

Sustav za identifikaciju osoba temeljen na računalnom vidu i dubokom učenju

Španović, Mihael

Undergraduate thesis / Završni rad

2022

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:200:411927>

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

Download date / Datum preuzimanja: **2024-11-26**

Repository / Repozitorij:

[Faculty of Electrical Engineering, Computer Science and Information Technology Osijek](#)



SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I
INFORMACIJSKIH TEHNOLOGIJA OSIJEK

Stručni studij

**Sustav za identifikaciju osoba temeljen na računalnom vidu
i dubokom učenju**

Završni rad

Mihael Španović

Osijek, 2022.

SADRŽAJ

1. UVOD	1
1.1. Zadatak završnog rada	2
2. DEFINIRANJE PROBLEMA PREPOZNAVANJA LICA I PREGLED POSTOJEĆIH METODA	3
2.1. Prepoznavanje lica iz slike	3
2.2. Pregled osnovnih metoda za detekciju i prepoznavanje lica	4
2.2.1. Pregled osnovnih metoda detekcije lica	4
2.2.2. Pregled osnovnih metoda prepoznavanja lica	4
2.3. Pregled postojećih programskih rješenja za prepoznavanje lica	5
2.4. Izazovi u prepoznavanju lica	6
2.4.1. Varijacija u pozi	6
2.4.2. Varijacija u osvjetljenju	6
2.4.3. Varijacija u izrazu lica.....	6
2.4.4. Starenje.....	6
2.4.5. Okluzije lica	7
2.4.6. Sličnost određenih lica	7
2.4.7. Rezolucija slike lica	7
3. PREDLOŽENI SUSTAV ZA IDENTIFIKACIJU OSOBE.....	8
3.1. Detekcija lica	9
3.2. Registracija osobe.....	10
3.3. Prepoznavanje lica	12
3.4. Baza podataka.....	12
3.4.1. Pohrana detekcije	14
3.5. Administratorsko sučelje	14
4. EVALUACIJA SUSTAVA	17
4.1. Evaluacija podsustava za detekciju lica	17
4.2. Evaluacija podsustava za prepoznavanje lica	19
4.3. Evaluacija sustava na RaspberryPi platformi.....	21
5. ZAKLJUČAK	22
LITERATURA.....	23
SAŽETAK	24
ABSTRACT.....	25
PRILOZI	26

1. UVOD

Visokotehnoški sustavi kontrole pristupa objektu ili prostoriji sve su popularniji i pristupačniji generalnoj javnosti i organizacijama. Takvi sustavi najčešće se koriste u svrhu zaštite imovine ili ograničavanja pristupa neovlaštenim osobama. Kroz većinu ljudske povijesti najpouzdaniji način kontrole pristupa bilo je korištenje ključa ili u slučaju nešto modernijih rješenja, korištenje pina. Međutim razvojem na tržište dolaze sustavi koji umjesto tradicionalnih metoda koriste biometriju.

U modernom svijetu sve više prevladavaju biometrijske metode identifikacije koje koriste individualne fizičke karakteristike pojedinaca. Ovom metodom osoba se prvo prepoznaje, nakon čega se odobrava pristup. Tehnologije prepoznavanja osobe, specifično prepoznavanja lica postale su sve popularnije metode utvrđivanja identiteta osobe u svrhu kontrole pristupa ili identifikacije osoba koje su obavljale neovlaštenu radnju. Prepoznavanje lica je kategorija biometrijske sigurnosti. Ostali oblici biometrijskog softvera uključuju prepoznavanje glasa, prepoznavanje otiska prsta i prepoznavanje mrežnice ili šarenice oka.

Postupak prepoznavanja lica zahtijeva uređaj koji ima digitalnu fotografsku tehnologiju za generiranje i dobivanje slika i podataka potrebnih za stvaranje i snimanje biometrijskog uzorka lica osobe koju kasnije treba identificirati. Sustavi za prepoznavanje lica mogu se koristiti za prepoznavanje ljudi na fotografijama, videozapisima ili u stvarnom vremenu na temelju signala s kamere. Sustav analizira ulaznu sliku i provjerava odgovaraju li biometrijski podaci osobi koja mora koristiti uslugu ili koja traži pristup aplikaciji, sustavu ili čak zgradi. Karakteristike ulazne slike ovise o karakteristikama kamere čiji senzor može biti RGB, termalni, dubinski i slično.

Zbog napretka u području umjetne inteligencije, strojnog učenja, a posebno dubokog učenja, sustavi za prepoznavanje lica brzo se razvijaju te mogu raditi s najvišim standardima sigurnosti i pouzdanosti. Kod sustava za prepoznavanje koriste se tehnike strojnog učenja za detekciju, prikupljanje, pohranjivanje i procjenu karakteristika lica kako bi se mogle usporediti sa slikama osoba u bazi podataka. Zahvaljujući integraciji ovih tehnika i rastu procesorske snage, proces prepoznavanja se može provoditi u stvarnom vremenu.

Unutar ovoga rada predstavljen je sustav za prepoznavanje lica temeljen na dubokom učenju i računalnom vidu. Sustav funkcionira na način da analizira dolazne slike sa kamere u potrazi za licem. Detektirano lice se onda u procesu prepoznavanja uspoređuje sa pohranjenim licima iz baze podataka. Informacije o pojedinoj detekciji se pohranjuju kako bi mogle biti korištene u svrhu kontrole pristupa ili pregledane od strane ovlaštene osobe.

U drugom poglavlju dane su osnovne informacije o načinima izvođena detekcije i prepoznavanja lica te mogući izazovi prilikom njihove implementacije. Sustav za identifikaciju osobe predstavljen je u trećem poglavlju unutar kojeg se nalaze detaljni opisi svakog od podsustava. Rezultati evaluacije i opis iste predstavljeni su u četvrtom poglavlju. Na kraju je dan zaključak rada.

1.1. Zadatak završnog rada

Zadatak ovog završnog rada je izrada sustava za kontrolu pristupa temeljenog na prepoznavanju lica. Za proces prepoznavanja lica koristi se računalni vid i duboko učenje. Podatci potrebni za duboko učenje dobivaju se iz dokumenata registriranih osoba te nekoliko njihovih slika koje su pohranjene u bazi podataka. Izrađeni sustav potrebno je evaluirati na odgovarajući način.

2. DEFINIRANJE PROBLEMA PREPOZNAVANJA LICA I PREGLED POSTOJEĆIH METODA

U ovom poglavlju predstavljen je opis procesa prepoznavanja lica, pregled osnovnih metoda za detekciju i prepoznavanje lica te su navedeni izazovi koji mogu utjecati na rezultate prepoznavanja.

2.1. Prepoznavanje lica iz slike

Proces prepoznavanja lica (eng. *face recognition*) započinje sa pronalaskom lica unutar dane slike što se naziva detekcija lica (eng. *face detection*). Aplikacije za detekciju lica koriste algoritme i strojno učenje (eng. *machine learning* - ML) za pronalaženje ljudskih lica unutar slika, koje često uključuju druge objekte bez lica kao što su krajolici, zgrade i drugi dijelovi ljudskog tijela poput stopala ili ruku. Metode za detekciju lica, specifično *feature invariant methods* obično započinju traženjem ljudskih očiju – što je jedna od najlakših značajki za otkrivanje. Nakon toga bi se pokušale otkriti obrve, usta, nos, nosnice i šarenicu. Treniranje modela za detekciju lica (u slučaju npr. metoda temeljenih na izgledu) poboljšava sposobnost modela da utvrdi postoji li lice na slici i gdje se ono nalazi. Kako bi se osigurala točnost, modeli strojnog učenja se moraju trenirati na velikim skupovima podataka koji uključuju stotine tisuća slika lica. Posljednje godine donijele su napredak u otkrivanju lica korištenjem dubokog učenja koje ima prednost u odnosu na tradicionalne metode ostvarene putem računalnog vida.

Nakon inicijalne detekcije lica potrebno je obaviti segmentaciju lica (izdvajanje lica iz slike) te ekstrakciju poznatih obilježja lica. Ekstrakcija obilježja lica je proces izdvajanja značajki komponenti lica kao što su oči, nos, usta, itd. iz slike ljudskog lica. Ekstrakcija crta lica vrlo je važna za inicijalizaciju tehnika obrade kao što su praćenje lica, prepoznavanje izraza lica ili prepoznavanje lica.

Prepoznavanje lica na slikama može biti komplicirano zbog varijabilnosti čimbenika kao što su poza, izraz, položaj, orijentacija, boja kože i vrijednost piksela, prisutnost naočala ili dlačica na licu, te razlike u rezoluciji kamere i uvjetima osvjetljenja. Informacije o snimljenom licu uspoređuju se s bazom podataka drugih obilježja lica. Broj lica koji se uspoređuje ovisi o bazi podataka i broju baza podataka kojima sustav ima pristup. Tehnologija prepoznavanja lica tada identificira podudaranje obilježja lica te vraća informacije o korisniku s pronađenim podudaranjem i drugim relevantnim informacijama, kao što su adresa i ime [1].

2.2. Pregled osnovnih metoda za detekciju i prepoznavanje lica

2.2.1. Pregled osnovnih metoda detekcije lica

- Metode temeljene na znanju ili na pravilima (eng. *Knowledge-based, or rule-based methods*) opisuju lice putem predefiniranih pravila. Izazov ovog pristupa je teškoća kreiranja dobro definiranih pravila.
- Metode stalnih značajki (eng. *Feature invariant methods* - koje koriste značajke kao što su oči ili nos osobe za otkrivanje lica. Svijetlost i šum mogu negativno utjecati na detekciju.
- Metode uparivanja predložaka (eng. *Template matching*) temelje se na usporedbi slika sa standardnim uzorcima lica ili značajkama koje su prethodno pohranjene i povezivanju njih kako bi se otkrilo lice. Nažalost, ove metode imaju problema ukoliko postoji varijacija u pozici, veličini i obliku lica.
- Metode temeljene na izgledu (eng. *Appearance-based methods*) koriste statističku analizu i strojno učenje kako bi pronašli relevantne karakteristike lica [2].

2.2.2. Pregled osnovnih metoda prepoznavanja lica

- *Holistic Matching* metoda kao ulaz za svoj *face catching system* uzima cijelo područje lica. Najbolji primjeri holističkih metoda su *Eigenfaces*, PCA, linearna diskriminantna analiza i analiza nezavisnih komponenti, itd.
- Metode temeljene na obilježjima lica (eng. *Feature-based*), ovdje se prije svega izdvajaju lokalna obilježja kao što su oči, nos, usta i njihova lokacija, koja se onda unose u strukturni klasifikator. Izazov za metode izdvajanja značajki je "obnavljanje" značajki, to je kada sustav pokušava dohvatiti značajke koje su nevidljive zbog velikih varijacija, npr. varijacija u pozici glave u usporedbi sa slikom profila.
- Metode temeljene na modelu (eng. *Model-based*) pokušavaju modelirati lice. Novi uzorak se uvodi u model, a parametri modela se koriste za prepoznavanje slike. Dijelimo ju na 2D i 3D pristup.
- Hibridne metode (eng. *Hybrid-methods*) koriste kombinaciju holističkih metoda i metoda izdvajanja značajki. Općenito se u ovim metodama koriste 3D slike. Slika lica je uhvaćena u 3D, kako bi se zabilježile krivulje očnih duplji, oblici brade i/ili čela. Čak bi i lice uhvaćeno u profilu poslužilo jer sustav koristi dubinu i mjernu os, što mu daje dovoljno informacija za konstruiranje cijelog lica. 3D sustav uključuje detekciju, položaj, mjerenje, predstavljanje i uparivanje [3].

U okviru ovog rada koristi se metoda za detekciju lica koja se temelji na metodama stalnih značajki dok se za prepoznavanje lica koristi metode temeljene na obilježjima lica iz područja dubokog učenja.

2.3. Pregled postojećih programskih rješenja za prepoznavanje lica

Tvrtke koje žele integrirati tehnologiju prepoznavanja lica sa svojim vlastitim proizvodima i aplikacijama mogu se odlučiti za programska rješenja temeljena na web-uslugama. Sustavi za prepoznavanje lica (engl. *Facial recognition systems* - FRS) obično nisu dostupni kao samostalno programsko rješenje, već dolaze kao dio usluge.

Od dostupnih rješenja ističe se Amazon Rekognition čije su osnovne značajke:

- Analiza lica i pretraživanje lica koriste se za provjeru korisnika, brojanje ljudi i slučajeve upotrebe u javnoj sigurnosti,
- Mogućnost davanja oznaka prepoznatim ljudima,
- Amazon ima prednost što ima na raspolaganju ogroman skup podataka što osigurava točnost,
- Rekognition omogućuje korisnicima da dodaju prilagođene oznake objektima i scenama kako bi odgovarale poslovnim potrebama.
- Amazon pruža moderiranje sadržaja. Sadrži testirani skup podataka koji koristi za označavanje neprikladnog sadržaja,
- Omogućuje i pretraživanje lica i provjeru unutar privatnog spremišta fotografija, ali u ovom slučaju nastaju troškovi skladištenja,
- Rekognition pruža prepoznavanje i analizu lica u videozapisima - uživo ili pohranjenim.
- Uz sve prijašnje navedene stavke Rekognition je također moguće koristiti bez ikakvog znanja o strojnom učenju [4].

Od besplatnih/otvorenog koda (eng. *open source*) rješenja ističe se *face_recognition* biblioteka. Ovo je vjerojatno najpopularnija besplatna biblioteka za prepoznavanje lica, jer trenutno ima 37,6 tisuća zvjezdica na GitHubu. Postoje dvije mogućnosti za korištenje; može se koristiti njihovo Python aplikacijsko programsko sučelje (engl. *application programming interface* - API) ili alat za konzolne naredbe. Postoje upute za instalaciju za sve glavne platforme, pa čak i *docker images* za brzo postavljanje. Iako postoje dobri razlozi za njegovu popularnost, postoje ozbiljni nedostaci koji se moraju uzeti u obzir prije korištenja. Za početak, zadnje izdanje bilo je 2018. (četiri godine u svijetu strojnog učenja je dugo). Drugo, koristi prilično zastarjeli model prepoznavanja lica koji je 99,38% točan u prepoznavanju lica unutar *Labeled Faces in the Wild* (LFW) skupa slika, ali mu

točnost značajno opada prilikom prepoznavanja lica u novijim skupovima, Konačno, nije ga tako lako integrirati jer nema REST API [5]. *Face_recognition* biblioteka bazira se na *dlib*-u. *Dlib* je C++ alat koji sadrži algoritme za strojno učenje i alate za stvaranje složenih programskih rješenja koji se primjenjuju unutar akademskih zajednica i industrije.

Za detekciju lica ističe se *Mediapipe* biblioteka. *MediaPipe* je Googleov projekt koji nudi *open-source* ML rješenja. Drugim riječima, *MediaPipe* pruža pristup širokom spektru snažnih modela strojnog učenja optimiziranih za rad na različitim platformama bez obzira na njihove performanse.

2.4. Izazovi u prepoznavanju lica

2.4.1. Varijacija u pozi

Varijacije u pozi uzrokuju značajne probleme u prepoznavanju lica. Varijacije poze najčešće su posljedica promjene položaja glave prilikom snimanja. Mnogi sustavi mogu tolerirati male varijacije, ali baza podataka većine sustava kao i skupovi slika za učenje modela obično se sastoje od slika frontalnog pogleda na lice osobe. Budući da su postojeći FRS-i vrlo osjetljivi na varijacije u pozi, korekcija poze je vrlo bitna. Korekcija se može postići učinkovitim tehnikama kojima je cilj rotirati lice i/ili ga poravnati s osi slike.

2.4.2. Varijacija u osvjetljenju

Varijacije u osvjetljenju nastaju prema ukupnom intenzitetu svjetlosti odnosno jačini izvora svjetlosti. Ako postoji prejak ili preslab izvor svjetlosti može doći do nemogućnosti analize lica. Također problem predstavlja nejednaka raspodjela svjetlosti npr. ukoliko je jedna strana lica u sjeni dok je na drugoj jaka svjetlost. Međutim postoje algoritmi koji se koriste za izjednačavanje razina osvjetljenja na licu kako bi riješili ovaj problem.

2.4.3. Varijacija u izrazu lica

Neke varijacije u slikama lica mogu biti uzrokovane zbog razlike u izrazu na koju utječe emocionalno stanje pojedinca. Stoga je važno prepoznati različite izraze lica za procjenu emocionalnog stanja. Ljudski izrazi se sastoje od makro-izraza kao što su gađenje, ljutnja, sreća strah, tuga ili iznenađenje. Kozmetika i frizure također mogu biti uključene u ove varijacije jer promjena frizure i šminkanje također mogu uzrokovati varijacije u izrazu lica.

2.4.4. Starenje

Drugi razlog za promjene u izgledu lica je starenje ljudskog lica koje ima utjecaj na cijeli proces prepoznavanja lica. Ako sustavu nije predana nova slika osobe nakon određenoga vremena postoji

moгуćnost krivog ili neuspješnog prepoznavanja zbog mogućih značajnih promjena u izgledu osobe. Prema različitim studijama koje su proveli znanstvenici, u rasponu od 10 godina događaju se značajne promjene u izgledu pojedinca.

2.4.5. Okluzije lica

Varijacije u izgledu lica također mogu biti uzrokovane zbog prisutnosti objekata na licu. Iako je lice pronađeno, ono može biti teško za prepoznati upravo zbog skrivenih dijelova lica, što otežava izvlačenje značajki. Ovo su situacije kada osoba, koja je u procesu prepoznavanja, razgovara na telefon, nosi kapu ili sunčane naočale.

2.4.6. Sličnost određenih lica

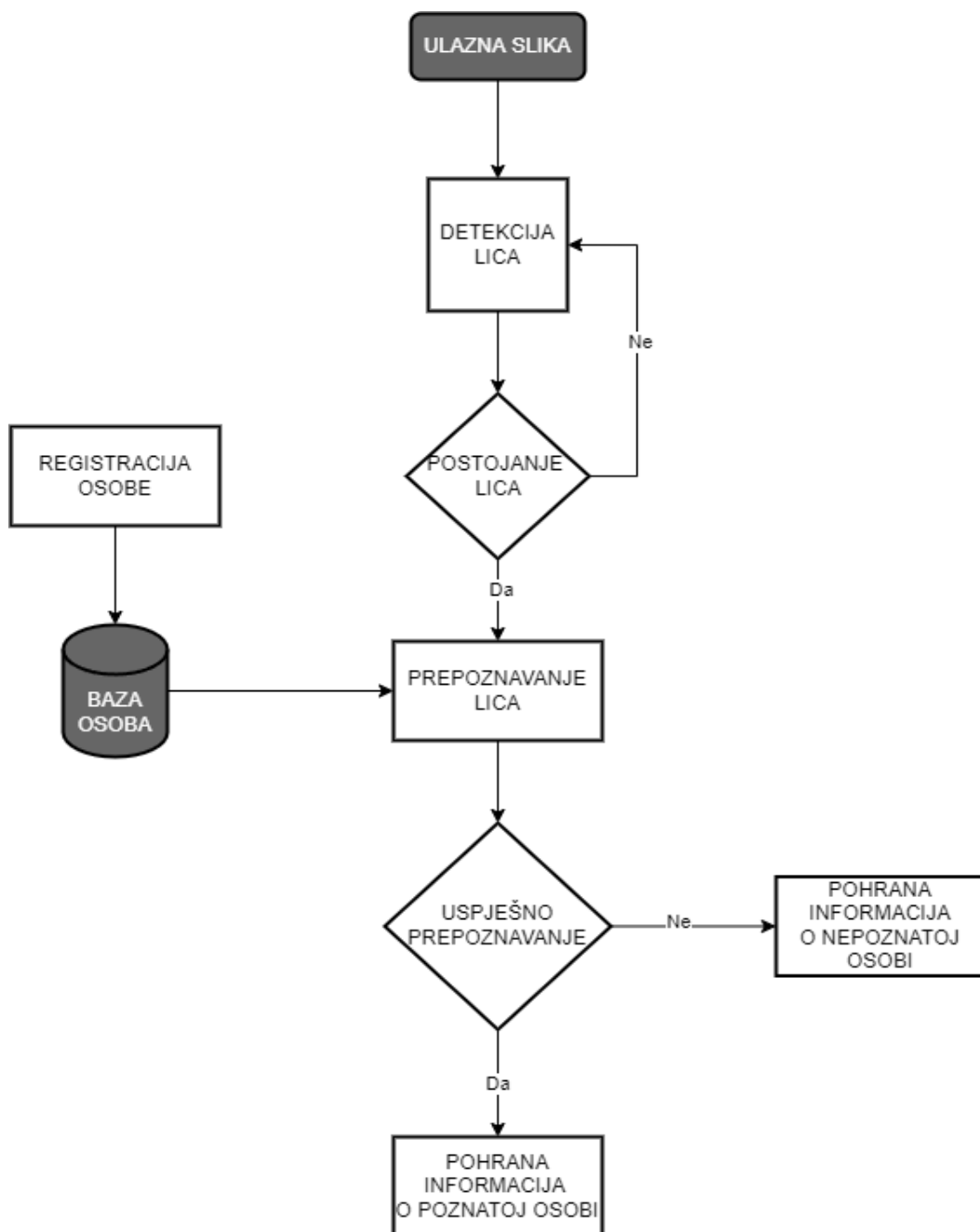
Ovo obično nije tako čest izazov, ali i kod ljudi postoje situacije gdje je teško izvršiti prepoznavanje zbog sličnih lica. Npr. identični blizanci sa sličnim crtama lica, oblikom, itd. postavljaju težak zadatak za sustav prepoznavanja lica. Ovaj problem najčešće uzrokuje povećanje stope lažnog prepoznavanja [6, 7].

2.4.7. Rezolucija slike lica

Smanjena rezolucija slike lica ima utjecaj na detekciju i identifikaciju lica. Prema [8] postoji pad od 14% u efikasnosti prepoznavanja nakon prelaska iz slike rezolucije lica koja iznosi 256*256 piksela u 21*21 piksela. Većina sustava koristi rezoluciju slike lica 128*128 piksela zbog najboljeg omjera efikasnosti i utjecaja na performanse.

3. PREDLOŽENI SUSTAV ZA IDENTIFIKACIJU OSOBE

U ovom poglavlju je predstavljen izrađeni sustav za identifikaciju osobe. Dijagram toka informacija u predloženom sustavu za identifikaciju osobe prikazan je na slici 3.1. Sustav dohvaća sliku s kamere te provodi detekciju lica. Ukoliko postoji lice u dohvaćenoj slici izvodi se proces prepoznavanja koji uspoređuje pohranjene podatke iz baze osoba sa podacima iz slike. Rezultat prepoznavanja se onda pohranjuje u bazu podataka. Registracija osobe provodi se tako da se slika osobe pohrani u bazu zajedno sa drugim osnovnim informacijama kao što je ime osobe.



Slika 3.1. Dijagram toka predloženog sustava za identifikaciju osobe temeljen na računalnom vidu.

Implementacija sustava temelji se na Python programskom jeziku gdje se za poslužiteljski dio koristi *Django framework* te JavaScript (*React framework* i *Axios*) za primjer korisničkog sučelja. Komunikacija između poslužitelja i korisnika odvija se pomoću REST API zahtjeva što također omogućuje integraciju drugih korisničkih sučelja.

3.1. Detekcija lica

Ulaz u podsustav za detekciju lica je digitalna slika u RGB formatu, a izlaz je pozicija lica na slici koja se označava s graničnim pravokutnikom (engl. *bounding box* - BB). Granični pravokutnik definiran je s četiri broja [*xmin*, *ymin*, *width*, *height*] gdje *xmin* i *ymin* predstavljaju koordinate ishodišta BB, a *width* i *height* su širina i visina BB. Detekcija lica je definirana kao BB koji sadrži informacije o položaju lica unutar ulazne slike. Proces detekcije izvodi se dva puta, prvi puta prilikom stvaranja profila te kasnije tijekom procesa prepoznavanja. Detekcija lica izvodi se pomoću Python biblioteke *mediapipe*. *Mediapipe* se temelji na BlazeFace modelu kojeg je razvio Google i koji je vrlo efikasan model za detekciju lica. Prvi korak je inicijalizacija modela iz *mediapipe* biblioteke što zahtijeva odabir tipa modela i postavljanje minimalne razine pouzdanosti detekcije. Razlikujemo dva tipa modela, model 0 je optimiziran za detekciju lica na udaljenosti do 2 metra od kamere, a model 1 za detekciju lica na udaljenosti do 5 metara. Prilikom implementacije podsustava za detekciju odabran je model 1. Odabrani model se ne može mijenjati bez pristupa programskom kodu. Za detekciju lica *mediapipe* koristi ključne točke lica koje pokušava pronaći unutar svakog okvira. Ako postoji lice unutar provjerenog okvira, *mediapipe* funkcija vraća listu koja sadrži BB svih pronađenih lica. Na slikama 3.2. i 3.3. prikazan je primjer ulazne slike i izlaza iz podsustava nakon detekcije lica.



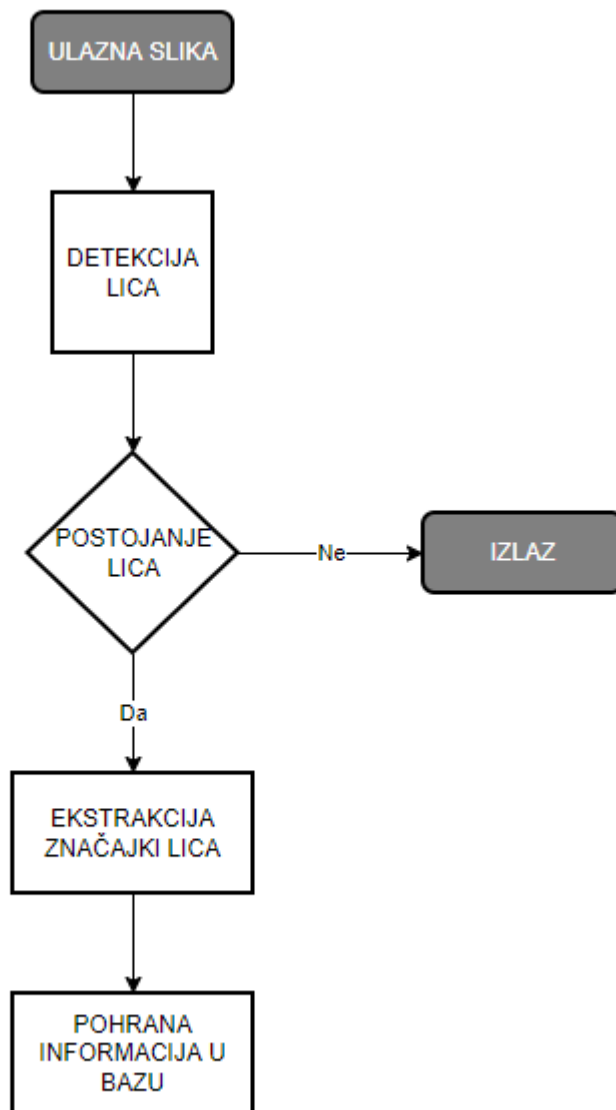
Slika 3.2. Primjer ulazne slike sa označenim BB



Slika 3.3. Primjer slike nakon detekcije

3.2. Registracija osobe

Podsustav za registraciju opisan je slikom 3.4. i služi za stvaranje profila osobe te predstavlja osnovni proces koji omogućuje kasniju radnju prepoznavanja lica. Prilikom registracije korisnika zahtjeva se unos imena i prezimena osobe i potrebno je priložiti dvije slike koje odgovaraju postavljenim zahtjevima. U ovom slučaju jedna slika je slika osobne/vozačke iskaznice, a druga je obični portret osobe. Prije same pohrane potrebno je provjeriti postoji li lice unutar danih slika. Podsustavu detekcije predaju se slike te ukoliko na predanim slikama postoji lice, kako ne bi trajno pohranjivali cijelu sliku nečijih privatnih dokumenata, pohranjuje se samo dio slike koji se nalazi unutar BB. U slučaju da na slikama nije pronađeno lice sustav vraća odgovarajuću poruku. Na pohranjenom BB-u lica radi se ekstrakcija značajki pomoću funkcije iz biblioteke *face_recognition*. Kod ekstrakcije značajki razlikujemo dva modela, „small“ model koristi 5 značajki lica dok „large“ model koristi 68 značajki. Prilikom implementacije odabran je small model zbog boljih performansi. Ekstrakcijom značajki lica korisnika predstavljeno je vektorom od 128 dimenzija. Na kraju se u bazu podataka za određenog korisnika pohranjuje vektor značajki, ime, prezime i dvije prijašnje navedene slike. Zbog postojanja velikog broja potencijalnih problema prilikom registracije kao što je korištenje dijakritičkih znakova u imenu/prezimeni ili postojanje razmaka na kraju istih bilo je potrebno osigurati da se podaci dobiveni putem registracije mogu nesmetano pohraniti u bazu. Kao što je prikazano na slici 3.5. za svaku registriranu osobu kreira se mapa na hard disku čiji se naziv sastoji od imena i prezimena osobe, a u nju se spremaju dvije priložene slike.



Slika 3.4. Dijagram toka podsustava za registraciju



Slika 3.5. Primjer pohrane u slučaju dva korisnika

3.3. Prepoznavanje lica

Ulaz u podsustav za prepoznavanje lica je lista koja sadrži lokacije lica unutar dane slike, slika iz koje se dobivaju značajke lica osobe koju želimo prepoznati te listu koja sadrži pohranjene vektore značajki lica. Prepoznavanje se izvodi koristeći Python biblioteku *face_recognition*. *Face_recognition* biblioteka temelji se na *dlib*-u što je moderna C++ biblioteka koja sadrži algoritme strojnog učenja i alate za stvaranje programa koji koriste strojno učenje. *Dlib* model ostvaruje 99.38% točnosti na LFW *face recognition benchmarku* [9]. Proces prepoznavanja lica svodi se na usporedbu pohranjenih značajki lica i značajki lica iz ulazne slike. Pozivanjem funkcije *recognise_face()* te predajom potrebnih parametara prvo se vrši ekstrakcija značajki lica iz slike u kojem je detektirano lice. Nakon toga se dobiveni vektor značajki uspoređuje sa značajkama lica iz baze podataka koristeći funkciju *.compare_faces()*. Navedena funkcija izračunava euklidsku udaljenost između dva vektora i u slučaju da je euklidska udaljenost za neki od danih vektora značajki ispod definiranog praga, smatra se da se radi o osobi kojoj pripada taj vektor. Vrijednost praga iznosi 0.6 te se ne može mijenjati bez pristupa programskom kodu. Ukoliko nije moguće pronaći značajke koje su unutar postavljenog praga navedeno lice označava se kao nepoznato.

3.4. Baza podataka

Za pohranu podataka služi integrirani sustav baze unutar *Django* biblioteke koji se bazira na *SQLite* bazi podataka. Podaci se pohranjuju u dvije tablice, jedna tablica koja služi za pohranu osnovnih informacija o registriranoj osobi i značajki lica (Tab. 3.1.) i jedna tablica koja služi za pohranu informacija o detekcijama lica (Tab. 3.2.). Osnovne informacije o registriranoj osobi uključuju ime i prezime osobe te dvije slike. Informacije potrebne za pohranu detekcije sastoje se od imena i prezimena prepoznate osobe (ako postoji) i vremena detekcije. Restrikcije za svaku od tablica uključuju maksimalni broj znakova za pojedina polja, nemogućnost uređivanja id-a profila koji je ujedno i primarni ključ i još par opcija vidljivih na slici 3.6. Kao što je navedeno ranije podsustav registracije dobiva 2 slike koje mogu sadržavati kompromitirajuće informacije kao što je broj osobne iskaznice. Kako bi se izbjeglo spremanje takvih informacija u obliku slike, prilikom primanja *http* zahtjeva, slike se prvo spremaju u privremenu mapu te se nad njima odmah provodi postupak detekcije lica nakon kojeg se slike odbacuju. Ovim postupkom sustav dobiva samo nužne informacije koje su potrebne za prepoznavanje lica, a to je dio osobnog dokumenta ili dio slike na kojoj se nalazi ljudsko lice. Vektori značajki prilikom pohrane prolaze kroz proces zvan *pickeling*. *Pickle* se koristi za serijalizaciju Python objekata u niz bajtova koji se onda lako mogu pohraniti u bazu podataka.

Tab. 3.1. Logičko oblikovanje tablice „Profile“

Naziv	Tip podatka	Opis
profile_id	Int	Jedinstveni broj profila
name	Varchar	Ime osobe
surname	Varchar	Prezime osobe
id_image	Image	Slika lica sa dokumenta
selfie_image	Image	Portret slika
encodings	PickleType	Vektor značajki lica

Tab. 3.2. Logičko oblikovanje tablice „Detections“

Naziv	Tip podatka	Opis
detecion_id	Int	Jedinstveni broj profila
name	Varchar	Ime osobe
surname	Varchar	Prezime osobe
time	DateTime	Vrijeme detekcije
unknown	Boolean	Uspješna identifikacija

```
class Profile(models.Model):
    profile_id=models.AutoField(primary_key=True,editable=False)
    name= models.CharField(max_length = 200)
    surname= models.CharField(max_length = 200)
    id_image= models.ImageField(default = '/No_image.png')
```



```

selfie_image= models.ImageField(default = '/No_image.png')

encodings = PickledObjectField(null=True,blank=True)

class Detection(models.Model):

    detection_id=models.AutoField(primary_key=True,editable=False)

    name= models.CharField(max_length = 200)

    surname= models.CharField(max_length = 200)

    time = models.DateTimeField(default=timezone.now(),null = False)

    unknown = models.BooleanField(default=True)

```

Slika 3.6. Fizičko oblikovanje baze podataka

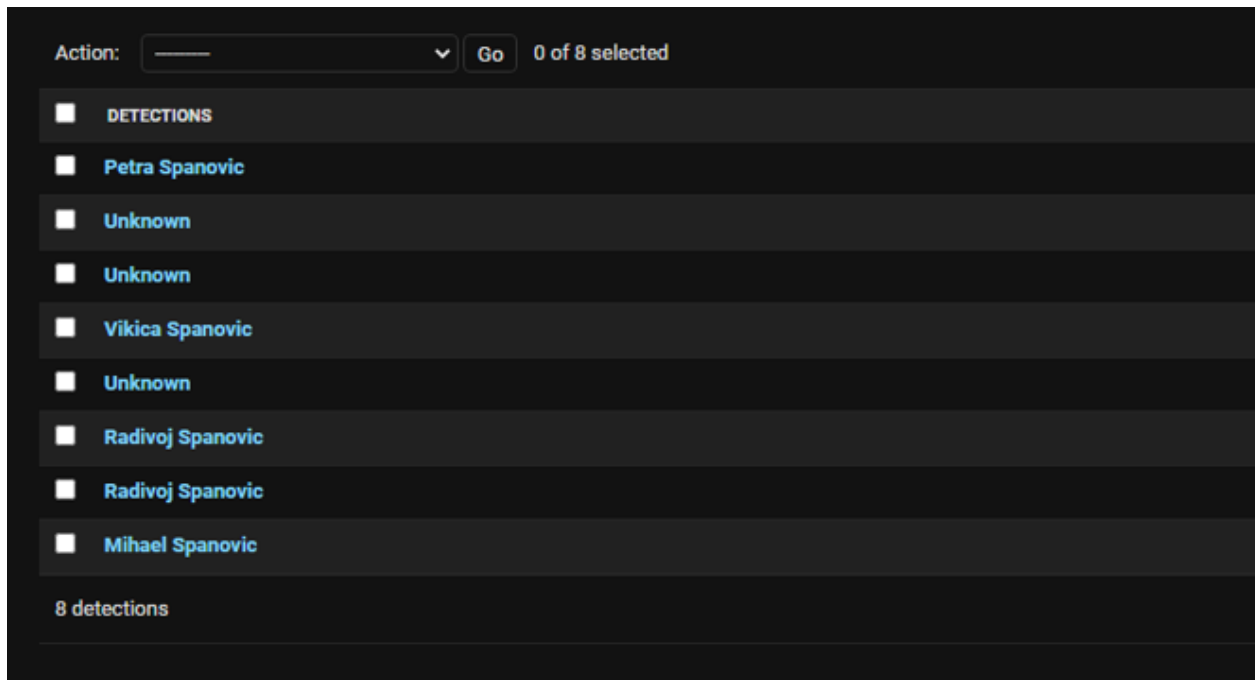
3.4.1. Pohrana detekcije

Kako bi se izbjeglo nepotrebno pisanje u bazu tijekom svake od detekcija što može biti i do nekoliko puta u sekundi postoji rješenje za ograničavanje broja unosa. Ukoliko dođe do procesa prepoznavanja opisanog u poglavlju 3.3., funkciji *mark_detection()* predaje se lista imena pronađenih unutar okvira. Navedenom funkcijom se filtriraju detekcije napravljene unutar postavljenog vremenskog intervala (15 sekundi) te se ponavljajuće detekcije unutar tog intervala ne pohranjuju. U slučaju da je ista prepoznata osoba detektirana više puta unutar postavljenog intervala ona će biti samo jednom zapisana u tablicu za pohranu informacija o detekcijama (Tab. 3.2.). Dodatna mogućnost bi bila spremanje slike ukoliko je detekcija označena kao nepoznata, ali zbog restrikcija postavljenih u vidu performansi, ovaj proces nije izvediv u trenutačnom stanju sustava bez značajnog gubitka na istima.

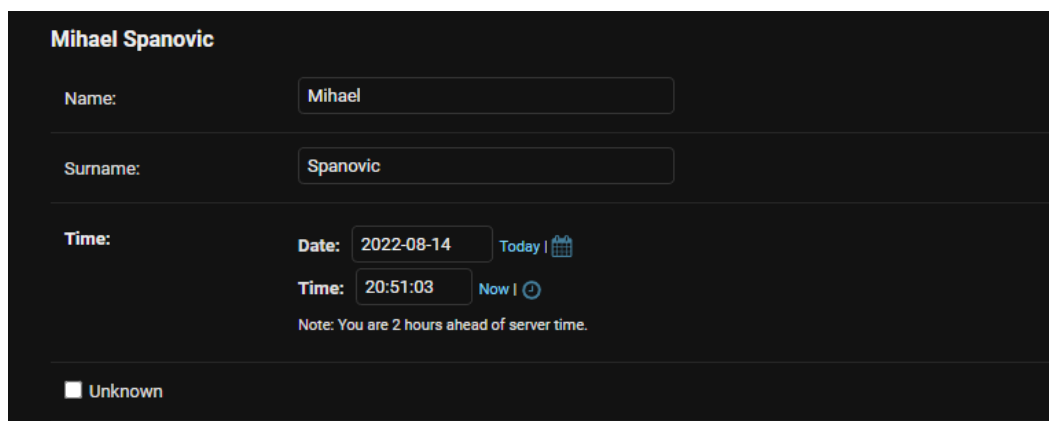
3.5. Administratorsko sučelje

Administratorsko sučelje služi za praćenje detekcija i uređivanje, dodavanje ili brisanje profila iz baze podataka. Pristup *Django* sučelju se ostvaruje pomoću unosa adrese e-pošte i lozinke koja se generira unutar samog sustava. *Django* sučelje služi za praćenje detekcija, dodavanje ili brisanje profila, dok se dodavanje profila može izvršiti ili pomoću *Django* sučelja ili korištenjem jednostavne forme izrađene pomoću *React frameworka* i *Axios* biblioteke. Na slici 3.7. prikazan je izgled *Django* sučelja, unutar sučelja vidljiva su imena i prezimena osoba koje su bile

identificirane. Otvaranjem pojedine stavke vidljivo je vrijeme identifikacije i datum (Slika 3.8.). Izrađeno *React* sučelje služi kao primjer za prikaz mogućnosti integracije sa REST API-em (Slika 3.9.).



Slika 3.7. Sučelje za pregled i uređivanje detekcija osoba



Slika 3.8. Detaljniji prikaz pojedine detekcije

Name

Surname

ID Image

Selfie Image

Submit

Slika 3.9. React sučelje za registraciju osobe

4. EVALUACIJA SUSTAVA

Platforma za testiranje sustava je osobno računalo s Windows operacijskim sustavom, i5 procesorom, 16 Gb radne memorije i web kamerom rezolucije 720x480 piksela. Evaluacija sustava uključuje podsustav za detekciju i podsustav za prepoznavanje. Svaki od koraka evaluacije proveden je u stvarnom vremenu na način da se osoba kroz hodnik približava kameri sustava za identifikaciju što je mogući slučaj upotrebe ovakvog sustava. Podsustavi za detekciju i prepoznavanje evaluirani su u raznim uvjetima osvjetljenja, s više osoba i u slučaju detekcije lica sa mogućim objektima koji služe za prekrivanje samog lica (npr. kirurška maska, zimska kapa, itd.). Također je provedena evaluacija na platformi RaspberryPi 3 Model B kako bi se mogao dobiti uvid u performanse sustava na tipičnoj ugradbenoj računalnoj platformi.

Bitno je napomenuti kako je parametar minimalne granice pouzdanosti detekcije postavljen na 0.6 prilikom izvođenja evaluacije. Što znači da ukoliko je sigurnost modela u predikciju ispod 0.6 smatrati će se kao da se lice ne nalazi unutar danog okvira.

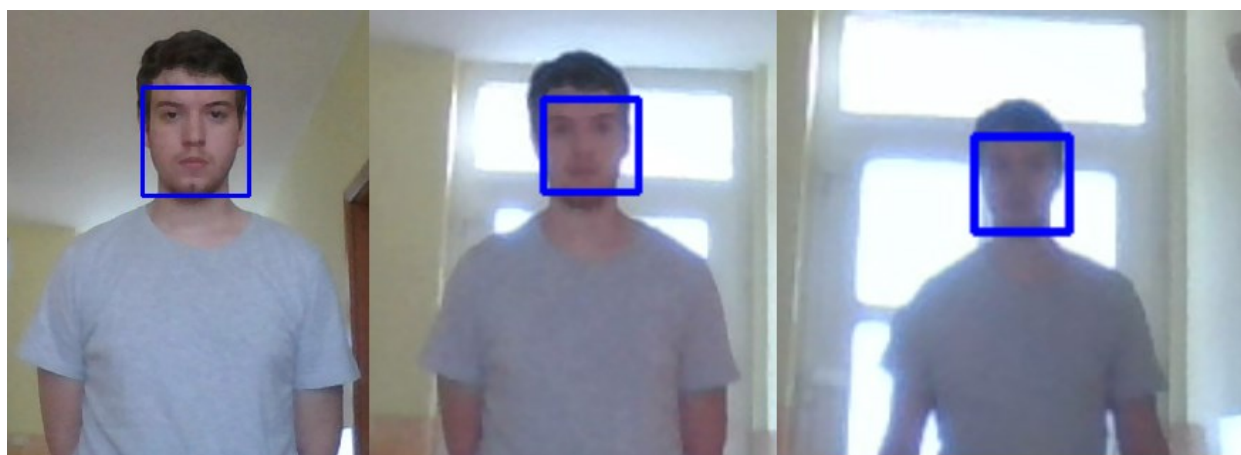
Evaluacija je provedena na temelju preciznosti detekcije/prepoznavanja i na temelju brzine obrade dobivenih slika s kamere koja se daje kao broj obrađenih slika u jedinici vremena (eng. *frames per second* – FPS). Za detekciju definiraju se tri slučaja, prvi slučaj je uspješna detekcija lica (eng. *true positive* -TP), drugi slučaj je kada algoritam na slici pogrešno označi lice (eng. *false positive* -FP) i treći slučaj je kada se lice nalazi na slici, ali nije označeno (eng. *false negative* - FN). Kod prepoznavanja postoje 4 slučaja. Prvi slučaj je kada se osoba koja je pohranjena u bazi nalazi na slici i uspješno je prepoznata - TP, drugi slučaj je kada osoba nije pohranjena u bazu i podsustav prepoznavanja ju označi kao nepoznatu (eng. *true negative* - TN). Označavanje osobe iz baze ili nepoznate osobe kao neke druge osobe pohranjene u bazu predstavlja treći slučaj – FP. Zadnji slučaj je kada se osoba koja je pohranjena u bazu nalazi na slici, ali algoritam kaže da je to nepoznata osoba – FN. Uz definirane slučaje koriste se pojmovi preciznosti i odziva. Preciznost se definira kao omjer TP slučaja i zbroja TP i FP slučajeva. Odziv je omjer TP slučaja i zbroja TP i FN slučajeva. Rezultati za svaku evaluaciju su predstavljeni tablično i uz njih je prikazano nekoliko vizualnih primjera detekcije i prepoznavanja.

4.1. Evaluacija podsustava za detekciju lica

Evaluacija se provodila nad jednom osobom pod normalnim svjetlosnim uvjetima na udaljenostima od otprilike 1, 3 i 5 metara sa i bez okluzija na licu. FPS detekcije bez provođenja prepoznavanja iznosi 60 te je za svaku udaljenost prikupljeno 500 uzoraka.

Tab. 4.1. Evaluacija detekcije lica bez okluzija

Udaljenost(m)	TP	FP	FN	Odziv
1	500	0	0	100%
3	498	0	2	99.6%
5	58	0	442	11.6%



Slika 4.1. Primjer detekcije lica bez okluzija na 1, 3 i 5 metara udaljenosti od kamere

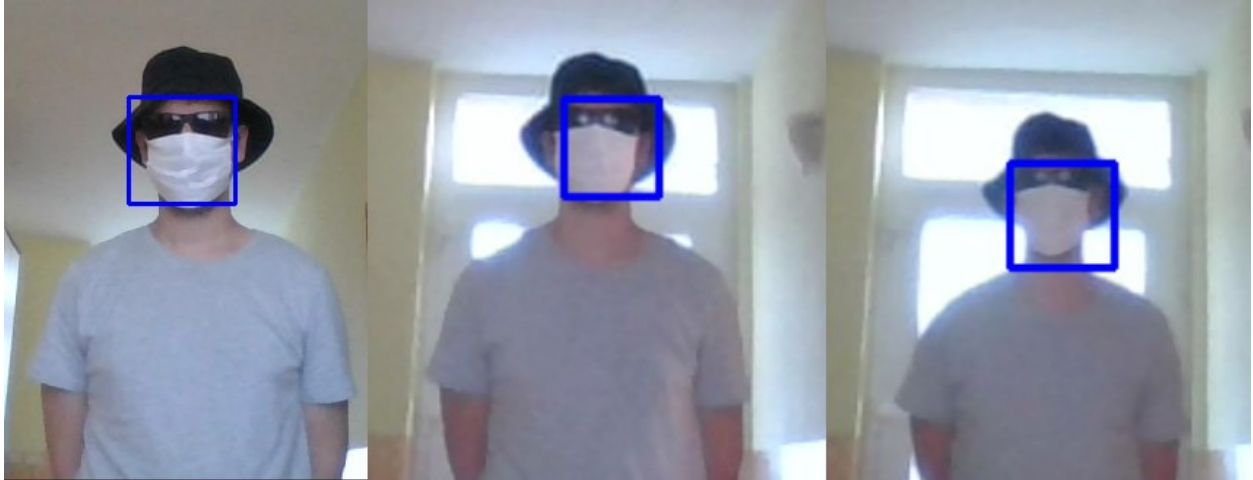
U tablici 4.1. dani su postignuti rezultati iz kojih je uočljivo kako postotak uspješnih detekcija značajno pada na udaljenosti od otprilike 5 metara, na toj udaljenosti sa rezolucijom od 720x480 više nije moguće jasno odrediti ključne točke lica koje su potrebne za detekciju. Na udaljenosti od 3 metra postižu se zadovoljavajući rezultati detekcije što je bitno za sam rad sustava čiji bi proces prepoznavanja lica trebao početi na otprilike toj udaljenosti.

Tab. 4.2. Evaluacija detekcije u slučaju djelomične okluzije lica

Udaljenost(m)	TP	FP	FN	Odziv
1	490	0	10	98%
3	453	0	47	90.6%
5	29	0	471	5.8%

Iz tablice 4.2. uočavamo značajan porast broja neuspješnih detekcija lica na udaljenosti od 3 metra, u usporedbi sa rezultatima iz tablice 4.1 postoji pad od 9% u odzivu te 5.76% pad u pouzdanosti detekcije što je očekivano s obzirom da je uvođenjem okluzija teže detektirati lice na većim

udaljenostima. Prilikom evaluacije detekcije nije zabilježen slučaj FP detekcije tako da preciznost za svaku od udaljenosti i u oba slučaja iznosi 100%.



Slika 4.2. Primjer detekcije lica u slučaju djelomične okluzije na 1,3 i 5 metara udaljenosti od kamere

4.2. Evaluacija podsustava za prepoznavanje lica

Evaluacija je provedena u normalnim svjetlosnim uvjetima na osobnom računalu sa web kamerom rezolucije 720x480 piksela. Testiranje se izvodilo na 5 registriranih korisnika te se simulirala kontrola ulaska u prostoriju. Svaka od osoba bila je udaljena približno 5 metara od kamere te se približavala kameri normalnom brzinom hoda. Podsustav za prepoznavanje lica aktivira se kada unutar okvira detektiramo lice. Proces prepoznavanja lica smanji FPS na 3-5, pošto je sam proces zahtjevniji za izvođenje.

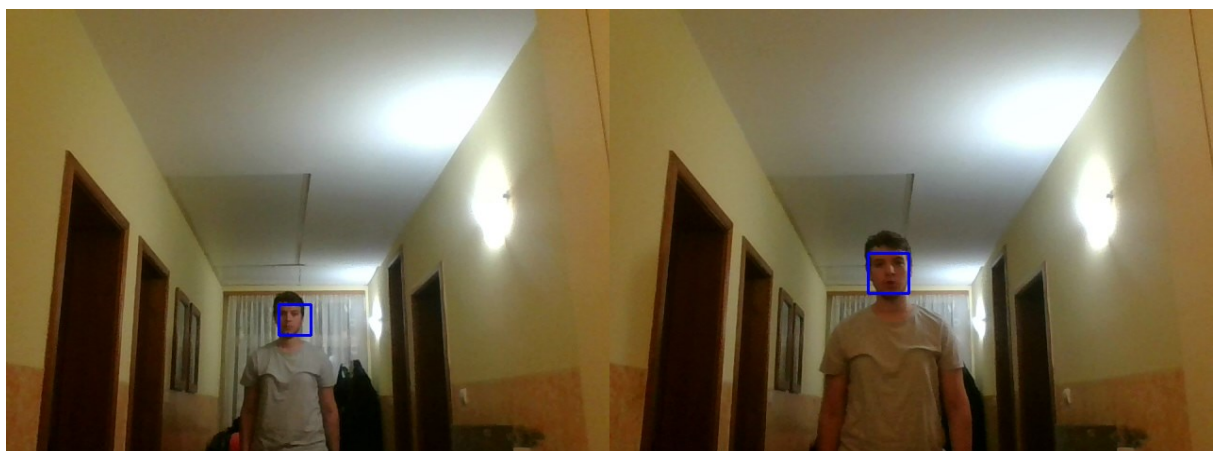
Tab. 4.3. Evaluacija prepoznavanja lica

	OSOBA 1	OSOBA 2	OSOBA 3	OSOBA 4	OSOBA 5
Broj pokušaja identifikacije	18	14	16	18	14
TP	12	14	15	12	12
FP	0	0	1	0	1
FN	6	0	0	6	1
Odziv	66.66%	100%	100%	66.66%	92.31%
Preciznost	100%	100%	93.75%	100%	92.31%

Unatoč nižem postotku odziva kod osobe 1 i 4 (Tab. 3.4.), svaka od osoba bila je uspješno zabilježena u bazu sa vremenom detekcije i prepoznavanja. Također limitirajući faktori za navedeni sustav je prepoznavanje lica na temelju samo dvije slike spremljene unutar baze te niska rezolucija kamere što utječe na točnost prepoznavanja na većim udaljenostima.



Slika 4.3. Isječci okvira sa uspješnim prepoznavanjem lica



Slika 4.4. Isječci okvira s neuspješnim prepoznavanjem lica

Pošto se proces prepoznavanja lica izvršava u trenutku kada podsustav detekcije pronađe lice uvijek će postojati dokumentirani zapis pokušaja prepoznavanja koji kasnije može pregledati odgovorna osoba.

4.3. Evaluacija sustava na RaspberryPi platformi

Evaluacija je također provedena na RaspberryPi 3 Model B v1.2 jednokartičnom računalu sa operacijskim sustavom baziranom na Linux Debianu. Cilj evaluacije je usporediti performanse sustava na dvije različite platforme. Unatoč činjenici da se radi o Debian Linuxu prilagodba napisanog programskog koda nije bila zahtjevna. Metrika za usporedbu sustava je prosječno vrijeme potrebno za izvršavanje procesa prepoznavanja lica. Za svaku od platformi prikupljeno je 30 uzoraka čiji je iznos u sekundama jednak vremenu potrebnom za izvršavanje procesa prepoznavanja lica. Prosječno vrijeme izvršavanja za osobno računalo čije su specifikacije navedene na početku poglavlja 4. iznosi 351 ms. Prosječno vrijeme izvršavanja na RaspberryPi platformi iznosi 2.123 s. Potrebno vrijeme izvršavanja na osobnom računalu je približno 6 puta kraće što je očekivano s obzirom na razliku u performansama.

5. ZAKLJUČAK

Izrađeni sustav temelji se na javno dostupnim, *open source* Python bibliotekama kao što su *Django*, *face_recogniton* i *mediapipe*. Koristeći *Django* izrađen je poslužiteljski dio i administratorsko sučelje, *mediapipe* se koristi za detekciju lica, a podsustav za prepoznavanje koristi *face_recognition*. Sustav analizira ulazne slike sa kamere u potrazi za licem. Slika koja sadrži lice se predaje podsustavu za prepoznavanje koji dobivenu sliku lica uspoređuje sa pohranjenim slikama registriranih korisnika te pokušava identificirati osobu. Informacije o identifikaciji se pohranjuju kako bi se omogućila npr. kontrola pristupa ili zabilježba dolazaka. Iz evaluacije vidljivo je kako sustav ostvaruje najbolje rezultate na udaljenostima manjima od 3 metra. Sustav je ostvario više nego zadovoljavajući rezultat kod prepoznavanja lica, broj FP slučajeva je nizak te je zabilježen visok broj TP slučajeva koji su bitni za pohranu informacije o identifikaciji.

Postoji mogućnost unaprjeđenja sustava povećavanjem rezolucije slike (eng. *upsampling*), ali cijeli proces bi još dodatno utjecao na performanse što nije poželjno osim ako se ne poduzmu dodatne optimizacije na podsustavu za prepoznavanje. Dodavanje više slika u proces registracije bi zasigurno imalo pozitivan utjecaj na proces prepoznavanja pogotovo u lošijim uvjetima. Također ukoliko se radi o velikom broju profila bilo bi potrebno odabrati bržu, skalabilnu bazu kao što je MongoDB ili ukoliko postoji više sustava koje želimo integrirati moguća opcija bila bi postavljanje na Amazon web servise te korištenje njihove baze ili korištenje neovisnog sustava baza kao što je Firebase. Dio koji bi se najviše mogao nadograditi bila bi sigurnost podataka, korištenjem enkripcije za sve osobne podatke korisnika i kontejnerizacija aplikacije (eng. *application containerization*) značajno bi se povećala razina sigurnosti što je ključno za implementaciju u nekontroliranom okruženju.

LITERATURA

- [1] *What is facial recognition technology- and how it works?* [online], Kajal Mishra, 2020. dostupno na: [pristupljeno u srpnju 2022.]
- [2] *Face Detecion* [online], Corinne Bernstein, 2020. dostupno na: <https://www.techtarget.com/searchenterpriseai/definition/face-detection> [pristupljeno u srpnju 2022.]
- [3] *Face Recognition Methods & Applications* [online], Divyarajsinh N. Parmar, Brijesh B. Mehta, 2013. dostupno na: <https://arxiv.org/ftp/arxiv/papers/1403/1403.0485.pdf> [pristupljeno u srpnju 2022.]
- [4] *Top 11 Facial Recognition Software in 2021* [online], Ramya Mohanakrishnan, 2021. dostupno na: <https://www.spiceworks.com/it-security/identity-access-management/articles/facial-recognition-software/> [pristupljeno u srpnju 2022.]
- [5] *What is the Best Facial Recognition Software to Use in 2022?* [online], Serhii Pospelov, 2021. dostupno na: <https://towardsdatascience.com/what-is-the-best-facial-recognition-software-to-use-in-2021-10f0fac51409> [pristupljeno u srpnju 2022.]
- [6] *Techniques and Challenges of Face Recognition: A Critical Review* [online], Shilpi Singh, 2018. dostupno na: <https://www.sciencedirect.com/science/article/pii/S1877050918321252> [pristupljeno u srpnju 2022.]
- [7] *Challenges Faced by Facial Recognition System* [online], Kajal Mishra, 2020. dostupno na: <https://www.pathpartnertech.com/challenges-faced-by-facial-recognition-system/> [pristupljeno u srpnju 2022.]
- [8] *Influence of low resolution of images on reliability of face detection and recognition* [online], Marciniak, T., Chmielewska, A., Weychan, R., 2015. dostupno na: <https://link.springer.com/article/10.1007/s11042-013-1568-8> [pristupljeno u rujnu 2022.]
- [9] *Dlib face recognition* [online], OpenSource, dostupno na: http://dlib.net/face_recognition.py.html [pristupljeno u kolovozu 2022.]

SAŽETAK

Naslov: Sustav za identifikaciju osobe temeljen na računalnom vidu i dubokom učenju

Tema ovog završnog rada je razvoj sustava za identifikaciju osobe koristeći računalni vid i duboko učenje. Opisani su problemi i izazovi razvoja takvog sustava, te su analizirane razne metode koje mogu biti korištene. Postavljeni su zahtjevi na spomenuti sustav i predstavljen je primjer osnovne strukture sustava koji je i programski razvijen koristeći programski jezik Python. Zatim je provedeno testiranje nad podsustavima detekcije i prepoznavanja te su predstavljeni dobiveni rezultati.

Ključne riječi: detekcija lica, duboko učenje, identifikacija lica, računalni vid

ABSTRACT

Title: Person identification system based on computer vision and deep learning

The topic of this final thesis is the development of a system for person identification using computer vision and deep learning. The problems and challenges of developing such a system are described, and various methods that can be used are analyzed. The requirements for the mentioned system were set and an example of the basic structure of the system was presented, which was also developed using the Python programming language. Then the detection and recognition subsystems were tested and the obtained results were presented.

Keywords: face detection, deep learning, face identification, computer vision

PRILOZI

Prilog 1: Programski kod na priloženom mediju za pohranu:

<https://github.com/Mihael283/FaceRecogniton-Django-React>