

**SVEUČILIŠTE U ZAGREBU
PRIRODOSLOVNO-MATEMATIČKI FAKULTET
MATEMATIČKI ODSJEK**

**Prepoznavanje emocija pomoću web kamere
Umjetna inteligencija**

Izradile:
Lea Markušić i Petra Salaj

Mentor:
prof. dr. sc. Luka Grubišić

15.2.2024.

Sadržaj

1. Uvod	3
2. Detekcija lica	3
2.1 Viola-Jones algoritam za detekciju lica	4
3. Prepoznavanje emocija - DeepFace.....	6
4. Implementacija algoritma	8
5. Zaključak	10
6. Literatura	10

1. Uvod

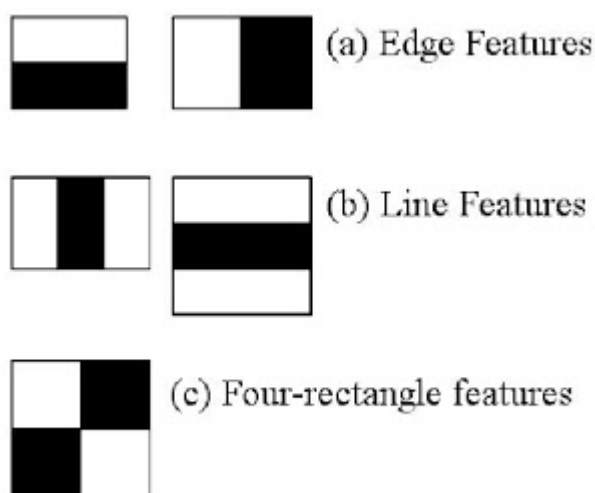
U ovome radu obradile smo temu prepoznavanja emocija pomoću web kamere, kod smo pisale u Pythonu. Kod otvara web kameru računala te se u stvarnom vremenu detektira i prepoznaje lice i izraz lica te na temelju toga određuje dominantnu emociju. Koristile smo OpenCV paket za računalni vid, strojno učenje i obradu slika te DeepFace koji koristi princip dubokog učenja u kojemu neuronske mreže uče prepoznavati uzorke u velikim bazama podataka. Tehnologija dubokog učenja radi u četiri faze: detekcija (pronalazi lice u slici), poravnanje (pomiče i rotira lice da gleda naprijed), predstavljanje (stvara numerički opis lica kao niz brojeva) i klasifikacija (uspoređuje numeričke opise različitih lica da bi utvrdio jesu li ista ili različita). Umjetna inteligencija prepoznaje sitne značajke koje čine objekt ili pojedinu jedinicu teksta, a zatim ih stavlja zajedno kako bi se stvorila cjelokupna slika stvari.

2. Detekcija lica

Detekcija lica prvi je od četiri koraka procesa prepoznavanja lica. Njegova pouzdanost je od velike važnosti za cijeli sustav prepoznavanja lica. Uloga detektora je identificiranje i lociranje svih prisutnih lica na nekoj slici ili videu bez obzira na njihov položaj, orijentaciju, dob, izraz lica, modne dodatke (npr. naočale) ili šminku. Također ne bi trebao ovisiti o vanjskim uvjetima osvjetljenja i sadržaju pozadine slike ili videa. Postoje razni algoritmi za detekciju lica koji imaju različite pristupe izvođenja. Neki se temelje na boji kože, pokretu, obliku lica ili glave, izgledu lica ili kombinaciji svega navedenog. Najuspješniji su appearance-based methods, odnosno algoritmi bazirani na izgledu bez korištenja ostalih parametara. Naime, u metodama temeljenim na izgledu detekcija lica tretira se kao problem klasificiranja skeniranog podprozora u jednu od dvije klase, lice ili ne lice. Klasifikator lica/ne lica, odnosno algoritam strojnog učenja za detekciju lica, za učenje koristi skup koji se sastoji od primjera lica u raznim uvjetima. Tijek obrade detekcije lica izvodi se na sljedeći način: najprije se ulazna slika skenira na svim mogućim mjestima i mjerilima pomoću podprozora i tada se uzorak u podprozoru detektira kao lice ili ne lice. Ova detekcija temelji se na tome što su pikseli na licu visoko korelirani, dok pikseli u podprozoru bez lica imaju mnogo manje pravilnosti. Međutim, izazov za klasifikator lica predstavljaju velike varijacije uzrokovane promjenama u osvjetljenju, izrazu, orijentaciji ili poziciji glave koje čine granice lica ili ne lica jako složenima. [1]

2.1 Viola-Jones algoritam za detekciju lica

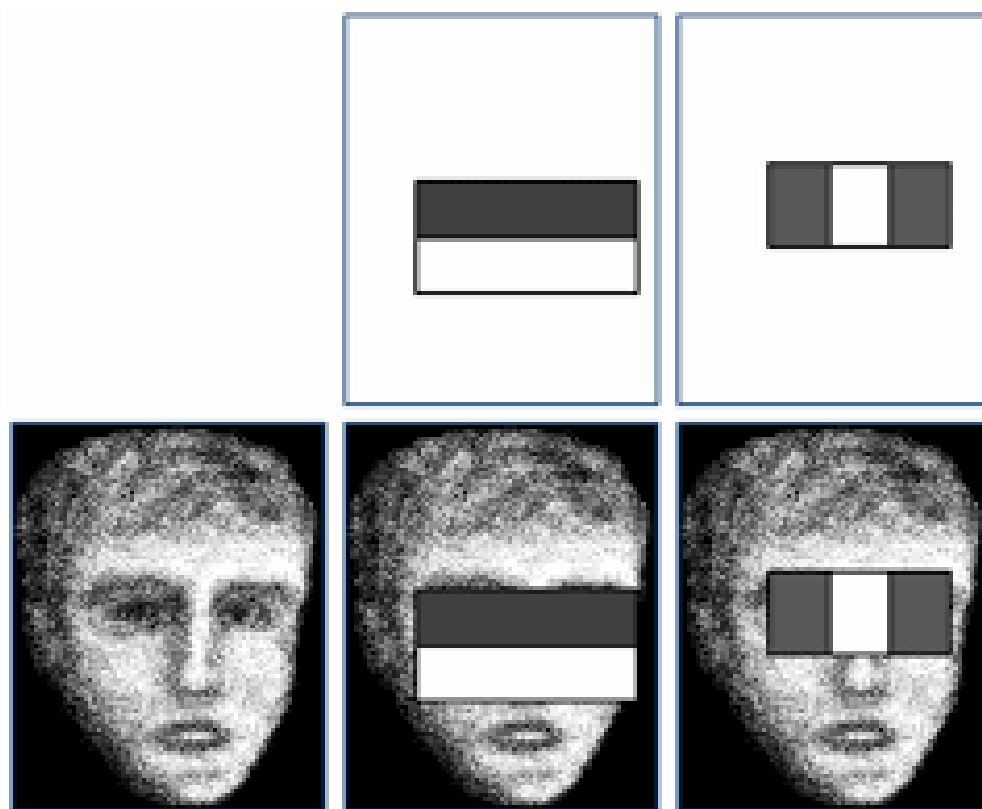
Detekcija objekata pomoću klasifikatora kaskadnih značajki Haar je učinkovita metoda koju su 2001. predložili američki znanstvenici Paul Viola i Michael Jones u svojem radu „Rapid Object Detection using a Boosted Cascade of Simple Features,, (hrv. Brza detekcija objekata pomoću pojačane kaskade jednostavnih značajki). Posebnost ovog klasifikatora je što osim detekcije lica, može detektirati i bilo koji drugi objekt koji ga se nauči (npr. životinje, automobile, prometne znakove i slično). U ovom seminaru mi ćemo se fokusirati na njegovu primarnu funkciju, detekciju lica. Ova metoda, kao i ostale bazirane na izgledu lica, traži veliku količinu pozitivnih slika (lica) i negativnih slika (bez lica) za treniranje klasifikatora. Nakon toga, potrebno je izvući značajke iz tih slika. Za to se koriste tzv. Haar-ove značajke prikazane na Slici 1 koje su poput konvolucijskih jezgri.



Slika 1: Tipovi Haarovih značajki (Izvor: <https://medium.com/analytics-vidhya/haar-cascades-explained-38210e57970d>)

Svaka značajka je pojedinačna vrijednost dobivena oduzimanjem zbroja piksela ispod bijelog pravokutnika od zbroja piksela ispod crnog pravokutnika. Sve moguće veličine i lokacije svake jezgre se koriste za izračunavanje velikog broja značajki. Treba imati na umu da za podprozor veličine 24x24 piksela dobivamo preko 160 000 značajki te da je za svaki izračun značajke potrebno pronaći zbroj piksela ispod bijelog i crnog pravokutnika. Kako bi se to riješilo, uvodi se integralna slika koja ne ovisi o veličini slike i svodi proračune za određeni piksel na operaciju koja uključuje samo četiri piksela što čini cijeli proces puno bržim. Međutim, među svim ovim izračunatim značajkama, većina ih je nevažna. Na primjer, na Slici 2 gornji red prikazuje dvije dobre značajke. Prva odabrana značajka fokusira se na svojstvo da je područje očiju često tamnije od područja nosa i obraza. Druga odabrana značajka temelji se na svojstvu da su oči

tamnije od mosta nosa. Ali, isti ti prozorčići nisu relevantni ako se primijene na obraze ili na bilo koje drugo mjesto.



Slika 2: Dvije Haar-ove značajke koje detektiraju oči i nos (Face Recognition: A Tutorial on Computational Aspects - Scientific Figure on ResearchGate. Available from: https://www.researchgate.net/figure/Two-Haar-features-that-recognize-the-eyes-and-the-nose-respectively_fig2_289252246 [accessed 16 Feb, 2024])

Dakle, algoritam bi se poboljšao kada bi od više od 160 000 značajki odabrali samo one najbolje. Upravo tome služi algoritam učenja Adaptive Boosting ili skraćeno AdaBoost u kojem se svaka pojedinačna značajka primjenjuje na sve slike za treniranje. Za svaku značajku on pronalazi najbolji prag koji će klasificirati lica na pozitivna i negativna. Očito će doći do pogrešaka ili pogrešnih klasifikacija. Odabiru se značajke s minimalnom stopom pogrešaka, odnosno značajke koje najpreciznije klasificiraju slike lica i bez lica. Međutim, proces nije tako jednostavan kao što se čini. Svaka slika na početku dobiva jednaku težinu. Nakon svake klasifikacije, težine pogrešno klasificiranih slika se povećavaju te se zatim isti proces ponavlja. Izračunavaju se nove stope pogrešaka i nove težine. Proces se nastavlja sve dok se ne postigne potrebna preciznost (dovoljno mala stopa pogrešaka) ili dok se ne pronađe potreban broj značajki. Konačni klasifikator je težinska suma slabih klasifikatora. Nazivamo ih „slabima“ jer sami ne mogu klasificirati sliku, no zajedno s ostalima čine moćan klasifikator. Rad u kojem se ova metoda predstavlja navodi da čak i 200 značajka omogućavaju detekciju s preciznošću od 95%, ali njihova posljednja verzija je imala oko 6000 (značajno manje od početnih 160

000+). No, optimizacija algoritma ne staje na tome. Autori rada primijetili su kako većina slika ne sadrži lica pa bi bilo bolje imati jednostavnu metodu koja provjerava je li uopće promatrani podprozor područje lica. Ako nije, odbacimo ga odmah i ne radimo na njemu daljnje proračune. Umjesto toga, posvećujemo više vremena provjeri potencijalnih područja lica. Ovime dolazimo do posljednjeg svojstva Viola-Jones algoritma: kaskadni klasifikatori. Umjesto primjene svih 6000 značajki na određeni podprozor, značajke su grupirane u različite faze klasifikatora i primjenjuju se jedna za drugom. Obično će prve faze sadržavati znatno manje značajki. Ako podprozor ne prođe prvu fazu, odbacuje se i ne razmatraju se preostale značajke na njemu. Ako prođe, primjenjuje se druga faza značajki i proces se nastavlja. Podprozor koji prođe sve faze smatra se područjem lica. [2]

3. Prepoznavanje emocija - DeepFace

DeepFace je jednostavan Python framework za prepoznavanje lica i analizu atributa lica (dob, spol, emocije, rasa). To je hibridni framework za prepoznavanje lica koji obuhvaća najmodernije modele: VGG-Face, Google FaceNet, OpenFace, Facebook DeepFace, DeepID, ArcFace, Dlib i SFace. Istraživanja pokazuju da ljudi s preciznošću od 97.35% prepoznaju lica, a ovi modeli to i premašuju. [3]

DeepFace je sustav dubokog učenja koji koristi devetoslojnu neuronsku mrežu s preko 120 milijuna veza. Facebook-ov DeepFace treniran je koristeći podatke o slikama korisnika (Social Face Classification baza podataka) - četiri milijuna portreta koji pripadaju gotovo četiri tisuće ljudi, od kojih je 5% slika ostavljeno za testiranje. Korištene su slike sa zabilježenim datumom kako bi se simuliralo kontinuirano prepoznavanje kroz starenje. Veliki broj slika po osobi daje jedinstvenu priliku za učenje invarijantnosti potrebne za temeljni problem prepoznavanja lica. Također su korištene i Labeled Faces in the Wild baza podataka koja sadrži fotografije poznatih osoba i YouTube Faces baza podataka, podskup baze poznatih osoba, ali za prepoznavanje lica u videozapisu. [5]

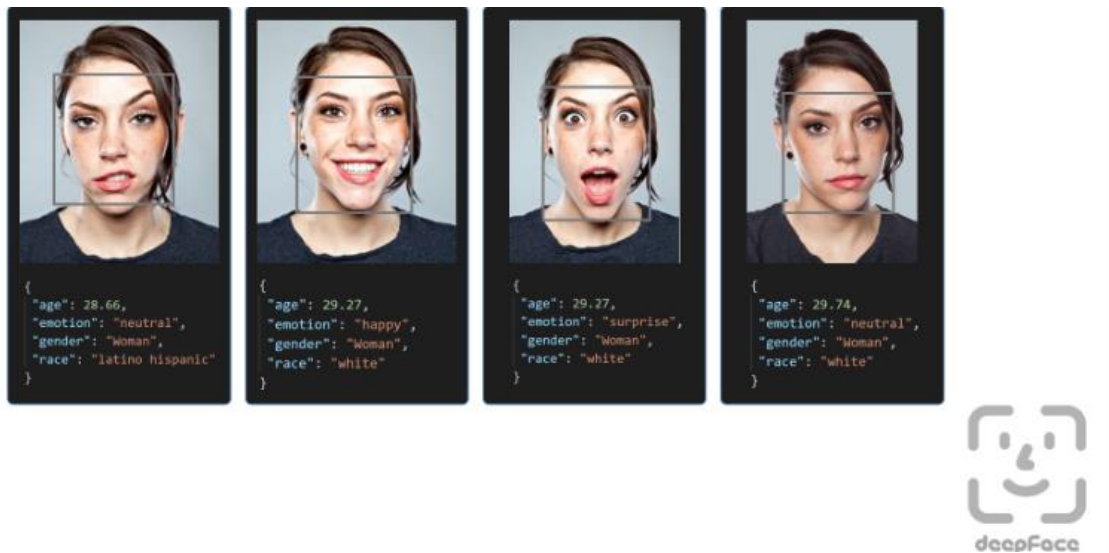
DeepFace postavlja faze poravnanja i predstavljanja na višu razinu. Koristeći trodimenzionalni model prosječnog čovjeka, softver postavlja lice da gleda naprijed. Ovo je opcionalan korak, kojeg nismo koristile u ovom radu, nego smo promatrale samo lica okrenuta prema naprijed. Zatim se pomoću simulirane neuronske mreže izračuna numerički opis (u obliku vektora) polaznog (preorijentiranog) lica. Funkcija DeepFace.verify() pronalazi imaju li dvije slike

dovoljno sličan numerički opis te ako da, vraća da te dvije slike prikazuju isto lice. Ona može i na slikama s više lica prepoznati ona koja se najviše podudaraju. Podudaranje ili čak sličnost između dvije osobe se može izračunati pomoću više metrika, na primjer kosinusne sličnosti, Euklidske norme i L2-norme. Po defaultu koristi se kosinusna sličnost iako istraživanja pokazuju da L2-norma daje točnije rezultate. Sve podatke koje računa verifikacijska funkcija prikazani su na slici 3. [3] [4] [5]



Slika 3: podaci koje vraća funkcija DeepFace.verify() (<https://pypi.org/project/deepface/>)

Deepface ima i funkciju *analyze()* koja proučava attribute lica te na temelju toga procjenjuje dob (eng. age), spol (eng. gender), izraz lica/emocije (eng. emotion) (ljutnja eng. angry, gađenje eng. disgust, strah eng. fear, sreća eng. happy, tuga eng. sad, iznenađenje eng. surprise, neutralno eng. neutral) i rasu (eng. race) (azijsku, bijelu, srednjeeuropsku, indijsku, latino i crnu) osobe na slici. Sve četiri značajke imaju jako dobru točnost, dob procjenjuje do na ± 4.65 godina, spol čak 97.44% točno, emocije 96.29%, a rasu sa 95.05% preciznosti. Primjer izračuna te funkcije prikazan je na slici 4 gdje se vidi da je na sve četiri slike ista osoba, kojoj model procjenjuje od 28.66 do 29.74 godina, ženski spol, rasu latino ili bijelu te emocije ovisno o izrazu lica na pojedinoj fotografiji. U ovome radu koristile smo samo značajku za prepoznavanje emocija. [3][6]



Slika 4: podaci koje vraća funkcija `Deepface.analyze()` (<https://pypi.org/project/deepface/>)

4. Implementacija algoritma

```
import cv2
from deepface import DeepFace
import numpy as np

face_cascade = cv2.CascadeClassifier(cv2.data.haarcascades + "haarcascade_frontalface_alt2.xml")
video = cv2.VideoCapture(0, cv2.CAP_DSHOW)

if not video.isOpened():
    raise IOError("Cannot open webcam")

while video.isOpened():
    _, frame = video.read()

    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    face = face_cascade.detectMultiScale(gray, scaleFactor=1.1, minNeighbors=5)
    for x, y, w, h in face:
        try:
            image = cv2.rectangle(frame, (x, y), (x + w, y + h), (199, 21, 133), 1)
            analyze = DeepFace.analyze(frame, actions=['emotion'])

            if analyze[0]['dominant_emotion'] == 'angry':
                cv2.putText(image, analyze[0]['dominant_emotion'], (x, y), cv2.FONT_HERSHEY_SIMPLEX, 1, (12, 12, 232), 2)

            elif analyze[0]['dominant_emotion'] == 'disgust':
                cv2.putText(image, analyze[0]['dominant_emotion'], (x, y), cv2.FONT_HERSHEY_SIMPLEX, 1, (52, 235, 79), 2)

            elif analyze[0]['dominant_emotion'] == 'fear':
                cv2.putText(image, analyze[0]['dominant_emotion'], (x, y), cv2.FONT_HERSHEY_SIMPLEX, 1, (235, 52, 214), 2)

            elif analyze[0]['dominant_emotion'] == 'happy':
                cv2.putText(image, analyze[0]['dominant_emotion'], (x, y), cv2.FONT_HERSHEY_SIMPLEX, 1, (52, 192, 235), 2)

            elif analyze[0]['dominant_emotion'] == 'sad':
                cv2.putText(image, analyze[0]['dominant_emotion'], (x, y), cv2.FONT_HERSHEY_SIMPLEX, 1, (235, 95, 52), 2)

            elif analyze[0]['dominant_emotion'] == 'surprise':
                cv2.putText(image, analyze[0]['dominant_emotion'], (x, y), cv2.FONT_HERSHEY_SIMPLEX, 1, (215, 163, 240), 2)

            else:
                cv2.putText(image, analyze[0]['dominant_emotion'], (x, y), cv2.FONT_HERSHEY_SIMPLEX, 1, (115, 111, 114), 2)
                print(analyze[0]['dominant_emotion'])
        except:
            print("No face")

    cv2.imshow('video', frame)
    key = cv2.waitKey(1)
    if key == ord('q'):
        break

video.release()
cv2.destroyAllWindows()
```

Izvor: [7]

Na početku samog koda, morale smo uvesti potrebne biblioteke: `cv2` – za zadatke računalnog vida, `DeepFace` – za analizu atributa lica i `numpy` – za operacije s brojevima. Zatim učitavamo

XML datoteku *Haar Cascade Classifier*-a sa istreniranim modelom za detekciju lica okrenutih prema naprijed (eng. *frontalface*). Budući da želimo lice detektirati pomoću web kamere, otvaramo kameru ili vraćamo error te ispisujemo poruku „*Cannot open webcam*“, ako kamera nije uspješno upaljena. Sve dok je kamera upaljena, svaki okvir koji kamera vidi, pretvara se u crno-bijelu sliku koje se koriste u Viola-Jones algoritmu za detekciju lica. Funkcija *detectMultiScale()* pronalazi lice na crno-bijeloj slici koju prima kao prvi argument, drugi argument *scaleFactor* skalira sliku kao bi se jednako dobro prepoznavala udaljena lica kao i ona blizu, dok treći argument specificira koliko susjeda svaki potencijalni pravokutnik treba imati kako bi ga zadržao. Ta funkcija vraća uređenu četvorku (x, y, w, h) gdje su (x, y) koordinate gornjeg lijevog ugla pravokutnika, a (w, h) širina i visina pravokutnika u kojemu je pronađeno lice. For petlja iterira po svakom takvom pravokutniku, prikazuje ga na video web kamere te analizira svako lice koristeći *DeepFace*. Funkcija *DeepFace.analyze(frame, actions=['emotion'])* vraća postotke svih promatranih emocija i na temelju najvećeg postotka određuje koja je najzastupljenija emocija te ju sprema kao '*dominant_emotion*'. Zatim se, ovisno o tome koja je dominantna emocija pronađena, ona ispisuje iznad gornjeg ruba pravokutnika u boji koja ju karakterizira (ljutnja = crvena, gađenje = zelena, strah = ljubičasta, sreća = žuta, tuga = plava, iznenađenje = ružičasta, neutralno = siva). Ako lice uopće nije pronađeno, ispisuje se poruka „*No face*“. Kamera se gasi pritiskom tipke *q* na tipkovnici čime završava izvršavanje programa.

Za pokretanje ovog programa morale smo instalirati *DeepFace* paket naredbom *pip install deepface*.

5. Zaključak

Zadnjih godina počinje se isticati sve veća važnost tehnologije prepoznavanja emocija i njihove detekcije. Sve više tehnoloških tvrtki i startup-a uključeni su u razvoj i marketing ove tehnologije. S većim računalnim mogućnostima, poboljšanim modelima dubokog učenja i dostupnošću skupova podataka na društvenim mrežama, tehnologija prepoznavanja emocija doseže nove prekretnice, potaknuta raznolikim oblicima podataka uključujući videozapise, audiozapise i slike. Modeli prepoznavanja i detekcije emocija koriste se u brojnim područjima kao što su primjerice kod pametnih automobila koji mogu detektirati kada se vozač osjeća pospano ili u testiranju tržišta zamjenom anketa o zadovoljstvu kupaca sa automatskim mjerenjem izraza lica i emocija kako bi se bolje razumjele njihove želje i potrebe. Tvrtka Unilever je čak integrirala ovu tehnologiju u svoj proces zapošljavanja kako bi procijenila razinu povjerenja kandidata, pomažući u donošenju odluka za uloge okrenute klijentima. Međutim, potrebno je i usavršavati ovu tehnologiju radi smanjenja pristranosti i poboljšanja točnosti, posebno u kritičnim situacijama donošenja odluka. [8]

U ovom seminaru opisan je sam proces detekcije i prepoznavanja izraza lica i emocija. Opisan je Viola-Jones algoritam koji je 2001. doveo veliki napredak u detekciji lica te algoritam DeepFace predstavljen 2014., koji koristeći višeslojnu neuronsku mrežu s ulaznim parametrima od 4 milijuna portreta, nadmašuje većinu postojećih sustava za prepoznavanje lica te po preciznosti nadmašuje razinu čovjeka. Od tada tehnologije i tržište prepoznavanja emocija jako su napredovali, a vjerujemo da će se ovaj trend razvoja nastaviti u dobrom smjeru otkrivanja novih primjena i poboljšanju postojećih algoritama.

6. Literatura

- [1] <https://repozitorij.unin.hr/islandora/object/unin:5019/datastream/PDF/download>
- [2] https://docs.opencv.org/3.4/db/d28/tutorial_cascade_classifier.html
- [3] <https://pypi.org/project/deepface/>
- [4] <https://www.cbsnews.com/news/facebooks-deepface-shows-serious-facial-recognition-skills/>
- [5] "DeepFace: Closing the Gap to Human-Level Performance in Face Verification". Facebook-ovo istraživanje. Preuzeto 16-02-2024.
- [6] <https://sefiks.com/2018/08/13/cosine-similarity-in-machine-learning/>
- [7] <https://girolamopinto1.medium.com/emotion-detection-in-python-8642fade50a9>
- [8] <https://indiaai.gov.in/article/real-time-emotion-recognition-potential-use-cases-and-challenges>