

# Razvoj i evaluacija softverskog alata za identifikaciju lica koristeći tehnike strojnog i dubokog učenja

---

**Jurić, Antonio**

**Undergraduate thesis / Završni rad**

**2024**

*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:235:620753>

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

*Download date / Datum preuzimanja:* **2025-03-07**

*Repository / Repozitorij:*

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



SVEUČILIŠTE U ZAGREBU  
FAKULTET STROJARSTVA I BRODOGRADNJE

# ZAVRŠNI RAD

**Antonio Jurić**

Zagreb, 2024.

SVEUČILIŠTE U ZAGREBU  
FAKULTET STROJARSTVA I BRODOGRADNJE

# ZAVRŠNI RAD

Mentori:

izv. prof. dr. sc. Tomislav Stipančić

Student:

Antonio Jurić

Zagreb, 2024.

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

Zahvaljujem se svojem mentoru izv. prof. dr. sc. Tomislavu Stipančiću na pomoći tijekom pisanja ovog rada.

Antonio Jurić



Sveučilište u Zagrebu	
Fakultet strojarstva i brodogradnje	
Datum	Prilog
Klasa: 602 – 04 / 24 – 06 / 1	
Ur.broj: 15 – 24 –	

## ZAVRŠNI ZADATAK

Student: **Antonio Jurić**

JMBAG: **0035239799**

Naslov rada na hrvatskom jeziku: **Razvoj i evaluacija softverskog alata za identifikaciju lica koristeći tehnike strojnog i dubokog učenja**

Naslov rada na engleskom jeziku: **Development and evaluation of a facial identification software tool using machine and deep learning techniques**

Opis zadatka:

S obzirom na rastuću važnost umjetne inteligencije u modernim tehnološkim aplikacijama, posebice u sigurnosnim sustavima, kontroli pristupa i osobnoj identifikaciji, precizno prepoznavanje lica postaje ključno u mnogim industrijama. Umjetna inteligencija nudi revolucionarne metode koje omogućuju brzu, točnu i automatiziranu identifikaciju osoba čak i u dinamičnim i nepredvidivim okruženjima, što tradicionalne metode ne mogu uvijek efikasno ostvariti.

U radu je potrebno razviti sofisticirani softverski alat za identifikaciju osoba na temelju lica, koristeći TensorFlow biblioteku za strojno učenje, sljedeći sljedeće korake:

- istražiti i implementirati tehnike za detekciju i prepoznavanje lica u realnom vremenu
- detaljno se upoznati s TensorFlow bibliotekom i primijeniti odgovarajuće metode za identifikaciju lica
- razviti metodologiju prepoznavanja koristeći koncepte prenesenog učenja (engl. Transfer Learning), konvolucijskih neuronskih mreža i tehnika temeljenih na ugrađenim karakteristikama (engl. Embeddings)
- ocijeniti karakteristike modela s obzirom na različite kutove promatranja lica.

Razvijeni softverski sustav treba evaluirati eksperimentalno s uključenjem stvarnih osoba kako bi se potvrdila njegova efikasnost.

U radu je potrebno navesti korištenu literaturu te eventualno dobivenu pomoć.

Zadatak zadan:

24. 4. 2024.

Zadatak zadao:

izv. prof. dr. sc. Tomislav Stipančić

Datum predaje rada:

2. rok (izvanredni): 11. 7. 2024.  
3. rok: 19. i 20. 9. 2024.

Predviđeni datumi obrane:

2. rok (izvanredni): 15. 7. 2024.  
3. rok: 23. 9. – 27. 9. 2024.

Predsjednik Povjerenstva:

izv. prof. dr. sc. Petar Ćurković

## Sadržaj

POPIS SLIKA.....	II
SAŽETAK.....	III
SUMMARY .....	IV
UVOD.....	1
1. UMJETNA INTELIGENCIJA (UI) .....	2
2. STROJNO UČENJE.....	3
3. DUBOKO UČENJE .....	4
4. RAČUNALNI VID .....	5
5. NEURONSKE MREŽE .....	6
5.1. Biološke neuronske mreže .....	6
5.2. Umjetne neuronske mreže .....	7
5.3. Arhitektura umjetne neuronske mreže.....	8
5.4. Učenje neuronske mreže.....	9
5.5. Konvolucijske neuronske mreže .....	11
6. METODE ZA DETEKCIJU I IDENTIFIKACIJU LICA .....	14
6.1. Tehnike prepoznavanja lica .....	15
6.2. Face ID .....	16
7. PREGLED KORIŠTENIH TEHNOLOGIJA .....	18
7.1. Python programski jezik.....	18
7.2. Google Colab .....	19
8. PROGRAM ZA IDENTIFIKACIJU LICA .....	20
8.1. Ulazni skup podataka .....	20
8.2. Parametri neuronske mreže.....	22
8.3. Prepoznavanje lica s kamerom u stvarnom vremenu .....	27
9. REZULTATI.....	31
10. ZAKLJUČAK.....	35
LITERATURA .....	36
PRILOG .....	39

**POPIS SLIKA**

Slika 1: Ciklus strojnog učenja [10] .....	3
Slika 2: Biološki neuron [22] .....	66
Slika 3: Usporedba biološkog i umjetnog neurona [6] .....	77
Slika 4: Neuron u umjetnoj neuronskoj mreži [7] .....	88
Slika 5: Arhitektura neuronske mreže [7] .....	99
Slika 6: Učenje neuronske mreže s ranim zaustavljanjem [5] .....	1010
Slika 7: Arhitektura konvolucijske neuronske mreže [10] .....	111
Slika 8: Operacija konvolucije [10] .....	111
Slika 9: Dodavanje nulte dopune [10] .....	122
Slika 10: Vrste udruživanja [11] .....	133
Slika 11: Izravnavanje [10] .....	133
Slika 12: Prepoznavanje lica u zračnoj luci [12] .....	144
Slika 13: 3D model ljudskog lica [15] .....	15
Slika 14: Prepoznavanje lica toplinskom kamerom [15] .....	1616
Slika 15: Face ID [14] .....	1717
Slika 16: Umjetno povećanje ulaznih podataka .....	2121
Slika 17: Pogreške mreže s 56 i 20 slojeva [23] .....	2323
Slika 18: Usporedba algoritama za optimizaciju [26] .....	2525
Slika 19: Prenešeno učenje [28] .....	2626
Slika 20: Proširivanje slike [31] .....	2929
Slika 21: Primjer rezultata .....	3030
Slika 22: Proces treniranja mreže .....	3131
Slika 23: Normalan pogled .....	3232
Slika 24: Zakret u desno .....	3232
Slika 25: Zakret u lijevo .....	3232
Slika 26: Zakret prema gore .....	3333
Slika 27: Zakret prema dolje .....	333

---

## SAŽETAK

Umjetna inteligencija je danas zastupljena u svakom području našega života, počevši od našeg obrazovanja i pomoći pri učenju, pa sve do zabave i odabira filmova i pjesama u slobodno vrijeme. Jedna od primjena je također računalni vid, u ovom slučaju sustav za prepoznavanje lica koji se koristi u sigurnosne svrhe. Glavni razlog njihove zastupljenosti je mogućnost učenja čime se oponaša ljudski mozak koji funkcionira na isti način. Evaluacijom takvog sustava analiziramo njegovu točnost, robusnost i brzinu.

U ovome radu je kreiran sustav za prepoznavanje ljudskih lica u realnom vremenu. Korištena su dva sustava koji se temelje na konvolucijskoj neuronskoj mreži, jedan za detekciju lica na slici, a drugi za prepoznavanje lica pojedine osobe. Cijeli rad se temelji na prenešenom učenju, tj. korištenju znanja stečenog iz jednog modela za rješavanje sličnog problema. U radu se također nalaze i ostvareni rezultati treniranja modela.

**Ključne riječi:** umjetna inteligencija, računalni vid, konvolucijske neuronske mreže, prenešeno učenje



---

## SUMMARY

Artificial intelligence is an interdisciplinary field that focuses on developing systems and algorithms enabling computers to perform tasks typically requiring human intelligence. Today, it is present in every aspect of our lives, from our education and learning assistance to entertainment and the selection of movies and songs in our free time. One of its applications is also a facial recognition system used for security purposes. The main reason for their presence is the possibility of learning, which imitates the human brain, which functions in the same way. By evaluating such systems, we analyze its accuracy, robustness and speed.

In this work, a real-time human faces recognition system was created. Two systems based on neural networks were used: one for face detection in an image and another for recognizing the face of an individual. The entire work is based on transfer learning, i.e. using the knowledge gained from one model to solve a similar problem. The work also includes the achieved results of the model training.

**Keywords:** artificial intelligence, computer vision, convolutional neural network, transfer learning

## UVOD

Računalni vid je područje umjetne inteligencije koje omogućuje računalima da interpretiraju vizualne informacije kao što to čine ljudi, pouzdanost takvih sustava nikada neće biti 100%. U kontekstu prepoznavanja lica, računalni vid omogućava detekciju, prepoznavanje i verifikaciju lica.

Kako bi model bio pouzdan, potrebno ga je trenirati na što većem i raznovrsnijem skupu podataka. Skup podataka se može podijeliti u tri skupine, a to su trening skup koji se koristi za treniranje modela, validacijski skup koji se koristi za podešavanje parametara modela i njegovu težinu te testni skup koji se koristi za konačnu evaluaciju modela nakon treniranja.

U okviru ovog rada najprije se objašnjavaju pojedine podskupine koje su bitne za razumijevanju rada, kao što su umjetna inteligencija, neuronske mreže itd.

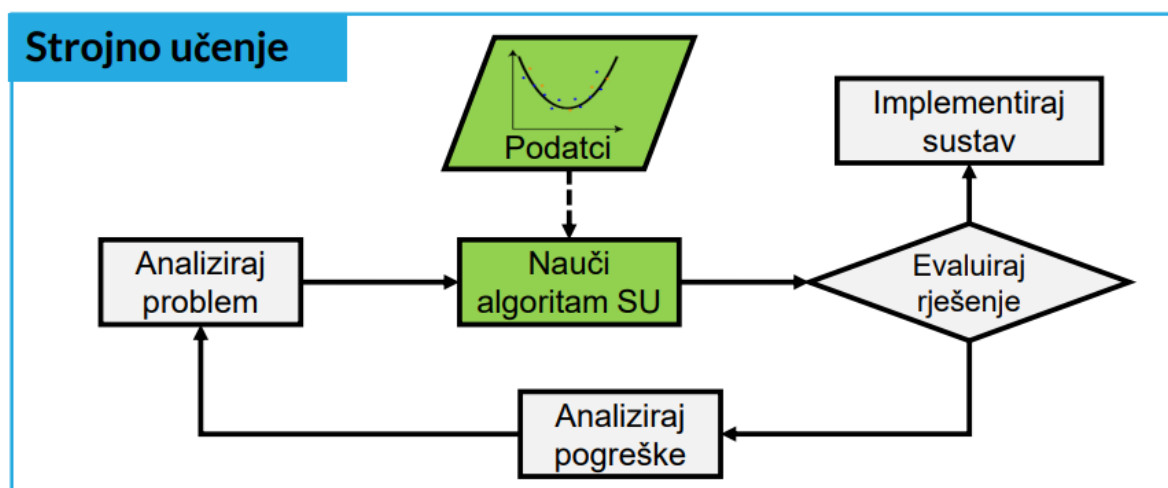
---

## 1. UMJETNA INTELIGENCIJA (UI)

Umjetna inteligencija je interdisciplinarno područje koje se bavi razvojem sustava i algoritama koji omogućuju računalima da izvršavaju zadatke koji obično zahtijevaju ljudsku inteligenciju. Umjetna inteligencija nije nova tehnologija jer neke tehnologije umjetne inteligencije postoje već desetljećima, ali pojava novih softvera i dostupnost velikih količina podataka su doveli do velikog napretka. Zbog široke primjene u obrazovanju, medicini, prometu, proizvodnji i sličnom, ovo je samo jedna od brojnih definicija koje možemo pronaći na internetu. Potrebno je poznavanje statistike, računarske znanosti, neuroznanosti, psihologije, analize podataka i drugih disciplina za potpuno razumijevanje. Glavni cilj umjetne inteligencije je automatizacija raznih poslova koji su monotoni za ljude, smanjenje faktora ljudske greške, predikcija budućih događaja na temelju povijesnih podataka te općenito poboljšanje kvalitete života [1]. Naravno, za svaku granu i znanstvenu disciplinu postoje specifični ciljevi i prednosti. Umjetna inteligencija se klasificira kao slaba i jaka UI. Slaba UI obavlja specifične zadatke bez svijesti ili razumijevanja, to uključuje aplikacije kao što su prepoznavanje teksta i slika. Jaka UI je teoretski koncept i podrazumijeva računalne sustave s ljudskom razinom svijesti i razumijevanja, ali takvi sustavi još uvijek nisu razvijeni [2]. Umjetna inteligencija obuhvaća različita područja poput: strojnog učenja, dubokog učenja, računalnog vida, automatizacije i robotike...

## 2. STROJNO UČENJE

Strojno učenje je jedna od kategorija umjetne inteligencije i uvijek se povezuju jedan uz drugi. Strojno učenje nastoji konstruirati i stvoriti algoritme i sustave koji se mogu vrlo lako prilagoditi novim i neviđenim situacijama jer uče na temelju iskustva. Dakle, strojno učenje predstavlja sposobnost programa da nauči zadatak ili problem bez eksplicitnog programiranja za njega. Kroz algoritme se otkrivaju obrasci i uče kako predviđati i dati preporuke. Na slici 1 je prikazan ciklus učenja, kreće se s analizom problema, zatim se modelu daju podaci za učenje, evaluiraju se dobiveni rezultati te se model implementira ako model zadovoljava i nema grešaka, a ako postoje greške, one se analiziraju i ponovno se vraća na prvi korak. Kao i umjetna inteligencija, strojno učenje ima jako široko područje primjene. Algoritmi strojnog učenja se mogu klasificirati 3 kategorije, nadzirano učenje, podržano učenje i nenadzirano učenje. Nadzirano strojno učenje je učenje pomoću podataka koji sadrže informacije o klasama i pripadajućim vrijednostima, dakle učenje pomoću označenih podataka, a najčešće se koriste za predikciju. Metode nadziranog učenja su klasifikacija, regresija i predviđanje. Podržano učenje je vrsta gdje agent uči kroz pokušaje i greške, koristeći povratne informacije iz okruženja. Dakle, uči kroz iskustvo te se svaka dobra akcija nagrađuje, a svaka loša kažnjava. Nenadzirano učenje koristi ulazne podatke bez oznaka, cilj je otkriti skrivene strukture i obrasce u podacima bez eksplicitnih izlaznih vrijednosti, što ga čini izuzetno korisnim u različitim istraživanjima [3].



Slika 1: Ciklus strojnog učenja [10]

---

### 3. DUBOKO UČENJE

Duboko učenje je područje umjetne inteligencije, podvrsta strojnog učenja, usredotočeno na stvaranje velikih modela neuronskih mreža sposobnih za donošenje odluka temeljenih na dostupnim podacima. Pridjev „duboko“ se odnosi na korištenje više slojeva u mreži. Duboko učenje omogućava modelima da uče složene funkcije direktno iz podataka, što je korisno u zadacima poput prepoznavanja slika i obrade jezika. Neuronske mreže su osnovni gradivni elementi dubokog učenja koje se sastoje od slojeva neurona povezanih sinapsama, neuroni označavaju jedinice, a sinapse težine. Dokazano je da je efikasnije koristiti mrežu s više slojeva s manje neurona nego manje slojeva s više neurona. Jedna od primjena je također prepoznavanje lica pojedinih osoba. Implementacija dubokog učenja zahtijeva veliki skup podataka s oznakama te kvalitetnu grafičku karticu za što brže treniranje modela, što je ionako zahtjevan proces. Prednosti dubokog učenja su visoka točnost na dovoljno velikim i kvalitetnim skupovima podataka, mogućnost primjene na širok spektar problema, od vizualnih zadataka do obrade zvuka i jezika.

---

## 4. RAČUNALNI VID

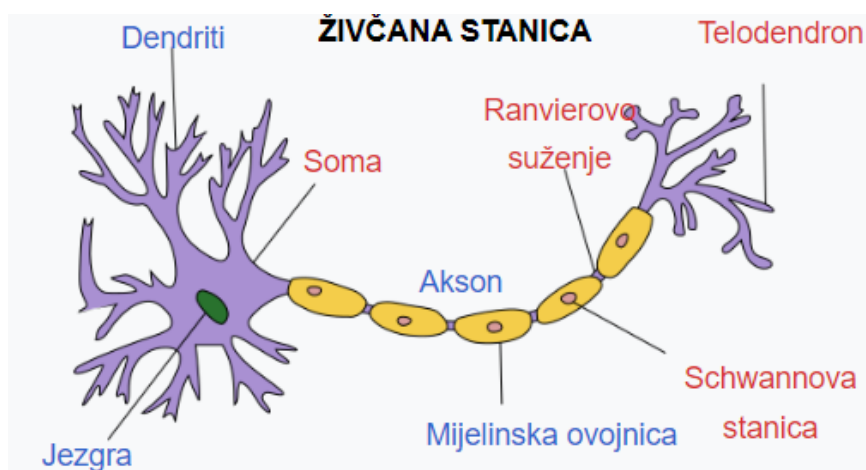
Računalni vid je područje umjetne inteligencije čiji je primarni cilj stvaranje digitalnih sustava koji mogu analizirati, obraditi i dati smisao vizualnim podacima na isti način na koji to čine ljudi. Dakle, pomoću računalnog vida izvlačimo korisne informacije iz slika i videozapisa s ciljem donošenja odluka i za rješavanje različitih problema. Danas se računalni vid najčešće povezuje s dubokim učenjem gdje se u pozadini analiziraju dobiveni podatci i donose zaključci temeljeni na njima. Računalni vid je zastupljen u svim modernijim granama industrija, primjerice autonomna vozila koriste računalni vid za percepciju okoline (kamere, radari, LIDAR), u medicinskoj dijagnostici se analiziraju rendgenske snimke, MRI i CT skenovi, također i u zabavi, za kreiranje vizualnih efekata te implementaciju proširene stvarnosti AR [4]. Jedan od najistaknutijih primjera je u sigurnosti i nadzoru, gdje računalni vid omogućava prepoznavanje lica te detekciju sumnjivog ponašanja, starije tehnologije su prepoznavale lice na temelju javno dostupnih slika i uspoređivale slike sa osobom, ali moderniji i puno efikasniji način je sa neuronskim mrežama gdje će model naučiti prepoznavati određenu osobu na temelju svojih zaključaka iz pojedinih slika. Da bi se replicirao ljudski vid na računalu, koristi se mnogo različitih postupaka kao što su:

- Detekcija objekata – identifikacija specifičnih objekata na slikama i videozapisima
- Prepoznavanje lica – napredni oblik detekcije objekata koji se često koristi za biometrijsku sigurnost
- Detekcija rubova – postupak koji se koristi za poboljšanje kvalitete slike putem prepoznavanja rubova slika i videozapisa
- Segmentacija slike – proces razdvajanja slike na manje dijelove koji su važni za određeni problem koji se analizira
- Klasifikacija objekata – nakon identifikacije objekata na slici, svrstavaju se u pripadajuće kategorije [5]

## 5. NEURONSKE MREŽE

### 5.1. Biološke neuronske mreže


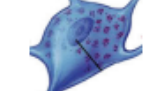


Biološke neuronske mreže su složeni sustavi koji se sastoje od velikog broja međusobno povezanih neurona, među kojima je omogućena komunikacija putem električnih i kemijskih signala [21]. Čine osnovu za funkcioniranje živčanog sustava svih živih bića te su odgovorni za sve aspekte ponašanja, od najjednostavnijih refleksa do najsloženijih misaonih procesa. Neuron je osnovna jedinica živčanog sustava i sastoji se od dendrita, tijela stanice i aksona, slika 2. Dendriti su kratke i razgranate strukture koje primaju signale od drugih neurona i prenose ih prema tijelu neurona. Tijelo stanice ili soma je glavni dio neurona jer sadrži jezgru i većinu staničnih organela. Akson je duža, tanka struktura koja prenosi impuls od tijela neurona prema drugim neuronima ili mišićima [6]. Prijenos signala se odvija putem impulsa, kada neuron primi dovoljno jak signal od drugih neurona, generira impuls koji putuje duž aksona do sinapse (mjesto gdje akson jednog neurona dolazi u kontakt s dendritom drugog neurona). Na sinapsi, signal potiče otpuštanje neurotransmitera koji prelaze sinaptičku pukotinu i vežu se za receptore na dendritima susjednog neurona, što potiče generiranje novog impulsa u tom neuronu.



Slika 2: Biološki neuron [22]

Tijekom evolucije ljudski mozak je poprimio niz karakteristika koje se povezuju s inteligencijom, kao što su paralelno odvijanje više operacija, sposobnost prilagođavanja, sposobnost učenja, tolerancija na pogreške i nepotpune informacije. Glavna karakteristika računalnih sustava je brzo rješavanje numerički složenih zadataka te rad s velikom količinom

podataka. Dakle, možemo pronaći sličnosti u biološkoj i umjetnom neuronskoj mreži, kao što to prikazuje slika 3.

	<b>Biološki neuron</b>	<b>Umjetni neuron</b>
	Prima ulazni signal putem dendrida (sinaptičke veze)	Prima ulaze ( $i$ ) koji su određeni težinskim koeficijentima ( $w$ )
	Obrada signala u somi	Obrada ulaza, unutarnji prag – bias ( $b$ )
	Pretvara obrađeni ulaz u izlaz putem aksona	Pretvara ulaze u izlaz (prijenosna funkcija)
	Šalje informacije putem sinapsi do svih neurona s kojima je neuron povezan	Šalje informaciju prema izlazu i sljedećim neuronima

Slika 3: Usporedba biološkog i umjetnog neurona [6]

## 5.2. Umjetne neuronske mreže

Umjetna neuronska mreža je model koji oponaša strukture i funkcionalne aspekte bioloških neuronskih mreža. Sastoji se od niza međusobno povezanih umjetnih neurona ili čvorova od kojih svaki ima svoju težinu – ulazni sloj, jedan ili više skrivenih slojeva te izlazni sloj. Na slici 4 je prikazana struktura neurona u umjetnoj neuronskoj mreži. Neuronske mreže koriste podatke o obuci kako bi naučile i poboljšale točnost tijekom vremena. Svaki čvor predstavlja matematički model, kada se utvrdi ulazni sloj, dodjeljuju se težine koje određuju važnost pojedine varijable. Svi se ulazi množe sa svojim težinama te se nakraju svi zbroje, uključujući i pristranost, kao što je prikazano u jednadžbi 5.1. .

$$y = \sum_{i=1}^n w_i x_i + b = w_1 x_1 + w_2 x_2 + \dots + w_n x_n + b \quad (5.1.)$$

Zatim se koristi funkcija koja određuje važnost informacije koju nosi trenutni neuron za daljnji izračun. Postoje tri vrste funkcije [8]:

- Linearna funkcija (funkcija identiteta) – jednostavna funkcija gdje je izlaz proporcionalan ulazu, dakle izlaz je jednak ulazu, rijetko se koriste (najčešće u izlaznim slojevima za regresijske zadatke)

$$f(y) = y \quad (5.2.)$$

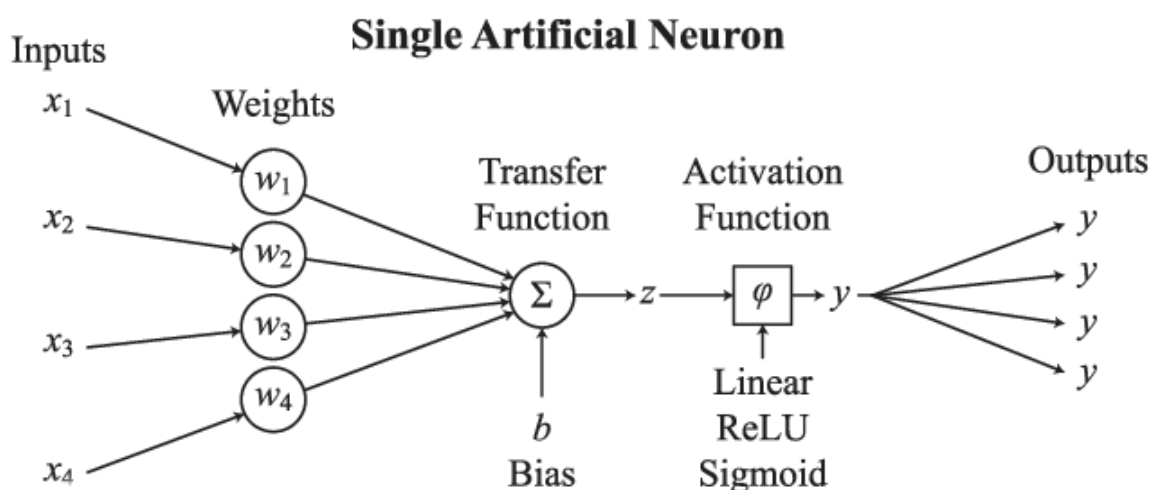


- Nelinearna funkcija – omogućuju modeliranje složenih odnosa između ulaza i izlaza, ključne jer omogućuju stvaranje višeslojnih modela s većom sposobnošću učenja. Postoji više vrsta ove funkcije, sigmoidna, tanh, ReLU, ELU... Danas je najkorištenija ReLU (eng. Rectified linear unit), nelinearna operacija kojoj je svrha uvesti nelinearnost u proces učenja budući da je većina podataka iz stvarnog svijeta nelinearna.

$$\text{ReLU}(y) = \max(0, y) \quad (5.3.)$$

- Funkcija skoka – ima izlaz koji „skače“ iz jedne vrijednosti u drugu bez kontinuiranog prijelaza. Također nisu često korištene jer ne pružaju glatke gradijente potrebne za učinkovito treniranje.

$$f(x) = \begin{cases} 0, & y < 0 \\ 1, & y \geq 0 \end{cases} \quad (5.4.)$$

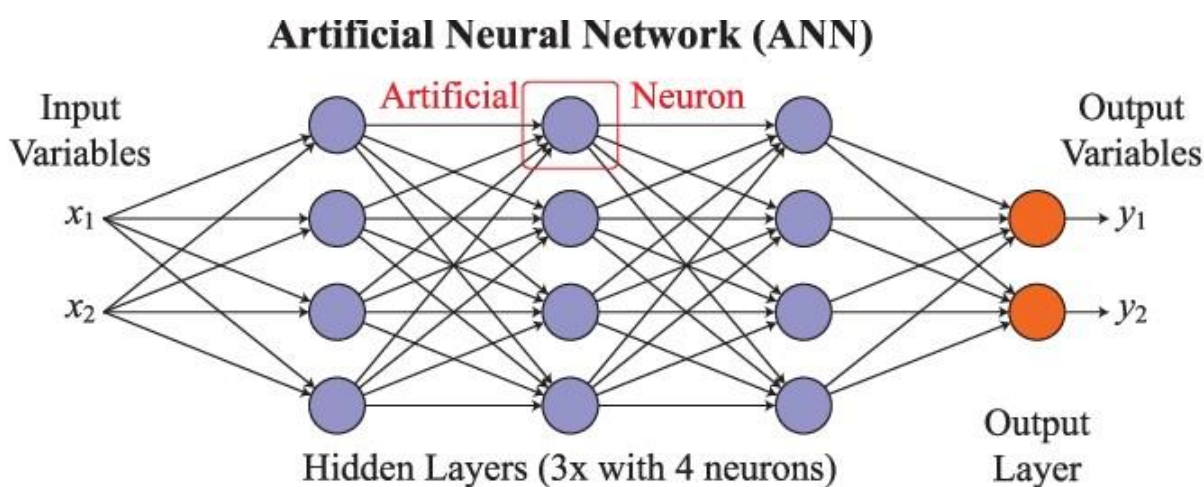


Slika 4: Neuron u umjetnoj neuronskoj mreži [7]

### 5.3. Arhitektura umjetne neuronske mreže

Umjetne neuronske mreže se sastoji od više slojeva međusobno povezanih neurona. Prvi sloj je ulazni sloj koji prima podatke i ne izvodi nikakve matematičke operacije već samo prenosi podatke u idući sloj. Broj neurona u ulaznom sloju odgovara dimenzionalnosti ulaznih podataka, dakle koliko ulaza toliko i neurona. Zatim slijedi jedan ili više skrivenih slojeva, broj

skrivenih slojeva utječe na dubinu neuronske mreže te ovisi o kompleksnosti zadatka i problema. Kao što vidimo na slici 5, svaki prethodni neuron je povezan sa svakim neuronom u sljedećem sloju. Skriveni slojevi izvode većinu računskih operacija i omogućavaju da mreža uči. Neke vrste skrivenih slojeva su konvolucijski slojevi, gusti slojevi, pooling slojevi... Na kraju se nalazi izlazni sloj koji daje konačne rezultate koji služe za donošenje odluka. Broj neurona u izlaznom sloju ovisi o zadatku, npr. broju klasa u klasifikaciji ili broju vrijednosti u regresiji.



Slika 5: Arhitektura neuronske mreže [7]

#### 5.4. Učenje neuronske mreže

Kako bi model bio što bolji i kvalitetniji, za učenje je potreban veliki skup označenih podataka odnosno ulaza. Taj skup se može podijeliti na tri kategorije:

- Skup podataka za učenje (70%) – koristi se za treniranje modela, točnije za prilagođavanje težina i parametara neuronske mreže. Ovaj skup je najveći i model prolazi nekoliko puta kroz te podatke kako bi naučio obrasce te bi bilo korisno da je reprezentativan, odnosno raznolik.
- Skup podataka za testiranje (15%) – koristi se za konačnu evaluaciju performansi modela nakon što je model potpuno naučen. Testiranje je potpuno nezavisan od procesa treniranja i provjere, čime se osigurava nepristranost evaluacije modela. Performanse

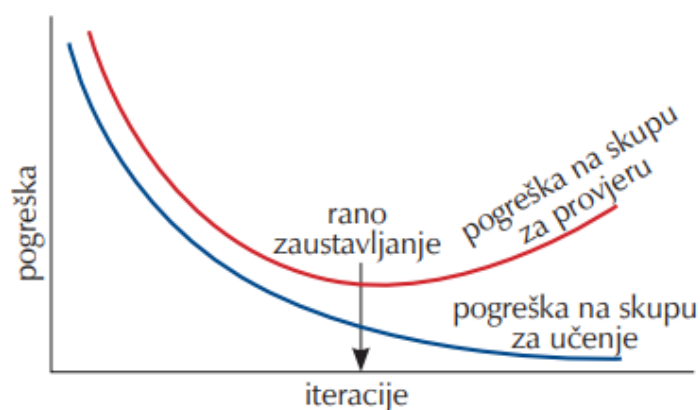
na ovom skupu će dati uvid u to koliko je model efikasan u realnim problemima, tj. kad se nađe u nepoznatim podacima.

- Skup podataka za provjeru (15%) – koristi se za fino podešavanje modela, izbor parametara i prevenciju prekomjernog prilagođavanja. Manji dio podataka izdvojen iz originalnog skupa, koristi se isključivo za evaluaciju performansi modela tijekom treniranja.

Jedna epoha predstavlja jedan prolazak kroz zadani trening skup te prilagodbu i optimiziranje parametara neuronske mreže. Na početku procesa učenja neuronske mreže težine postavljamo nasumično, stoga je potrebno definirati funkciju pogreške (engl. loss function) koja mjeri koliko je dobar model neuronske mreže u obavljanju određenog zadatka. Da bismo poboljšali neuronsku mrežu, moramo smanjiti vrijednost funkcije pogreške [9]. Postoje različite vrste funkcije pogreške kao što su srednja apsolutna greška, funkcija pogreške u unakrsnoj entropiji, a najčešće korištena je srednja kvadratna pogreška koja izračunava prosjek kvadrata razlike između predviđenih izlaznih vrijednosti i stvarnih izlaznih vrijednosti koje izračuna neuronska mreža:

$$L(\Theta) = \frac{1}{N} \sum_{i=0}^N (y_i - \hat{y}_i)^2 \quad (5.5.)$$

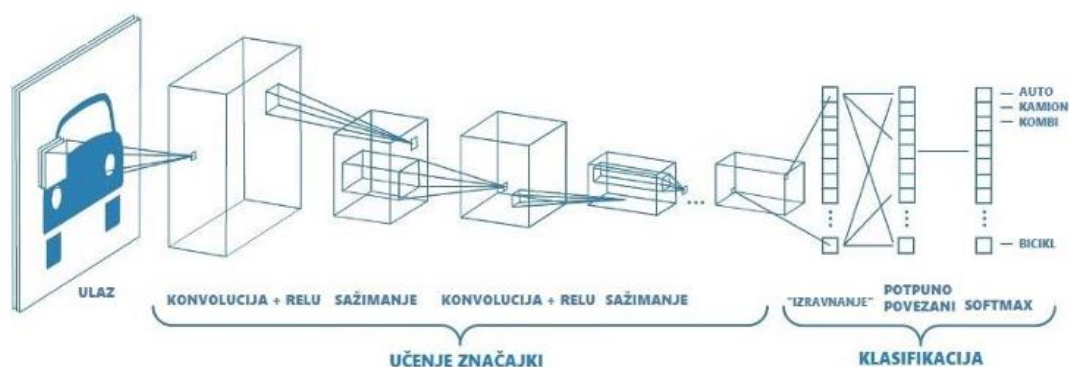
Na slici 6 vidimo tehniku učenja neuronske mreže pod nazivom rano zaustavljanje koja sprječava pretreniranost modela. Prije točke minimuma pogreške na skupu za provjeru i na skupu za učenje padaju, a nakon točke minimuma pogreška na skupu za učenje pada dok pogreška na skupu za provjeru raste, što ukazuje na pretreniranost modela.



Slika 6: Učenje neuronske mreže s ranim zaustavljanjem [5]

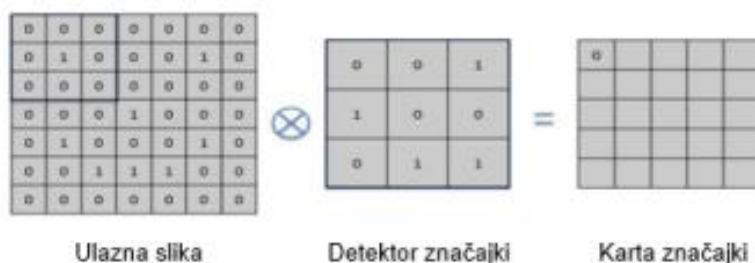
## 5.5. Konvolucijske neuronske mreže

Konvolucijske neuronske mreže su vrsta umjetnih neuronskih mreža za obradu nestrukturiranih podataka, a posebice slika, zvuka te videozapisa. Konvolucijska neuronska mreža se razlikuje od običnih neuronskih mreža po kompleksnijem skrivenom sloju koji se sastoji od konvolucijskog sloja, sloja udruživanja, aktivacijskog sloja, sloja izravnavanja te potpune veze, kao što to prikazuje slika 7.



Slika 7: Arhitektura konvolucijske neuronske mreže [10]

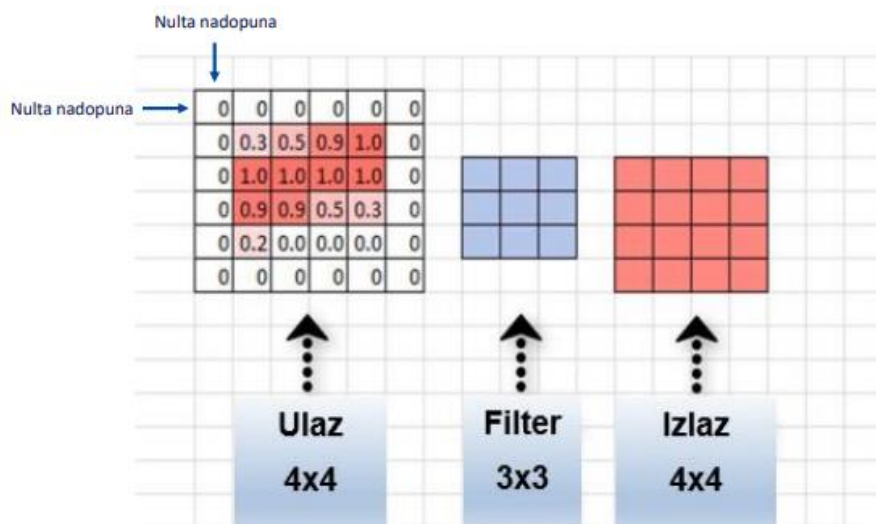
Konvolucijski sloj je prikazan na slici 8 te je to prvi sloj u konvolucijskim neuronskim mrežama. Konvolucija je matematička operacija ili proces izdvajanja značajki iz ulaznih slika. Svaka slika se smatra matricom vrijednosti piksela. Konvolucija čuva prostorne odnose između piksela učenjem značajki iz izvorne slike koristeći detektor značajki (eng. Filter, Kernel) – mali kvadrat ulaznih podataka [10]. Matrica formirana klizanjem filtera preko slike i izračunavanjem točkastog produkta se naziva karta značajki (eng. Activation Map, Feature map).



Slika 8: Operacija konvolucije [10]

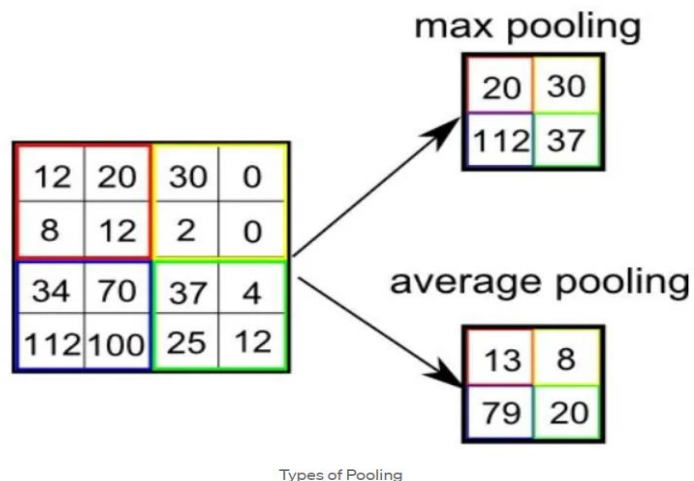
Postoji nekoliko operacija konvolucije kao što su dubina, korak, nulta dopuna te ReLU sloj. Dubina (eng. Depth) odgovara broju filtera koji se koriste za operaciju konvolucije, što je veći broj filtera to je i točniji rezultat. Korak (eng. Stride) je broj piksela za koji se filterska matrica

pomiče preko ulazne matrice, veći korak će proizvesti manje karte značajki. Dodavanje nulte dopune (eng. Zero-padding) se naziva široka konvolucija, a izostanak nulte dopune uska konvolucija. Ponekad je poželjno nadopuniti izvornu ulaznu matricu nulama oko granice da se može primijeniti filter na granične elemente. Prednost nulte dopune je to što omogućava kontrolu veličine karte značajki, kao što to vidimo na slici 9.



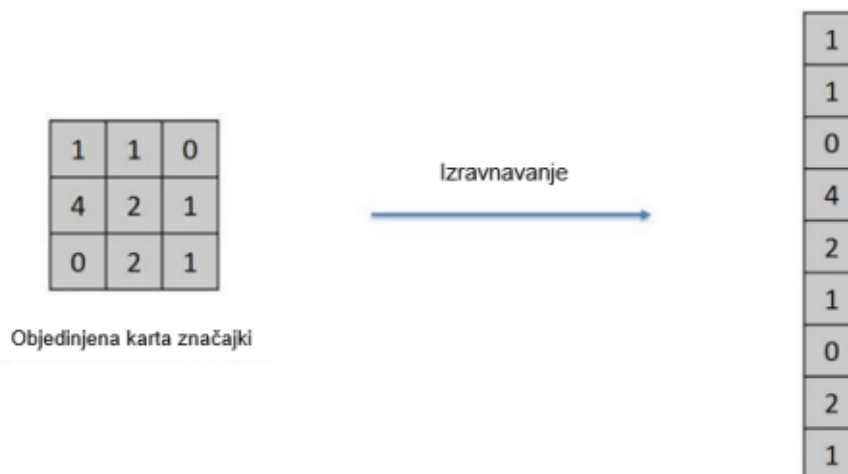
Slika 9: Dodavanje nulte dopune [10]

ReLU sloj je nelinearna operacija, koristi se nakon svake operacije konvolucije. To je operacija primijenjena po pikselu i zamjenjuje sve negativne vrijednosti piksela u karti značajki nulom. Svrha je uvođenje nelinearnosti u proces učenja konvolucijske neuronske mreže. Nakon konvolucije slijedi udruživanje (eng. pooling) čiji je cilj smanjenje dimenzionalnosti svake karte značajki uz zadržavanje najvažnijih informacija. Postoje različite vrste, a najčešće korišteno je maksimalno udruživanje i udruživanje srednjom vrijednošću. Maksimalno udruživanje odabire najveću vrijednost piksela na dijelu slike koji se trenutno analizira, udruživanje srednjom vrijednošću odabire srednju vrijednost, kao što je to prikazano na slici 10.



Slika 10: Vrste udruživanja [11]

Zatim slijedi izravnavanje (eng. flattening), prikazano na slici 11, proces pretvaranja svih rezultirajućih 2-dimenzionalnih nizova u jedan dugi kontinuirani linearni vektor.

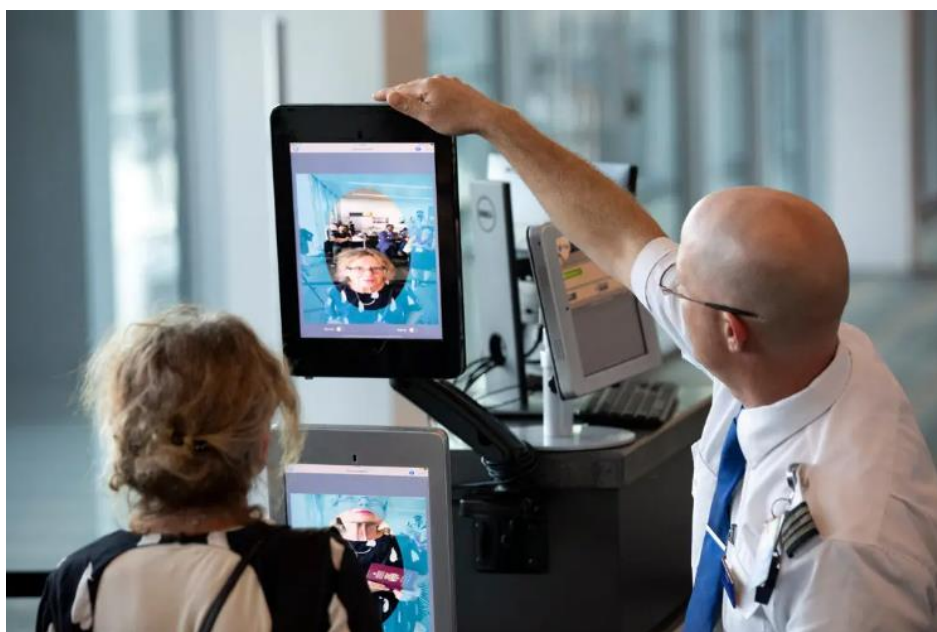


Slika 11: Izravnavanje [10]

Na kraju slijedi potpuna veza. Potpuno povezani sloj je višeslojni perceptron koji koristi aktivacijsku funkciju u izlaznom sloju, a sirove izlaze neuronske mreže pretvara u vektor vjerojatnosti pojedine klase [10].

## 6. METODE ZA DETEKCIJU I IDENTIFIKACIJU LICA

Detekcija i identifikacija lica predstavljaju ključne komponente u suvremenim sustavima za prepoznavanje lica koji imaju široku primjenu kao što su sigurnost i nadzor, analiza slika i videa, vizualizacija produkata, autonomna vozila, biometrijska autentifikacija... Prepoznavanje lica u sigurnosne i nadzorne svrhe jedna je od najvažnijih primjena modernih tehnologija. Umjetne neuronske mreže imaju ključnu ulogu u razvoju, omogućavaju računalima da automatski uče i prepoznaju složene obrasce i karakteristike lica, što rezultira visokom preciznošću i efikasnosti, čak i u složenijim okruženjima i postavkama. Korištenjem neuronskih mreža suvremeni sustavi mogu analizirati velike količine podataka, identificirajući specifične karakteristike lica koje se zatim uspoređuju sa poznatim podacima u bazama. Razvoj i primjena ovih tehnologija značajno su unaprijedili sigurnost, ali se istovremeno postavljaju pitanja o privatnosti i etičkim aspektima. Razvoj naprednih algoritama i metoda je najviše napredovao u zadnjih nekoliko godina zbog brzih i moćnih grafičkih i procesorskih komponenti. Jedan od primjera primjene je prepoznavanje lica u zračnim lukama, kao što to prikazuje slika 12.



*Slika 12: Prepoznavanje lica u zračnoj luci [12]*



## 6.1. Tehnike prepoznavanja lica

Postoje različite tehnike za prepoznavanje lica, a neke od najznačajnijih su: tradicionalna tehnika prepoznavanja lica, prepoznavanje lica na daljinu, trodimenzionalno prepoznavanje i prepoznavanje pomoću toplinske kamere.

Tradicionalna tehnika je najjednostavniji način prepoznavanja lica koji traži odgovarajuće značajke čijom se obradom i analizom zaključuje o pojedincu iz baze podataka o kojem se radi.

Jedan od najčešćih problema koji se javljaju kod sustava nadzornih kamera su problemi niske rezolucije slika. Da bi se prepoznavanje lica na daljinu (engl. HID) omogućila, potrebno je slike niske rezolucije izoštriti koristeći različite filtere. To je tehnika koja omogućava prepoznavanje lica na velikim udaljenostima koja koristi napredne algoritme i tehnologije visoke rezolucije. Koriste se kamere s visokom rezolucijom koje omogućavaju snimanje detaljnih slika, što je ključno za identifikaciju lica. Također se koriste algoritmi za povećanje jasnoće i oštine slike, smanjenje šuma te korekciju izobličenja izazvanih udaljenošću ili osvjetljenjem. [15]

Trodimenzionalno prepoznavanje koristi 3D senzore za uzimanje dubinskih informacija za kreiranje trodimenzionalnog modela lica, slika 13. Jedna od prednosti ove tehnike je otpornost na promjene osvjetljenja u usporedbi sa ostalim tradicionalnim 2D tehnikama. Također može prepoznati lice iz različitih kutova gledanja, uključujući i prikaz profila. Koriste se 3D skeneri koji koriste infracrvenu svjetlost za mapiranje i kreiranje 3D modela lica. Ovi skeneri mjere dubinu i konture lica s visokom razinom detalja [15].



Slika 13: 3D model ljudskog lica [15]

Drugačijim oblik uzimanja ulaznih podataka za prepoznavanje lica je upotrebom toplinske kamere koja koristi infracrvenu svjetlost za snimanje toplinskog zračenja koje emitira ljudsko



tijelo. Ova tehnika se koristi u specifičnim uvjetima gdje konvencionalne kamere ne bi bile efikasne, na primjer u uvjetima tamnog osvjetljenja ili prepoznavanje lica koje je u potpunosti prekriveno odjećom, snima samo oblik i konturu lica, bez detalja. Svaki dio tijela zrači različitu količinu topline, a infracrvene kamere to registriraju i prikazuju kao sliku u nijansama, slika 14. [15]



Slika 14: Prepoznavanje lica toplinskom kamerom [15]

## 6.2. Face ID

Face ID je napredni sustav za prepoznavanje lica koji je razvila tvrtka Apple, a prvi put je predstavljen na iPhone X 2017. godine. Ovo je tehnologija koja omogućava korisnicima otključavanje uređaja, pristupanje aplikacijama, autentifikaciju tijekom plaćanja, pristupanje osjetljivim podacima... Način rada Face ID-a može se podijeliti na nekoliko koraka. Prvi korak je skeniranje lica pomoću TrueDepth kamere, slika 15, koja se sastoji od više senzora i projektor, kada korisnik gleda svoj uređaj kamera aktivira infracrveni projektor koji projektira mrežu od preko 30000 nevidljivih infracrvenih točaka na lice korisnika, infracrvenu kameru koja snima refleksiju točaka sa lica korisnika i osvjetljivač (engl. flood illuminator) koji emitira infracrvenu svjetlost kako bi omogućio funkcionalnost kod slabih osvjetljenja ili u potpunom mraku. Drugi korak je stvaranje 3D mape lica na osnovu reflektiranih infracrvenih točaka, ta mapa se zatim pretvara u matematički model koji predstavlja jedinstveni set karakteristika lica. Zadnji korak je analiza uzorka, Apple koristi specijalno razvijene neuronske mreže za obradu i

analizu 3D mape lica koje omogućavaju prepoznavanje lica u različitim varijacijama, poput različitih izraza lica, brade, promjene frizure, nošenja kapa ili naočala... [13] Najveće kritike su se javile tijekom pandemije COVID-a kada je nošenje maski bilo obavezno, što je stvorilo probleme u prepoznavanju lica jer maska prekriva ključne dijelove lica poput nosa i usta. Kasnije je Apple uveo opciju koja omogućava prepoznavanje lica i sa maskom. Ova funkcionalnost koristi samo gornji dio lica, posebno područje oko očiju, za autentifikaciju.



Slika 15: Face ID [14]

## 7. PREGLED KORIŠTENIH TEHNOLOGIJA

U ovom poglavlju nalazi se prikaz tehnologija korištenih za detekciju i prepoznavanje lica pojedinaca.

### 7.1. Python programski jezik

Implementacija sustava za detekciju i prepoznavanje lica napravljena je u Python-u. Python je objektno orijentirani programski jezik visoke razine, poznat po svojoj jednostavnosti, čitljivosti i fleksibilnosti. Rad s bibliotekama je ključni aspekt korištenja ovog programskog jezika jer omogućava proširenje funkcionalnosti osnovnog jezika i olakšava rad na kompleksnim problemima. Python dolazi sa bogatom standardnom bibliotekom, ali omogućava instaliranje i korištenje dodatnih eksternih biblioteka. Korištene biblioteke za izvedbu sustava su:

- numpy – Puni naziv je Numerical Python, jedna od najvažnijih biblioteka za numeričke proračune. Pruža podršku za velike multidimenzionalne nizove i matrice, omogućava efikasno rukovanje i obradu velikih numeričkih podataka. Biblioteka je uvela vektorizaciju i mogućnost rada sa više-dimenzionalnim podacima što je omogućilo značajno ubrzanje operacija. Glavna struktura podataka u numpy-u je ndarray (n-dimenzionalni niz) koji je optimiziran za rad s homogenim podacima. [16]
- cv2 – Puni naziv biblioteke je engl. Open Source Computer Vision Library. Jedna je od najpoznatijih i najmoćnijih biblioteka za rad sa slikama i videom u području računalnog vida. Omogućava rad na slikama i videima, uključujući prepoznavanje lica, detekciju pokreta, pisanje i čitanje. U području računalnog vida, omogućava detekciju i prepoznavanje lica pomoću Haar kaskade te omogućava praćenje objekata. [17]
- tensorflow – Biblioteka otvorenog koda za strojno učenje razvijena od Google-a [18]. Dizajnirana je kako bi omogućila jednostavnu izradu vlastitih modela strojnog i dubokog učenja. Također omogućava implementaciju složenih modela i nadogradnju istih. U ovoj biblioteci, tenzori su osnovne strukture podataka, mogu biti jednodimenzionalni (vektori) ili višedimenzionalni (matrice). Tensorflow koristi grafove, gdje se čvorovi u grafu odnose na matematičke funkcije, a rubovi predstavljaju

tok podataka između tih operacija. Tensorflow je primjenjiv u različitim područjima i industrijama zbog svoje fleksibilnosti i snage za rad sa složenim modelima. Koristi se u računalnom vidu, za prepoznavanje slika, objekata i lica, obradi prirodnog jezika, prediktivnoj analitici...

## 7.2. Google Colab

Google Colaboratory, poznatiji kao Google Colab je besplatno online okruženje za programiranje bez potrebe za bilo kakvim instalacijama na računalo [19]. Jedna od najbitnijih prednosti je pružanje besplatnog pristupa snažnim grafičkim i procesorskim jedinicama (GPU-ovima i TPU-ovima) koji su idealni za treniranje složenih modela. Sve operacije se izvode na serverima u oblaku, što znači da korisnici ne moraju imati moćno računalo za rad na složenim zadacima. Prvenstveno je dizajnirano za rad s Python-om te ima već unaprijed instalirane pojedine biblioteke za područje računalnog vida poput TensorFlow-a i Pytorch-a, ali podržava i druge jezike.

## 8. PROGRAM ZA IDENTIFIKACIJU LICA

U prethodnim poglavljima su objašnjeni najbitniji pojmovi, poput računalnog vida, neuronskih mreža i biblioteka u Python-u, za razvijanje vlastitog sustava za prepoznavanje lica pojedinaca u realnom vremenu. U ovome poglavlju će biti objašnjen kompletna postupak izrade te rezultati testa na realnim osobama.

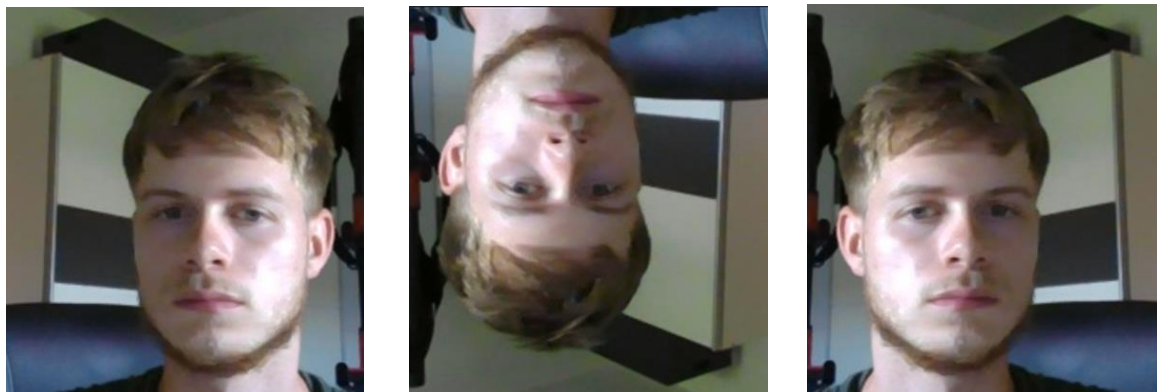
Program se dijeli na dva glavna dijela. Prvi dio je trening modela mreže da identificira lica koje je potrebno, a drugi dio je prilagođavanje i optimiziranje modela da se u realnom vremenu preko web kamere može identificirati pojedinac.

### 8.1. Ulazni skup podataka

Ulazni podatci predstavljaju temelj svakog modela za prepoznavanje lica u realnom vremenu. Za najpreciznije rezultate, najjednostavniji pristup prikupljanja ulaznih podataka podrazumijeva korištenje slika na kojima je lice prikazano u različitim položajima i udaljenostima te različitom kutu u odnosu na kameru, ali kao i uvijek postoje različite prepreke koje sprečavaju ovakav pristup uzimanja ulaznih podataka. Naime, za postizanje visoke točnosti potrebno je imati veliki broj slika svake osobe kako bi mreža mogla pouzdano razlikovati pojedinca. Kod manjih skupova osoba, ovaj pristup je izvediv, međutim, cilj je stvoriti robustan i fleksibilan sustav te baza podataka s fotografijama svih osoba postaje prevelika. Kako bi se izbjegao problem velike količine slika, potrebno je razviti drugačiji pristup koji omogućuje učinkovito treniranje sustava, a da pritom proces prikupljanja podataka bude što jednostavniji. U nastavku će biti prikazana dva konkretna načina na koja se ulazni podatci mogu prikupljati, kako bi se iz svake slike izvuklo što više informacija.

Prvi način uključuje modifikaciju postojećeg skupa slika primjenom posebnih operacija koje mijenjaju njihov izvorni izgled. Ova tehnika naziva se umjetno povećanje ulaznih podataka (engl. dana augmentation), a u kontekstu prepoznavanja lica obuhvaća transformacije kao što su horizontalna i vertikalna refleksija, rotacija i zumiranje i slično [20]. Korištenjem ove metode, jedna slika može generirati nekoliko varijanti koje zadržavaju ključne značajke, dok istovremeno uvode raznolikost u ulazni skup, što uvelike poboljšava preciznost neuronske

mreže. Primjenom ovog pristupa smanjuje se potreba za velikim brojem različitih slika jer se ulazni podatci za svaku osobu mogu proširiti umjetno generiranim dodatnim slikama.



*Slika 16: Umjetno povećanje ulaznih podataka*

Kao primjer, na slici 16 izvorna slika je prikazana lijevo, dok su ostale slike generirane ovom tehniku, prva slika je nastala rotacijom, druga zrcaljenjem. Ovaj postupak može značajno povećati raznolikost ulaznih podataka, ali je i važno uzeti u obzir da će usporiti proces učenja neuronske mreže. Jedan od glavnih nedostataka ove metode je rizik od deformiranja slika različitih dimenzija kada se prilagode potrebnim veličinama za neuronsku mrežu, što može rezultirati nerealnim prikazom ljudskog lica. Iako je ovaj pristup jednostavan, u ovome programu neće biti korišten isključivo on. Najveći nedostatak je što i dalje zahtijeva slike osobe koju treba prepoznati prije nego što program počne raditi.

U drugoj se metodi web kamera koristi kao izvor ulaznih podataka. Kada se program pokrene, aktivira se kamera koja se koristi i za prepoznavanje lica u realnom vremenu. Kamera snima video, a svakih nekoliko sličica (engl. frame) automatski pohranjuje slike u datoteku s ulaznim podatcima. Nakon prikupljanja dovoljnog broja slika, kamera se isključuje. Ovim pristupom je omogućeno prikupljanje slika željene osobe bez potrebe za unaprijed pripremljenim slikama. Važno je da se osoba tijekom snimanja lagano pomiče kako bi se osigurala raznovrsnost ulaznih podataka.

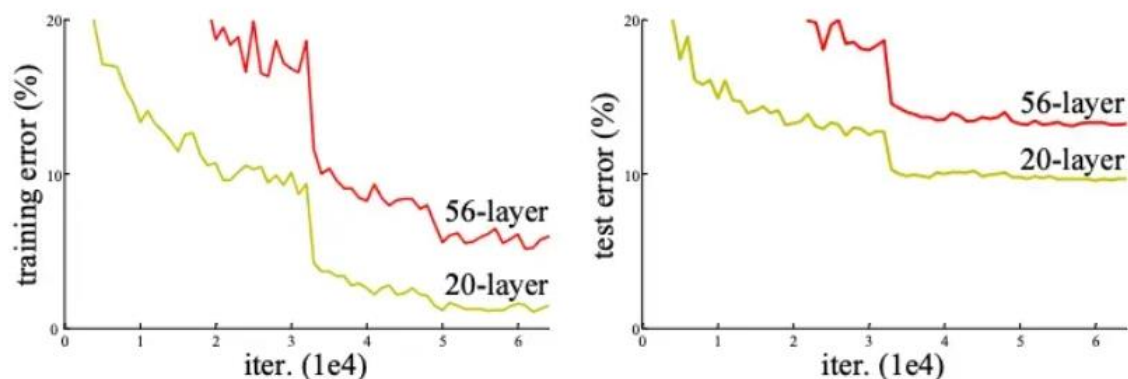
Kombiniranjem prednosti oba pristupa, moguće je izraditi pouzdan sustav s velikim brojem različitih slika svake osobe koje treba prepoznati. U ovom programu, koristit će se obje metode, slike prikupljene pomoću web kamere umjetno će se modificirati kako bi se povećala

raznolikost ulaznog skupa. Osim toga, da bi se spriječilo snimanje mutnih slika ili slika bez prisutnosti osobe u kadru, slike će se uzimati samo kada program detektira prisutnost lica. Ovaj dio programa detaljnije je opisan u poglavlju 8.3. Važno je razlikovati pojmove detekcije i identifikacije osobe jer detekcija i identifikacija mogu funkcionirati neovisno. Detekcija označava prepoznavanje lica osobe na slici, a identifikacija dodjeljuje prepoznatom licu odgovarajuću oznaku iz baze podataka.

## 8.2. Parametri neuronske mreže

Za problem kao što je prepoznavanje i identifikacija lica potrebno je koristiti složenu neuronsku mrežu poput one opisane u poglavlju 6.5. . Ove napredne mreže sastoje se od mnogobrojnih slojeva čija bi ručna konfiguracija bila izuzetno izazovna. Potrebno je pažljivo povezati sve slojeve kako bi mreža ispravno funkcionirala. Potrebno je i detaljno ispitati i optimizirati funkcije svakog sloja, da bi dobili što točnije rezultate.

Zbog toga se češće koriste unaprijed razvijeni modeli mreža za pojedini problem, kao što je prepoznavanje i identifikacija lica. U prvoj fazi ovog sustava, trenira se neuronska mreža za prepoznavanje osoba u realnom vremenu, pri čemu je korišten ResNet50, unaprijed strukturirana mreža prilagođena prepoznavanju objekata na slikama. ResNet50 pripada obitelji ResNet, niz modela dizajniranih za rješavanje izazova povezanih s treniranjem dubokih neuronskih mreža. Primarni problem koji je ResNet riješio bio je problem degradacije u dubokim neuronskim mrežama izazvanim poteškoćama u optimizaciji treninga. Kako mreže postaju dublje, njihova točnost se zasićuje, a zatim i brzo razgrađuje [23]. Slika 17 prikazuje kako dublja mreža, u ovom slučaju s 56 sloja, ima veću pogrešku treninga, a zatim i testa, od mreže s 20 slojeva. U ovom slučaju, ResNet50 se sastoji od 48 konvolucijskih slojeva, jednog Max Pool i jednog Average Pool sloja.



Slika 17: Pogreške mreže s 56 i 20 slojeva [23]

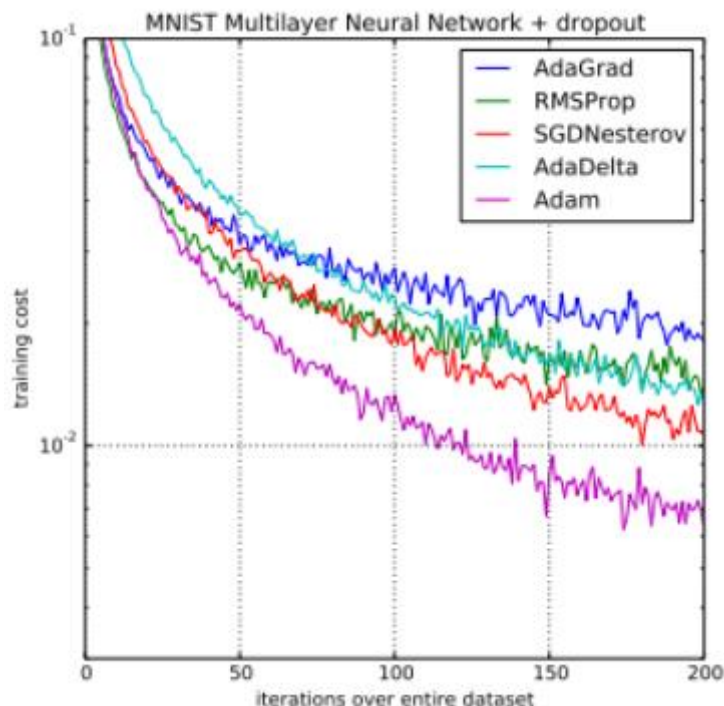
Ovaj model koristi rezidualno učenje neuronskih mreža. Mreže koje se oslanjaju na rezidualno učenje se ne fokusiraju isključivo na učenje specifičnih značajki sa slike, već nastoje naučiti neke rezidualne vrijednosti, odnosno rezidue. Rezidualne vrijednosti se mogu prikazati kao oduzimanje značajki naučenih na ulazu u sloj [24]. To se postiže korištenjem tzv. preskočnih veza (engl. skip connections) koje povezuju slojeve u mreži s nekoliko slojeva udaljenim slojevima. Ovakav pristup omogućava jednostavnije i učinkovitije učenje u usporedbi s tradicionalnim konvolucijskim mrežama.

ResNet50 model neuronske mreže nije samo definiran svojom složenom strukturom, već je i njegova primjena od izuzetne važnosti. Kako bi se izgradio takav kompleksan i pouzdan model, nužno je koristiti odgovarajuću bazu podataka. U ovom slučaju, baza podataka koja će se koristiti za treniranje modela naziva se ImageNet. ImageNet je baza slika organiziranih prema WordNet hijerarhiji, fokusirajući se isključivo na imenice, pri čemu je svaki čvor hijerarhije predstavljen stotinama ili čak tisućama slika [25]. WordNet je opsežna baza engleskih riječi, gdje su imenice opisane na način da se sve, kroz hijerarhijski sustav, na kraju povežu s temeljnim čvorom. Dakle, sve imenice su instance ili sinonimi neke krovne imenice. U ImageNet bazi podataka sve imenice su opisane slikama uz odgovarajuće oznake koje pojašnjavaju što svaka slika predstavlja. Ova baza podataka sadrži više od 14 milijuna označenih slika. Za potrebe ovog programa, koristit će se dio baze koji se sastoji od 1000 različitih objektnih klasa te 1,2 milijuna slika za treniranje, 50 000 za validaciju te 100 000 za testiranje[25].



Kada je u potpunosti definiran model koji će se koristiti za prepoznavanje lica pojedinca, potrebno ga je optimizirati i prilagoditi specifičnim zahtjevima. To se najjednostavnije postiže „zamrzavanjem“ dijela težina u modelu, dodavanjem dodatnih izlaznih slojeva koji su specifično dizajnirani za prepoznavanje lica i treniranjem modela koristeći ulazne podatke koji su opisani u poglavlju 9.1. Prvi korak u ovoj prilagodbi je „zamrzavanje“ težina, što znači da se njihove vrijednosti neće mijenjati tijekom treniranja mreže jer ne želimo mijenjati sve težine u već istreniranom modelu, već samo one koje su potrebne za prilagodbu mreže našim specifičnim potrebama. Nakon što se model učita i „zamrznu“ odabrane težine, dodaju se dva dodatna sloja na kraj mreže. Predzadnji sloj biti će Global Average Pooling sloj, a posljedni sloj biti će prilagođen broju osoba koje želimo prepoznati na slici. Ovaj sloj imati će onoliko neurona koliko je osoba u setu podataka, a njegova struktura naziva se potpuno povezani sloj.

Dodatne modifikacije implementiranog modela obuhvaćaju nekoliko važnih elemenata, poput automatskog spremanja modela tijekom procesa treniranja. To omogućava da se model pohranjuje svakih nekoliko koraka, čime se osigurava da proces učenja ne mora započeti ispočetka ako dođe do neočekivanog prekida ili problema. U TensorFlow biblioteci, ova funkcionalnost se ostvaruje korištenjem ModelCheckpoint objekta. Također je potrebno odrediti tehniku optimizacije mreže, odnosno algoritam koji će prilagođavati težine tijekom procesa učenja. Za ovaj zadatak koristit će se Adam optimizacijski algoritam, puni naziv Adaptive Movement Estimation. Adam je napredni optimizacijski algoritam koji se koristi umjesto tradicionalnog stohastičkog gradijentnog spuštavanja kako bi se iterativno mijenjale težine tijekom učenja [26]. Tradicionalni algoritmi za gradijentno spuštavanje računaju ciljnu funkciju s obzirom na vrijednost ulaznih varijabli i mijenjaju težine na način da se pogreška smanjuje u smjeru suprotnom od izračunatog gradijenta. Prilagođena verzija gradijentnog spuštavanja, u kojoj se gradijent računa za nasumično odabranu točku, umjesto za cijeli skup podataka, naziva se stohastički gradijentni spust. Rezultati dobiveni korištenjem Adam algoritma pokazuju njegovu visoku točnost i brzu konvergenciju prema globalnom maksimumu ciljne funkcije. Na slici 18, prikazana je usporedba Adam algoritma s drugim poznatim optimizacijskim algoritmima.



Slika 18: Usporedba algoritama za optimizaciju [26]

Jedna od najčešće korištenih funkcija prema kojoj se izračunava pogreška na izlazu mreže (funkcija cilja) je suma najmanjih kvadrata, međutim, kod problema klasifikacije, češće se koristi Cross-Entropy koju je potrebno minimizirati. Cross-Entropy mjeri razliku između dviju distribucija vjerojatnosti za dane varijable, a definira se sljedećim izrazom:

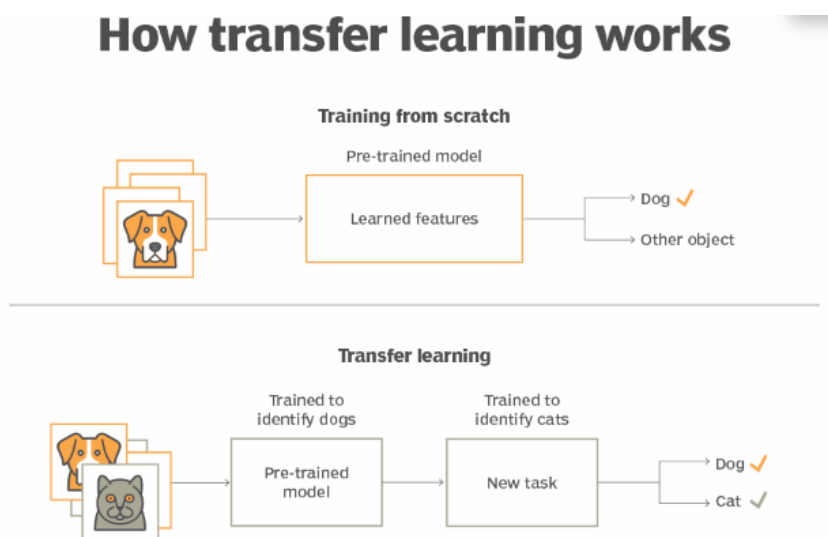
$$\mathbf{H} = - \sum p(x) \log p(x) \quad (27)$$

gdje je  $H$  vrijednost funkcije cilja, a  $p(x)$  vjerojatnost događaja  $x$  u distribuciji  $p$ . Za minimizaciju ove funkcije koristit će se Adam optimizacijski algoritam s postavljenom stopom učenja od 0,01.

Kao mjera brzine učenja ResNet50 neuronske mreže, koristit će se vrijednost pogreške prilikom validacije modela. Skup podataka za validaciju čini oko 15% ukupne baze podataka koja se koristi za treniranje mreže, to znači da će se 15% slika svake osobe koristiti za validaciju mreže tijekom treninga. Vrijednost pogreške prilikom validacije trebala bi padati kako mreža napreduje kroz učenje. Također je definiran prestanak faze učenja kada se vrijednost pogreške pri validaciji nije značajno promijenila 3 uzastopne epohe, što ukazuje da pogreška stagnira i nije potrebno daljnje treniranje. U TensorFlow biblioteci, ova funkcionalnost ostvaruje se

korištenjem funkcije EarlyStopping, čiji su parametri podešeni prema opisanom kriteriju za zaustavljanje treniranja neuronske mreže.

Dakle, ovakav način korištenja već kreiranog modela i prilagodbu istog, naziva se prenešeno učenje (engl. Transfer Learning). Ova tehnika postaje sve važnija u području dubokih neuronskih mreža, posebno kada je dostupno ograničeno vrijeme ili resursi. Prenešeno učenje omogućava korištenje unaprijed treniranih modela, čime se značajno smanjuje vrijeme treniranja. Većina današnjih sustava zasniva se na ovom principu, gdje se može poslužiti nekim javno dostupnim sustavom kojeg je potrebno samo prilagoditi za specifičan zadatak i problem.



Slika 19: Prenešeno učenje [28]

Slika 19 jasno ilustrira korištenje preneseno učenje kao učinkovitiju metodu u usporedbi s kreiranjem modela iz nule. Preneseno učenje se ostvaruje „zamrzavanjem“ dijela težina postojeće mreže, na koju se dodaju novi slojevi prilagođeni za prepoznavanje osoba na temelju novih lica.

Možemo se osvrnuti na različite parametre koji se koriste prilikom treniranja prilagođenog ResNet50 modela. Težine se mogu mijenjati nakon obrade svakog ulaznog podatka ili nakon obrade određenog broja podataka. U ovom programu, težine u posljednja dva sloja mreže se

mijenjaju nakon što kroz mrežu prođe 16 slika. Ovakav način treniranja poznat je kao treniranje mini-setom podataka (engl. mini-batch), a hiperparametar koji definira broj podataka u mini-setu naziva se veličina seta (engl. batch size). Standardizirane vrijednosti su 16, 32, 64 ili 128 [29].

Također je potrebno definirati i koliko puta cijeli podataka mora proći kroz mrežu da bi se postigla željena točnost modela. Ovaj hiperparametar se naziva broj epoha. Epoha označava jedan potpuni prolazak kompletnog skupa ulaznih podataka kroz mrežu. Obično je potrebno nekoliko epoha kako bi mreža postigla odgovarajuću točnost. U ovom programu, broj epoha je postavljen na 50. U praksi, često nije potrebno proći kroz svih 50 epoha jer se vrijednosti pogreške pri validaciji brzo konvergiraju prema minimumu, što može dovesti do ranog prekida procesa učenja. Ako model ne postiže željene rezultate, moguće je povećati traženu točnost ili smanjiti dopuštenu pogrešku.

Nakon definiranja svih hiperparametara, proces treniranja mreže može započeti. Trajanje treniranja ovisi o broju osoba koje treba prepoznati. Model bi mogao prepoznati veliki broj osoba, ali povećanjem broja osoba povećava se i složenost sustava i broj parametara mreže. Time se povećava mogućnost da se željena točnost ne postigne kroz sve epohe treniranja. U ovom programu, broj osoba za prepoznavanje je 3. Nakon završetka treninga, model mreže se sprema u obliku h5 datoteke, datoteke u hijerarhijskom formatu koja sadrži višedimenzionalne tenzore sa spremljenim podacima. Na ovaj način se zapisuju vrijednosti svih težina u cijeloj mreži.

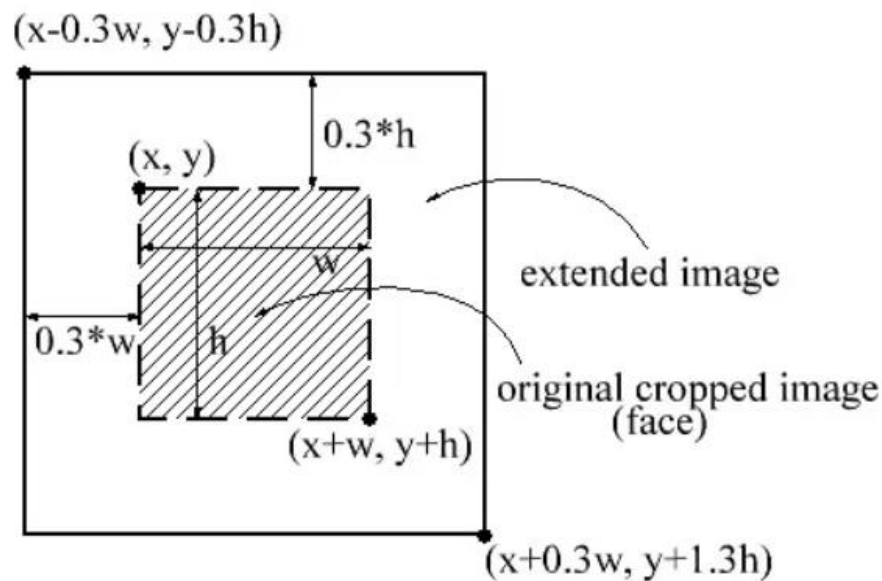
### **8.3. Prepoznavanje lica s kamerom u stvarnom vremenu**

Za detekciju i prepoznavanje lica u stvarnom vremenu, bitna biblioteka za operacije na slikama i videozapisima je OpenCV, koja se lako integrira u Python. Kao što je već napomenuto u poglavlju 8.1., ova biblioteka se koristi za strojno učenje i računalni vid jer sadrži preko 2500 optimiziranih algoritama iz područja računalnog vida koji se mogu koristiti u svakakvim problemima, kao što su prepoznavanje lica, prepoznavanje i praćenje objekata i drugim.

Otvoreni pristup kodu (engl. Open Source) rezultiralo je širokom primjenom ove biblioteke u komercijalne i industrijske svrhe.

OpenCV biblioteka nudi mogućnost korištenja već definiranog algoritma za detekciju lica. U okviru ove biblioteke, tja algoritam poznat je kao *Cascade Classifier* i služi za detekciju objekata na temelju značajki prisutnih na slici [30]. Algoritam se temelji na tzv. *Haar* značajkama koje predstavljaju sume pojedinih piksela u određenim područjima slike. Funkcioniraju na sličan način kao kreiranje mape značajki kod konvolucijskog sloja neuronskih mreža. Za sustav detekcije lica, sve te značajke moraju se izračunati na svim slikama tijekom faze treniranja algoritma, a ako zadovolje određeni prag, algoritam zaključuje da se na tom području slike nalazi lice. Zatim slijedi odabir onih značajki koje su mogle najpreciznije odrediti nalazi li se na slici lice ili ne, u ovom radu ih ima oko 6000. Međutim, problem je u tome što ovakav postupak zahtijeva veliki broj računskih operacija, pa se uvodi koncept kaskadnih klasifikacija. Taj koncept najprije identificira potencijalna mjesta gdje bi lice moglo biti, a tek onda provjerava nalazi li se lice na tom mjestu. To se postiže organiziranjem značajki u slojeve te oni izračunavaju jedan po jedan na svakom dijelu slike, ako neki od ranijih slojeva ne zadovolji određeni prag, odbacuje se mogućnost da se na tom području slike nalazi lice te se prelazi na slijedeće područje. Na taj način se značajno smanjuje broj potrebnih matematičkih operacija. U ovom radu, algoritam od 6000 značajki je podijeljen u 38 slojeva, kaskada. Kada algoritam prepozna lice na slici, rezultat će biti pravokutnik koji omeđuje to lice. Konkretno, algoritam generira 4 parametra,  $x$  koordinata gornjeg lijevog kuta,  $y$  koordinata gornjeg lijevog kuta te širina i visina pravokutnika, koji jednoznačno definiraju položaj i dimenzije pravokutnika.

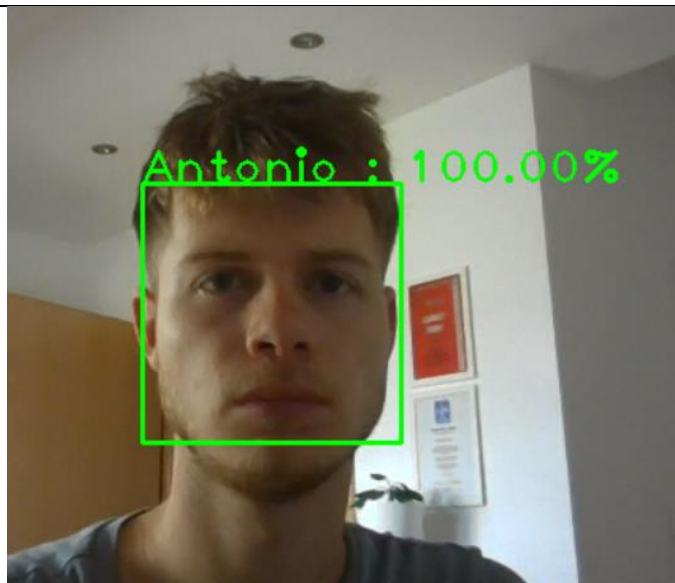
Prije nego se model za prepoznavanje lica integrira u algoritam za detekciju lica, potrebno je izvršiti određene promjene na slikama. Unutar pravokutnika kojeg daje algoritam, nalazi se samo lice bez okolnih elemenata, što se razlikuje od slika koje su korištene za treniranje modela neuronske mreže za prepoznavanje pojedinaca. Zbog toga je potrebno proširiti sliku, kao što to prikazuje slika 20, kako bi model mogao točno prepoznati pojedinca. To je postignuto korištenjem funkcije koja koristi parametre pravokutnika koji omeđuje lice, izvornu sliku iz web kamere i parametar  $k$  koji označava faktor proširenja slike. Faktor  $k$  se prilagođava ovisno o ulaznim podacima sa kojima je trenirana ResNet50 mreža.



Slika 20: Proširivanje slike [31]

Sada je moguće implementirati neuronsku mrežu za prepoznavanje lica u algoritam za detekciju lica u stvarnom vremenu. Kako bi sustav mogao učinkovito obrađivati slike, ulazne slike su konvertirane iz RGB formata u *Greyscale* format te su dimenzije slika također smanjene na 250 x 250 piksela.

Model za prepoznavanje lica putem web kamere daje dva ključna podatka nakon što se lice detektira i obradi kroz neuronsku mrežu. Prvi podatak je vektor vrijednosti izlaznog sloja neurona koji predstavlja koliko je mreža sigurna da to detektirano lice pripada određenoj osobi. Te vrijednosti su u rasponu od 0 do 1, što se može interpretirati kao postotak sličnosti detektiranog lica sa svakim licem iz skupa podataka na kojem je mreža trenirana. Drugi podatak je ime osobe za koju je mreža procijenila najveću vjerojatnost da se podudara s detektiranim licem. Na osnovu ova dva podatka, rezultat prepoznavanja se prikazuje tako da se oko lica, omeđenog pravokutnikom, prikaže ime prepoznate osobe te postotak sigurnosti mreže u to prepoznavanje, kao što je to prikazano na slici 21.



*Slika 21: Primjer rezultata*

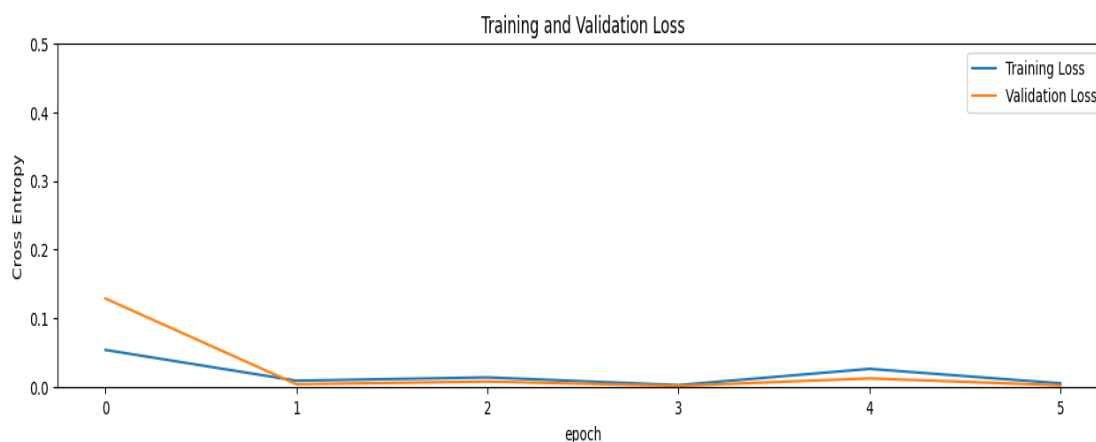
Dodatno, postavljen je prag sigurnosti koji mora biti zadovoljen kako bi program zaista potvrdio da je prepoznao pravu osobu. Ako taj prag nije zadovoljen, prikazuje se samo crveni pravokutnik oko detektiranog lica, bez imena i postotka sigurnosti. Ovo je korisno kada mreža nije sigurna ili kada se na slici pojavi osoba koja nije bila obuhvaćena ulaznim skupom podataka za treniranje. U takvim slučajevima, mreža može dati rezultate koje ne želimo smatrati ispravnima. Stoga je prag sigurnosti postavljen na 0,8 što odgovara sigurnosti od 80% da je osoba na slici prepoznata ispravno.

## 9. REZULTATI

U ovom radu, zadatak je bio testirati program na stvarnim osobama i provjeriti njegovu učinkovitost ovisno o kutu gledanja, odnosno kutovima pod kojima je lice okrenuto u odnosu na kameru. Testiranje učinkovitosti provodit će se na tri osobe.

Program je testiran na tri osobe: Andrea, Antonio i Mia. Slike za treniranje neuronske mreže, dobivene su snimanjem fotografija web kamerom i raspodjelom na dva skupa, za treniranje i za testiranje mreže. Nakon toga, slike su umjetno modificirane na način kao što je opisan u poglavlju 9.1., čime se dobiva konačan broj slika. Andrein skup podataka sadrži oko 10 000 slika, Antonijev oko 30 000 te Mijin oko 5 000 slika.

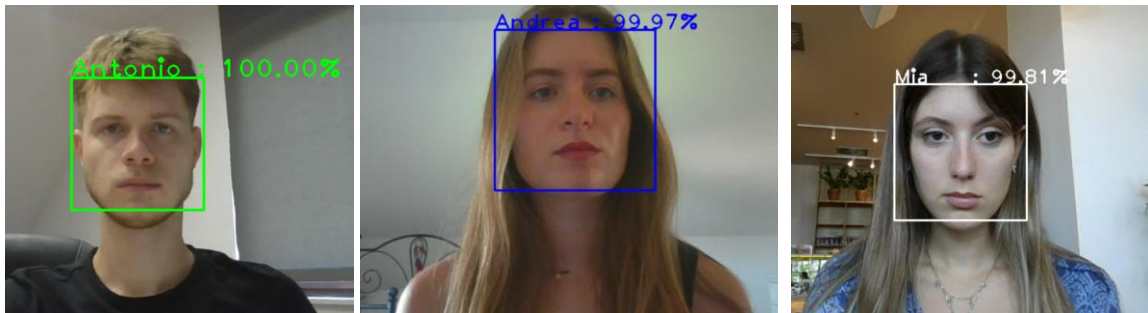
Slika 22 prikazuje graf gubitka tokom procesa treniranja neuronske mreže. Na početku treniranja (epoha 0), gubitak je relativno visok i za treniranje i za validaciju, ali već poslije prve epohe, gubitak značajno opada. Nakon prve epohe, gubitak ostaje nizak i stabilan tokom ostatka epohe. Male oscilacije nakon prve epohe su gotovo zanemarive, što sugerira da se model dobro trenirao te ukazuje na dobru sposobnost generalizacije modela.



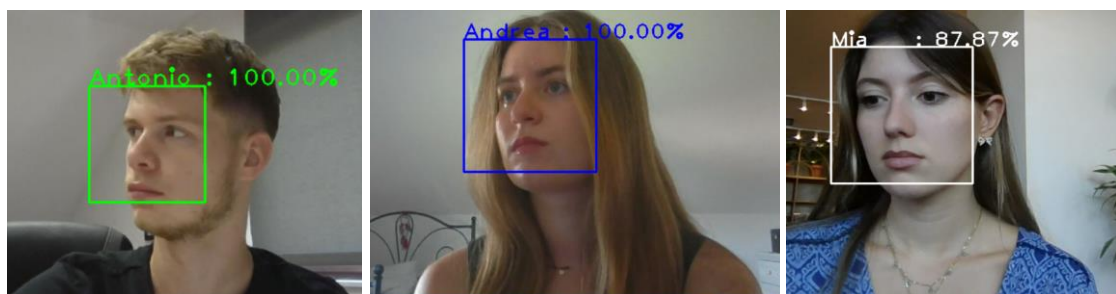
Slika 22: Proces treniranja mreže



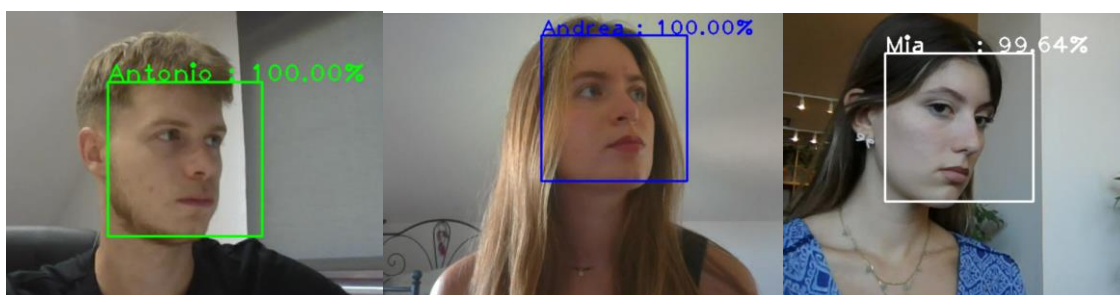
Nakon što smo završili trening, možemo procijeniti sigurnost mreže u odnosu na različite kuteve pod kojima je lice okrenuto prema kameri. Ti kutevi definiraju zakretanje glave u desno i lijevo, pomicanje glave prema gore ili dolje te naginjanje glave u lijevo ili desno. Slika 23 prikazuje lice kako mreža prepoznaje lice u različitim kutevima, što nam omogućuje da procijenimo njenu pouzdanost u realnim situacijama.



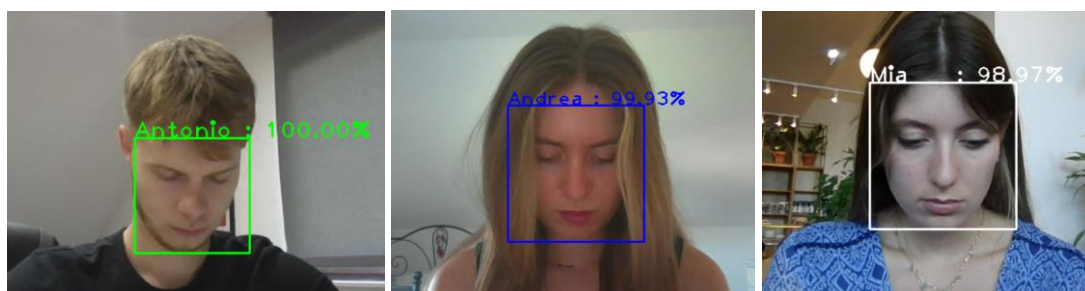
Slika 23: Normalan pogled



Slika 24: Zakret u desno



Slika 25: Zakret u lijevo

*Slika 26: Zakret prema gore**Slika 27: Zakret prema dolje*

Slike 23-27 prikazuju vrlo visoke postotke sigurnosti modela za svaki pogled. Važno je naglasiti da ukoliko osoba više zakrene glavu, program ne bi prepoznao lice sa slike. Možemo vidjeti da ovaj program zapravo ne obuhvaća širok raspon položaja glave, ali kada lice bude detektirano, sustav ga vrlo precizno identificira. Na temelju toga može se zaključiti da je ograničenje programa zapravo povezano s mogućnostima sustava za detekciju lica. Također, možemo vidjeti da sustav najbolje prepoznaje Antonia, odnosno mene, pretpostavka je zbog toga što u mom ulaznom skupu ima najviše slika.

Dosadašnje analize fokusirale su se na prepoznavanje lica jedne osobe, ali je važno uzeti u obzir i situaciju kada je na slici više lica. Algoritam iz OpenCV biblioteke može detektirati više lica na jednoj slici, algoritam skenira svaki dio slike i na osnovu izračunatih karakteristika određuje je li prisutno lice na tom dijelu. Kada je na slici više osoba, izlaz algoritma više nije samo jedan vektor redak, već matrica u kojoj svaki redak sadrži podatke o položaju lica. Dakle, prisustvo većeg broja lica na slici ne predstavlja problem za ovaj algoritam, ali će procesiranje zahtijevati više vremena. Ostali dio programa također je nezavisan o broju lica jer se svako detektirano lice individualno obrađuje kroz neuronsku mrežu. Slika koja ulazi u neuronsku mrežu predstavlja samo dio sličice na kojoj se nalazi lice. Stoga, prisustvo većeg broja osoba na slici

---

neće smanjiti preciznost prepoznavanja pojedinačnih lica niti će utjecati na sigurnost neuronske mreže u njenu klasifikaciju.

---

## 10. ZAKLJUČAK

U ovom radu razvijen je sustav za prepoznavanje lica koristeći tehnike strojnog i dubokog učenja s posebnim naglaskom na sigurnosne aplikacije i biometrijsku autentifikaciju. Sustav je uspješno implementiran i testiran na stvarnim podacima, što je omogućilo evaluaciju njegove brzine i točnosti. Rezultati pokazuju da je model sposoban prepoznati lica u stvarnom vremenu s visokim stupnjem točnosti, što potvrđuje učinkovitost korištenih metoda.

Iako su rezultati vrlo zadovoljavajući, uvijek postoji prostor za daljnje unaprjeđenje, poput korištenja većih i raznovrsnijih skupova podataka za treniranje te optimizacija modela za bržu izvedbu. Također trenutni model ima poteškoća s prepoznavanjem lica koja su jako zakrenuta. Jedno od mogućih rješenja je korištenje trodimenzionalnih modela lica, koji bi mogli poboljšati otpornost na ove varijacije. Još jedan izazov leži u etičkim pitanjima vezanim uz privatnost i sigurnost, jer tehnologije za prepoznavanje lica mogu biti zloupotrijebljene. Iako sustavi poput ovoga nude značajne prednosti u sigurnosnim primjenama, važno je osigurati jasne regulacije kako bi se spriječile potencijalne povrede privatnosti i zloupotrebe.

Ovaj rad predstavlja solidnu osnovu za buduća istraživanja i razvoj sustava za prepoznavanje lica koji bi se mogli koristiti u različitim industrijama.

Treba istaknuti da, iako računalni vid nikada neće biti 100% pouzdan, kontinuirani napredak u tehnologijama omogućava sve precizniju i bržu identifikaciju i prepoznavanje lica, čime se značajno poboljšava kvaliteta.

## LITERATURA

- [1] Što je umjetna inteligencija: <https://www.umjetnaintelijai.com/sto-je-umjetna-inteligencija/> (Pristupljeno: 3.8.2024.)
- [2] Tomislav Stipančić , predavanje „Metode i alazi umjetne inteligencije“ (Pristupljeno: 3.8.2024.)
- [3] cadcam-group, Što su umjetna inteligencija i strojno učenje i zašto su važni?: <https://www.cadcam-group.eu/hr/knowledge/sto-su-umjetna-inteligencija-i-strojno-ucenje-i-zasto-su-vazni/> (Pristupljeno: 3.8.2024.)
- [4] Milan Hrga, Računalni vid: <https://hrcak.srce.hr/file/292457> (Pristupljeno: 4.8.2024.)
- [5] SAS, Computer Vision: What it is and why it matters: [Computer Vision: What it is and why it matters | SAS](#) (Pristupljeno: 4.8.2024.)
- [6] Ž. Ujević Andrijić, Umjetne neuronske mreže: <https://hrcak.srce.hr/file/322233> (Pristupljeno: 5.8.2024.)
- [7] Guillod, T., Papamanolis, P., & W. Kolar, J. (2020). Artificial Neural Network (ANN) Based Fast and Accurate Inductor Modeling and Design. *IEEE Open Journal of Power Electronics*, 1, 284-299 : [https://www.semanticscholar.org/paper/Artificial-Neural-Network-\(ANN\)-Based-Fast-and-and-Guillod-Papamanolis/67ec2c7be314bcde334ed3ababdd81565d9c70ca](https://www.semanticscholar.org/paper/Artificial-Neural-Network-(ANN)-Based-Fast-and-and-Guillod-Papamanolis/67ec2c7be314bcde334ed3ababdd81565d9c70ca) (Pristupljeno: 5.8.2024.)
- [8] Medium, How do we 'train' neural networks? : <https://towardsdatascience.com/how-do-we-train-neural-networks-edd985562b73> (Pristupljeno: 6.8.2024.)
- [9] builtin, How Loss Functions Work in Neural Networks and Deep Learning: <https://builtin.com/machine-learning/loss-functions#:~:text=A%20loss%20function%20measures%20how,make%20the%20neural%20network%20better> (Pristupljeno: 6.8.2024.)
- [10] Dragutin Lisjak i Davor Kolar, predavanje „Konvolucijske neuronske mreže“ (Pristupljeno: 6.8.2024.)
- [11] Medium, Convolutional Neural Networks: <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53> (Pristupljeno: 6.8.2024.)

- [12 ] The New York Times, What You Need to Know About Facial Recognition at Airports: <https://www.nytimes.com/2022/02/26/travel/facial-recognition-airports-customs.html> (Pristupljeno: 7.8.2024.)
- [13] Face ID, [https://en.wikipedia.org/wiki/Face\\_ID](https://en.wikipedia.org/wiki/Face_ID) (Pristupljeno: 7.8.2024.)
- [14] Vox, How can we ban facial recognition when it's already everywhere? <https://www.vox.com/recode/2020/7/3/21307873/facial-recognition-ban-law-enforcement-apple-google-facebook> (Pristupljeno: 7.8.2024.)
- [15 ] Facial recognition system: [https://en.wikipedia.org/wiki/Facial\\_recognition\\_system](https://en.wikipedia.org/wiki/Facial_recognition_system) (Pristupljeno: 7.8.2024.)
- [16] Real Python, What is NumPy?: <https://realpython.com/tutorials/numpy/> (Pristupljeno: 8.2024.)
- [17] GeeksforGeeks, What is OpenCV Library?: <https://www.geeksforgeeks.org/opencv-overview/> ,7.8.2024.
- [18] GeeksforGeeks, Introduction to TensorFlow: <https://www.geeksforgeeks.org/introduction-to-tensorflow/> (Pristupljeno: 7.8.2024.)
- [19] Google colab, Google Colaboratory: <https://colab.google/> (Pristupljeno: 7.8.2024.)
- [20] aws, What is Data Augmentation?: [https://aws.amazon.com/what-is/data-augmentation/#:~:text=Data%20augmentation%20is%20the%20process,machine%20learning%20\(ML\)%20models](https://aws.amazon.com/what-is/data-augmentation/#:~:text=Data%20augmentation%20is%20the%20process,machine%20learning%20(ML)%20models) (Pristupljeno: 8.8.2024.)
- [21] Neural network: [https://en.wikipedia.org/wiki/Neural\\_network\\_\(biology\)](https://en.wikipedia.org/wiki/Neural_network_(biology)) (Pristupljeno: 8.8.2024.)
- [22] Neuron: <https://hr.wikipedia.org/wiki/Neuron> (Pristupljeno: 8.8.2024.)
- [23] roboflow, What is ResNet-50?: <https://blog.roboflow.com/what-is-resnet-50/#:~:text=ResNet%2D50%20is%20a%20convolutional,it%2C%20and%20categorize%20them%20accordingly> (Pristupljeno: 8.8.2024.)
- [24] OpenGenus IQ, Residual Network (ResNet): <https://iq.opengenus.org/resnet/> (Pristupljeno: 8.8.2024.)

- 
- [25] paperswithcode, ImageNet: <https://paperswithcode.com/dataset/imagenet>  
(Pristupljeno: 8.2024.)
- [26] Machine Learning Mastery, Gentle Introduction to the Adam Optimization Algorithm for Deep Learning: <https://machinelearningmastery.com/adam-optimization-algorithm-for-deep-learning/> (Pristupljeno: 9.8.2024.)
- [27] datacamp, Cross-Entropy Loss Function in Machine Learning: Enhancing Model Accuracy: [https://www.datacamp.com/tutorial/the-cross-entropy-loss-function-in-machine-learning?dc\\_referrer=https%3A%2F%2Fwww.google.com%2F](https://www.datacamp.com/tutorial/the-cross-entropy-loss-function-in-machine-learning?dc_referrer=https%3A%2F%2Fwww.google.com%2F)  
(Pristupljeno: 9.8.2024.)
- [28] TechTarget, What is transfer learning?:  
<https://www.techtarget.com/searchcio/definition/transfer-learning> (Pristupljeno: 9.8.2024.)
- [29] kaggle, Full batch, mini-batch and online learning:  
<https://www.kaggle.com/code/residentmario/full-batch-mini-batch-and-online-learning>  
(Pristupljeno: 9.8.2024.)
- [30] Medium, Haar Cascades, Explained: <https://medium.com/analytics-vidhya/haar-cascades-explained-38210e57970d> (Pristupljeno: 17.8.2024.)
- [31] Medium, How to Create a Real-Time Face Detector:  
<https://towardsdatascience.com/how-to-create-real-time-face-detector-ff0e1f81925f>  
(Pristupljeno: 17.8.2024.)

---

# PRILOG

**Kod:** [https://github.com/Toni030702/Zavrсни\\_rad](https://github.com/Toni030702/Zavrсни_rad)