, Fakultet strojarstva i brodogradnje, Zagreb, 2021.
- [4] <https://hr.wikipedia.org/wiki/Neuron>, zadnji pristup 15.2.2023.
- [5] <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53>, zadnji pristup 15.2.2023.
- [6] <https://en.wikipedia.org/wiki/ImageNet>, zadnji pristup 15.2.2023.
- [7] <https://hr.wikipedia.org/wiki/Wikipedija>, zadnji pristup 15.2.2023.
- [8] [https://hr.wikipedia.org/wiki/Internet\\_Movie\\_Database](https://hr.wikipedia.org/wiki/Internet_Movie_Database), zadnji pristup 15.2.2023.
- [9] [https://data.vision.ee.ethz.ch/cvl/publications/papers/proceedings/eth\\_biwi\\_01229.pdf](https://data.vision.ee.ethz.ch/cvl/publications/papers/proceedings/eth_biwi_01229.pdf), zadnji pristup 15.2.2023.
- [10] <https://data.vision.ee.ethz.ch/cvl/rrothe/imdb-wiki/>, zadnji pristup 15.2.2023.
- [11] <https://viso.ai/deep-learning/vgg-very-deep-convolutional-networks/>, zadnji pristup 15.2.2023.
- [12] <https://www.upgrad.com/blog/basic-cnn-architecture/>, zadnji pristup 15.2.2023.
- [13] <https://www.python.org/>, zadnji pristup 15.2.2023.
- [14] [https://hr.wikipedia.org/wiki/Python\\_\(programski\\_jezik\)](https://hr.wikipedia.org/wiki/Python_(programski_jezik)), zadnji pristup 15.2.2023.
- [15] <https://www.fireblazeaischool.in/blogs/python-libraries-for-machine-learning/>, zadnji pristup 15.2.2023.
- [16] <https://opencv.org/>, zadnji pristup 15.2.2023.
- [17] <https://en.wikipedia.org/wiki/OpenCV>, zadnji pristup 15.2.2023.
- [18] [https://en.wikipedia.org/wiki/Pip\\_\(package\\_manager\)](https://en.wikipedia.org/wiki/Pip_(package_manager)), zadnji pristup 15.2.2023.
- [19] <https://fileinfo.com/extension/caffemodel>, zadnji pristup 15.2.2023.
- [20] <https://fileinfo.com/extension/prototxt>, zadnji pristup 15.2.2023.
- [21] [https://docs.opencv.org/3.4/db/d28/tutorial\\_cascade\\_classifier.html](https://docs.opencv.org/3.4/db/d28/tutorial_cascade_classifier.html), zadnji pristup 15.2.2023.
- [22] Hrga, M.: Računalni vid, Veleučilište u Šibeniku, Šibenik, 2018.
- [23] [https://docs.opencv.org/3.4/d1/de5/classcv\\_1\\_1CascadeClassifier.html](https://docs.opencv.org/3.4/d1/de5/classcv_1_1CascadeClassifier.html), zadnji pristup 15.2.2023.
- [24] [https://en.wikipedia.org/wiki/Binary\\_large\\_object](https://en.wikipedia.org/wiki/Binary_large_object), zadnji pristup 15.2.2023.

---

**PRILOZI**

I. *Python* kod

## PYTHON KOD

```
import cv2

godine = 0

#ucitavanje treniranih modela
godine = cv2.dnn.readNetFromCaffe('C:/Završni/Datoteke/imdbwiki/age.prototxt',
'C:/Završni/Datoteke/imdbwiki/dex_imdb_wiki.caffemodel')
spol = cv2.dnn.readNetFromCaffe('C:/Završni/Datoteke/imdbwiki/gender.prototxt',
'C:/Završni/Datoteke/imdbwiki/gender.caffemodel')

#ucitavanje slike
ime = input('Naziv slike:')
lokacija = 'C:/Završni/Datoteke/imdbwiki/ljutnja/' + ime + '.jpg'
slika = cv2.imread(lokacija)

#priprema slike
detektor =
cv2.CascadeClassifier('C:/Završni/Datoteke/imdbwiki/haarcascade_frontalface_default.xml')
lice = detektor.detectMultiScale(slika, 1.3, 6)

x , y , w , h = lice[0]

x = int(x)
y = int(y)
w = int(w)
h = int(h)

detektiranoLice = slika[y:y+h , x:x+w]
detektiranoLice = cv2.resize(detektiranoLice,(224, 224))

detektiranoLiceBlob = cv2.dnn.blobFromImage(detektiranoLice)

#odredjivanje spola
spol.setInput(detektiranoLiceBlob)
spol_rez = spol.forward()

zenski = round(spol_rez[0][0])
muski = round(spol_rez[0][1])

if zenski == 1:
    print('spol: zenski')
else:
    print('spol: muski')
```

```
#odredjivanje godina
godine.setInput(detektiranoLiceBlob)
godine_rez = godine.forward()

len = len(godine_rez[0])

for i in range (0, len):
    godine = godine + (i * godine_rez[0][i])
    godine = round(godine)

print('godine: ', godine)

#prikaz detekcije
cv2.imshow('detektiranoLice',detektiranoLice)
cv2.waitKey()
```