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

**Sveučilišni studij**

**KLASIFIKACIJA EMOCIJA NA OSNOVI SLIKE LICA**

**Diplomski rad**

**Josip Baketarić**

**Osijek, 2017.**

**Sadržaj**

[1. UVOD 1](#_Toc475725993)

[1.1. Zadatak rada 1](#_Toc475725994)

[2. RAČUNALNO RASPOZNAVANJE EMOCIJA 2](#_Toc475725995)

[2.1. Emocije i afekt 2](#_Toc475725996)

[2.2. Modaliteti podataka 3](#_Toc475725997)

[2.2.1. Slike lica kao izvor informacija 3](#_Toc475725998)

[2.3. Računalni postupci 4](#_Toc475725999)

[2.3.1. Klasifikacijski problem 4](#_Toc475726000)

[2.3.2. Viola-Jones algoritam 5](#_Toc475726001)

[2.3.3. Gaborovi filteri 6](#_Toc475726002)

[2.3.4. Analiza glavnih komponenti 7](#_Toc475726003)

[2.3.5. Vektori podrške 7](#_Toc475726004)

[2.4. Komercijalna rješenja 10](#_Toc475726005)

[3. RJEŠENJE ZA RASPOZNAVANJE EMOCIJA NA OSNOVI LICA 12](#_Toc475726006)

[3.1. Specifikacije i zahtjevi 12](#_Toc475726007)

[3.2. Korišteni alati i tehnologije 14](#_Toc475726008)

[3.2.1. OpenCV 14](#_Toc475726009)

[3.2.2. EmguCV 14](#_Toc475726010)

[3.2.3. Accord.NET 15](#_Toc475726011)

[3.2.4. AForge.NET 15](#_Toc475726012)

[3.2.5. Weka 15](#_Toc475726013)

[3.2.6. ARFF format 17](#_Toc475726014)

[3.3. Prikaz ključnih elemenata rješenja 17](#_Toc475726015)

[3.4. Prikaz izgleda, rad i uporaba rješenja 20](#_Toc475726016)

[3.5. Analiza ponašanja i performansi 25](#_Toc475726017)

[3.5.1. Analiza rezultata na trening podatcima 25](#_Toc475726018)

[3.5.2. Analiza rezultata na nepoznatim podatcima 26](#_Toc475726019)

[3.6. Osvrt i mogućnost unaprjeđenja 28](#_Toc475726020)

[4. ZAKLJUČAK 30](#_Toc475726021)

[5. LITERATURA 31](#_Toc475726022)

[6. SAŽETAK 33](#_Toc475726023)

[7. ŽIVOTOPIS 34](#_Toc475726024)

[8. PRILOZI 35](#_Toc475726025)

1. UVOD

„Budućnost leži u projektiranju i prodaji računala za koje ljudi ne shvaćaju da su računala.“ - Adam Osborne.

Prilikom međuljudske komunikacije, a i komunikacije između ostalih živih bića, govor ima jako važnu ulogu. Jednako važnu, ako ne i važniju ulogu prilikom komunikacije imaju emocije koje, bolje od izrečenih riječi, pokazuju trenutno emocionalno stanje osobe koje se naziva afektivno stanje. U svijetu gdje je sve veća komunikacija između računala i čovjeka ne može se zanemariti jedan tako važan aspekt ljudske prirode kao što su emocije. Kao što se sad na temelju pretraživanja sadržaja na tražilicama nude odabrane reklame, tako će se u budućnosti na temelju afektivnog stanja nuditi određeni sadržaji.

U drugom poglavlju opisani su podatci te računalni postupci na temelju kojih se raspoznavaju emocije. Sljedeće poglavlje sadrži opis i analizu programskog rješenja kao i analizu dobivenih rezultata. Zadnje poglavlje namijenjeno je za osvrt i kratki zaključak implementiranog rješenja.

* 1. Zadatak rada

Cilj rada jest istražiti mogućnosti računalnog raspoznavanja emocija i afektivnih stanja. U teorijskom dijelu rada potrebno je opisati stanje u relevantnoj literaturi u pogledu modaliteta korištenih podataka, računalnih postupaka i dostupnih komercijalnih alata s posebnim naglaskom na uporabu slika lica kao izvora informacija. U praktičnom dijelu rada potrebno je implementirati vlastito rješenje koje omogućuje raspoznavanje emocija uporabom slika lica.

1. RAČUNALNO RASPOZNAVANJE EMOCIJA

Proučavanje emocija postoji već jako puno godina i veliki broj radova napisan je na tu temu. Jedan od temeljnih radova na spomenutu temu je Darwinov rad iz 1872. godine koji povezuje fizičke radnje, kao na primjer podizanje obrva u slučaju iznenađenja ili zbunjenosti, sa stanjem uma [1]. Raspoznavanje emocija složen je proces koji uključuje kontekst, govor tijela, kulturna obilježja, izraze lica te još par stvari [2]. U ovom radu prepoznavanje emocija bazirano je samo na izrazima lica. Računalno prepoznavanje emocija ima raznoliku upotrebu, sve od igara u kojima bi se moglo pomoću detektiranih emocija prepoznati koliko je igra zanimljiva ili dosadna pa do različitih softvera za učenje i sustava namijenjenih za brigu o zdravlju korisnika (detekcija panike).

* 1. Emocije i afekt

Emocije su nešto što je teško definirati. Bilo je mnogo je pokušaja da se emocije definiraju, a jedan od njih glasi da su emocije mentalna stanja [3]. Postoji više mišljenja o tome koliko ima emocija i koje su, ali neka osnovna podjela kaže da postoji šest različitih (osnovnih) emocija. To su: sreća, tuga, ljutnja, iznenađenje, strah i gađenje. U ovom radu vršit će se detekcija spomenutih šest emocija.

Emocije se izražavaju kroz cijelo ljudsko tijelo. Mogu se prepoznati iz tona, načina pisanja, ponašanja te izraza lica. Ovaj se rad bazira na prepoznavanju emocija pomoću slika lica. Ovisno o nekoliko parametara, na licu se mogu prepoznati osnovne emocije [4]:

1. Ljutnja: Obrve su povučene prema dolje, gornji i donji kapci su povučeni prema gore, usne zategnute i malo povučene prema unutra.
2. Strah: Obrve povučene prema gore i jedna prema drugoj, gornji kapci povučeni prema gore i usta rastegnuta.
3. Gađenje: Obrve su povučene prema dolje, nos je naboran, gornja usna povučena prema gore, usne opuštene.
4. Sreća: Mišići oko očiju su stegnuti, bore oko očiju, obrazi podignuti i krajevi usana podignuti dijagonalno.
5. Tuga: Unutarnji krajevi obrva su podignuti, kapci opušteni i krajevi usana su spušteni prema dolje.
6. Iznenađenje: Obrve podignute prema gore, kapci podignuti prema gore i usta otvorena.

Spomenuti parametri pomoću kojih se mogu raspoznavati emocije još se nazivaju i akcijske jedinice i ima ih sve ukupno oko sto. Glavnih akcijskih jedinica, koje se ujedno mogu poznati na licu, ima 46.

* 1. Modaliteti podataka

Emocije su nešto što se odražava na cijelo biće i zbog toga ih je moguće prepoznati na više načina. Obično uz izraz lica dolaze i druge popratne pojave, ako je emocija snažna. Način govora promjeni se u smislu visine tona, brzine pričanja, podrhtavanja glasa i slično. Neke od bitnih značajki prepoznaju se i po pokretima tijela kao što su skupljanje ramena, tapkanje nogama, trešnja ruku te mnoge druge. Emocije se očituju i po načinu pa čak i po brzini pisanja, ali takve značajke često znaju varirati od osobe do osobe. Uza sve spomenute načine prepoznavanja emocija i kultura čini bitnu razliku.

Slike lica kao izvor informacija

Kako bi bilo moguće prepoznati emociju, to jest klasificirati izraz lica, potrebno je imati podatke za testiranje i ocjenu korištenih algoritama. Podatci korišteni u ovom radu su slike lica osoba. Po osobi ima ukupno sedam slika, šest sa prije navedenim emocijama i jedna koja je neutralna (bez emocija).

Slike koje će biti korištene prilikom stvaranja modela za klasifikaciju su slike lica koje se sastoje od 70 individua sa sedam različitih emocija te su slikane iz različitih kutova. U svrhu ovog rada koristiti će se samo frontalne slike. Slike su preuzete sa Karolinska instituta koji se nalazi u Švedskoj [5]. Primjer slika je prikazan na slici 2.1.

|  |
| --- |
|  |
| Sl. 2.1. Primjer korištenih slika u radu, [5] (©Karolinska Institutet, Department of Clinical Neuroscience, Section of Psychology, Stockholm, Sweden.) |

* 1. Računalni postupci

Cilj ovog rada je naučiti računalo da prepoznaje emocije osoba sa slika lica. Da bi se to moglo ostvariti prvo je potrebno prikupiti odgovarajući skup podataka s kojim će se raditi. Nakon odgovarajuće baze podataka, potrebno je napraviti pred obradu slika, kao što je na primjer promjena veličine, pretvorba slike u sliku sa sivim tonovima i slično. Iz pripremljenih slika treba izdvojiti informacije koje će biti relevantne za prepoznavanje emocija te pomoću kojih će se formirati vektor značajki i na kraju istrenirati klasifikator.

Klasifikacijski problem

Klasifikacija uzoraka znanstvena je disciplina koja ima za zadatak razvrstavanje objekata u klase, prema [6, str.1]. Problem se nalazi u odabiru reprezentativnih značajki. Mali pokreti određenih dijelova lica određuju dali je osoba nasmijana, tužna ili nešto drugo. Ako se prilikom oblikovanja modela, prema kojem se vrši klasifikacija, koriste nepotrebne informacije (značajke dobivene iz regija pomoću kojih nije moguće detektirati emociju) vrlo je vjerojatno da će relevantne informacije biti zagušene i uspješnost klasifikacije će biti manja. Kako bi se utjecaj nebitnih ili manje bitnih regija lica smanjio, prilikom odabira značajki koriste se samo dijelovi lica gdje se nalaze oči, usta, nos i obrve. Iz svake regije potrebno je nekim metodama dobiti dovoljan broj informacija i kreirati vektor značajki koji će služiti prilikom treniranja klasifikatora. Klasifikator je algoritam koji na temelju značajki određuje klasu, ili u ovom slučaju emociju [7, str. 24].

Viola-Jones algoritam

Da bi se mogle raspoznavati emocije na licu prvo je potrebno imati sliku lica. Iako su za ovaj rad korištene slike lica, dodatno se koristi i algoritam koji sa slika detektira samo lica kako bi se izbjegao utjecaj okoline u kojoj je slika nastala. Jedan od najpoznatijih algoritama za detekciju lica na slikama je Viola-Jones algoritam. Algoritam koristi Haarove značajke koje primjenjuje na sliku i na taj način traži odgovarajući objekt, u ovom slučaju lice, usta, oči i nos [8].Haarove značajke su slike ili matrice koje na nekim mjestima imaju tamna, a na nekim mjestima svjetla područja i svojim oblikom odgovaraju traženom objektu. Primjer se nalazi na slici 2.2.

|  |
| --- |
|  |
| Sl. 2.2. Primjer Haarove značajke, [8] |

Značajke se primjenjuju na cijelu sliku od početka do kraja i pomiču se po jedan piksel nakon primjene. Kada se prođe cijela slika, značajka se povećava i sve se opet primjenjuje ponovno. Ovo nije učinkovito raditi jer je jako sporo. Kako bi se ubrzao proces traženja objekata, Viola-Jones koriste nešto što se zove integralna slika i također se definira minimalna i maksimalna granica do koje se povećava značajka. Integralna slika je proces koji omogućuje računanje zbroja piksela unutar bilo kojeg kvadrata koristeći samo referentna polja, što je prikazano na slici 2.3. [8]. Vrijednost integralne slike na lokaciji 1 je zbroj piksela u A kvadratu. Vrijednost na lokaciji 2 je A + B, na lokaciji 3 A + C i na lokaciji 4 A + B +C +D. Zbroj piksela unutar polja D može se izračunati na način da se zbroje vrijednosti na glavnoj dijagonali i oduzmu od zbroja vrijednosti na sporednoj dijagonali (1 + 4 - ( 2 + 3) ).

|  |
| --- |
|  |
| Sl. 2.1. Primjer algoritma integralne slike, [8] |

Kako bi se algoritam još ubrzao izbacuju se nepotrebne značajke pomoću Adaboost-a. Adaboost je algoritam koji izdvaja optimalne značajke iz jako velikog skupa. Nakon izdvajanja značajki, dodjeljuje im se težinska vrijednost [9]. Smatra se da je značajka prihvatljiva, ako može detektirati traženi objekt u barem pola slučajeva.

Gaborovi filteri

Gaborov filter je kompleksna sinusoidna funkcija, jednadžba 2-1, koja se primjenjuje na određenim frekvencijama i orijentacijama. Koristi se prilikom detekcije kutova, granica, klasifikacija tekstura [10]…

|  |  |  |
| --- | --- | --- |
|  |  | (2-1) |
|  |  | (2-2) |
|  |  | (2-3) |

gdje je:

-*g* – Gaborova funkcija,

-*x*, y - koordinate

-– valna dužina,

-– orijentacija,

- – Fazni pomak,

- – standardna devijacija,

- - omjer slike.

Na slici 2.2. se nalazi primjer Gaborovih filtera primijenjenih na sliku lica u četiri različite orijentacije i četiri različite valne dužine (sve ukupno 16 filtera).

|  |
| --- |
|  |
| Sl. 2.2. Gaborovi filteri primijenjeni na sliku lica |

Dvije najvažnije komponente Gaborovih filtera su valna dužina i orijentacija. Orijentacija određuje koje će se linije na slici prepoznavati i vrijednost joj se kreće od 0° do 360°. Što se tiče valne dužine, ona određuje širinu filtera koji se primjenjuje i što je širi, više linija je moguće naći, ali je moguće i pogriješiti, ako je filter preširok. Nakon što su definirani filteri, tako zvana banka filtera, oni se primjenjuju na sliku i kao rezultat dobiva se nova slika koja ima visok intenzitet piksela gdje se poklapa s filterom.

Analiza glavnih komponenti

Analiza glavnih komponenti (engl. *Principal component analysis*, PCA) je statistička metoda koja služi za smanjenje dimenzionalnosti podataka, detekciju ne pripadajućih vrijednosti i tako dalje [11]. Jedna od glavnih primjena PCA je smanjivanje dimenzionalnosti. PCA prebacuje podatke u manju dimenziju na način da gleda najrelevantnije podatke, a ostale zanemaruje i time se uz male gubitke informacija postigne velika ušteda u smislu brzine izračuna. PCA određuje za svaku dimenziju koliko ona sadrži kvalitetnih ili korisnih informacija i poreda ih u smislu osnovnih komponenti. Ako bi postojao prostor od 200 dimenzija, PCA bi odredio 200 glavnih komponenti koje bi bile poredane po korisnosti. Prva glavna komponenta bi bila i najkorisnija, pa iza nje druga i tako dalje. Na taj se način može odrediti koliko će se koristiti dimenzija i izbjegavanje nepotrebnih podataka koje uz dodatni napor prilikom računanja mogu unijeti i pogrešku. Na slici 2.3. prikazan je dvodimenzionalni graf s podatcima. Iz grafa se vidi da se podatci prostiru skoro kao po pravcu uz malo odstupanja. PCA prebacuje podatke u dvodimenzionalni prostor u kojemu se umjesto osi, nalaze glavne komponente. Vidi se da su podatci rasprostranjeni po glavnoj komponenti 1, a dosta manje po glavnoj komponenti 2. Ako bi se htjelo smanjiti dimenzionalnost podataka bez puno gubitaka, može se promatrati samo projekcija podataka na glavnu komponentu 1 jer ona sadrži većinu informacija.

|  |
| --- |
|  |
| Sl. 2.3. PCA glavne komponente |

Vektori podrške

Metoda vektora podrške (engl. *Support Vector Machines*, SVM) je binarni klasifikator koji se koristi pri klasifikaciji i regresiji [6, 119. str. – 138. str]. Prostorno promatrano, SVM pokušava odvojiti značajke suprotnih klasa hiper-ravninom tako da je razmak između najbližih elemenata i hiper-ravnine najveći, slika 2.4. Kada podatci nemaju oznaku klase, nadzirano učenje nije moguće. Potrebno je dobivene podatke prvo grupirati i potom izračunati optimalnu hiper-ravninu.

|  |
| --- |
|  |
| Sl. 2.4. SVM |

U slučaju linearno odvojivih podataka, što znači da se može konstruirati hiper-ravnina koja odvaja uspješno dijeli sve podatke na dvije različite klase, hiper-ravnina se opisuje izrazom (2-4).

|  |  |  |
| --- | --- | --- |
|  |  | (2-4) |

gdje je:

-*w* – normala hiper-ravnine,

- - okomita udaljenost hiper-ravnine od ishodišta

- – podatak.

Hiper-ravnine koje odvajaju podatke i između sebe imaju najveću udaljenost su margine i mogu se opisati formulama (2-5) i (2-6), te margine se još nazivaju i vektori podrške.

|  |  |  |
| --- | --- | --- |
|  |  | (2-5) |
|  |  | (2-6) |

Ako se želi SVM koristiti i na linearno neodvojive podatke, potrebno je dozvoliti određenu grešku prilikom stvaranja hiper-ravnine. Kao i prethodno, u ovom slučaju se margine pokušavaju konstruirati sa što većim razmakom, ali pritom da je što manje pogrešno klasificiranih elemenata. Na slici 2.5. prikazana je hiper-ravnina u slučaju linearno neodvojivih podataka. Odabir hiper-ravnine svodi se na pronalaženje:

|  |  |  |
| --- | --- | --- |
|  | takav da | (2-7) |

gdje je:

-*C* – faktor pogreške,

- - mjera udaljenosti promatranog podatka od pripadajućeg vektora podrške.

|  |
| --- |
|  |
| Sl. 2.5. SVM hiper-ravnina sa linearno neodvojivim podatcima |

Iako binarni, klasifikator SVM može se koristiti i prilikom više klasne klasifikacije. Dva su moguća načina na koje se SVM može koristiti kao više klasni klasifikator:

1. Konstruirati n klasifikatora koji svrstava podatke u jednu od klasa ili u preostale n-1 klase. Uz rezultat klasifikacije, klasifikator vraća i mjeru sigurnosti u svoj odabir. Nakon što svi klasifikatori odrade klasifikaciju, u obzir se uzimaju samo oni koji nisu podatku pridružili klasu „ostali“. Rezultat se uzima od klasifikatora sa najvećom mjerom sigurnosti. Ovakav način rada zove se jedan protiv svih (engl. *One versus all*). U slučaju da svi klasifikatori odaberu klasu „ostali“, rezultat se uzima od onog sa najmanjom mjerom sigurnosti. Prikaz rada algoritma nalazi se na slici 2.6.
2. Konstruirati klasifikatora koji svaki pojedinačno određuje klasu podatka. Svaki put kad klasifikator odredi klasu toj klasi se pridodaje jedan glas. Nakon što se svi klasifikatori izvedu i postupak glasovanja završi, vrijeme je za prebrojavanje glasova. Klasa s najvećim brojem glasova pridjeljuje se podatku. U slučaju istog broja glasova, obično se pridjeljuje klasa sa manjim indeksom. Ovakav način rada zove se jedan protiv jednog (engl. *One versus one*). Prikaz rada algoritma nalazi se na slici 2.7.

|  |
| --- |
|  |
| Sl. 2.6. SVM način rada algoritma jedan protiv svih |

|  |
| --- |
|  |
| Sl. 2.7. SVM način rada algoritma jedan protiv jednog |

* 1. Komercijalna rješenja

Prepoznavanje emocija nije ništa novo te postoji dosta gotovih rješenja za privatnu, a i komercijalnu upotrebu. Neka od poznatijih rješenja su:

1. Affectiva je jako dobar alat za prepoznavanje emocija u stvarnom vremenu. Besplatan je za osobno korištenje i za tvrtke koje godišnje zarade manje od milijun dolara. Postiže veliku preciznost jer ima ogromnu bazu podataka za treniranje klasifikatora (četiri milijuna lica i konstantno raste). Može se koristiti u razne svrhe, ali glavna ideja je dobivanje povratne informacije od korisnika o nekom proizvodu [12].
2. EmoVu uz prepoznavanje emocija također nudi i prepoznavanje spola, godina, lica, praćenje lica i metrike raspoloženja. Ulazni podatci šalju se pomoću kamere, slike i videa. Pruža se podrška prebacivanja izračuna na grafički procesor kako bi se proces ubrzao [13].
3. Kairos također pruža uslugu prepoznavanja emocija, spola i dobi. Njihovo rješenje može se besplatno isprobati na njihovoj stranici [14].
4. Microsoftov projekt Oksford je popularan alat za prepoznavanje emocija koji radi samo na slikama lica. Prepoznaje sedam emocija i neutralno stanje. Radi na način da se pošalje slika, a kao rezultat dobije se JSON (*JavaScript Object Notation* ) format sa lokacijom lica na slici i postotkom za svaku od pruženih emocija [15].
5. RJEŠENJE ZA RASPOZNAVANJE EMOCIJA NA OSNOVI LICA

Cilj rješenja je pokušati što točnije prepoznati emocije na osnovu slika lica. Prepoznavanje emocija korisnika dosta je korisna informacija u smislu ocjenjivanja neke aplikacije i služi kao jako dobar povratni odgovor proizvođaču. Postoji dvadesetak rješenja od velikih firmi koji omogućuju korištenje njihovih rješenja u poslovne i privatne svrhe. Većina tih rješenja zahtjeva dosta računalne snage jer se koristi veliki broj klasifikatora prilikom analize lica u potrazi za akcijskim jedinicama i za svaku akcijsku jedinicu koju se želi prepoznati potrebno je imati poseban klasifikator. Većinom se sva klasifikacija vrši na njihovim računalima, a rezultat se vraća korisniku. U ovom radu pokušati će se pojednostavniti taj proces uz što manje gubitaka i što brže te će se koristiti samo jedan klasifikator.

* 1. Specifikacije i zahtjevi

Aplikacija mora moći učitati sliku i/ili dohvatiti sliku s kamere. Na učitanoj slici program mora pokušati pronaći lice. U slučaju pronalaska lica na slici, poziva se rutina za obradu slike i prepoznavanje emocije s pripadajuće slike lica. Razvojno okruženje koje se koristi je *Visual Studio*, a programski jezik C#. Aplikacija se dijeli u nekoliko projekata:

* Projekt.Zajedničko– Projekt koji je referenciran u svim ostalim projektima. Koristi se za pomoćne transportne klase i sliče stvari.
* Projekt.Dohvaćanje značajki– Konzolna aplikacija. Sastoji se od tri klase. Klasa program i dvije klase za zapis značajki u datoteku u dva različita formata.

1. Program- Klasa koja služi za učitavanje podataka iz baze (baza sa slikama koja se nalazi lokalno na računalu). Poziva metodu za prepoznavanje lica na slici. Poziva metodu iz servisa koja obrađuje dohvaćeno lice i vraća vektor značajki. Nakon dohvaćenog vektora značajki, poziva metode za spremanje značajki na disk.
2. Zapis značajki– Klasa koja prima listu značajki i zapisuje ih u jednu .txt datoteku.
3. Zapis značajki arff– Klasa koja prima listu značajki i zapisuje ih u .arff formatu koji je kasnije u radu opisan.

* Projekt.Servis– Biblioteka klasa (engl. *Class library*). Glavni dio ovog projekta je klasa za obradu učitane slike, klasa za prepoznavanje lica na slici i klasa koja služi za generiranje Gaborovih filtera koji se primjenjuju na sliku.

1. Obrada slike– Klasa koja prima sliku, a kao rezultat vraća vektor značajki. Primljenoj se slici širina i visina postavlja na fiksnu dimenziju (68x80) i pretvara se u sliku sivih tonova. Slika se potom izrezuje u pet regija: gore lijevo, gore desno, dolje lijevo, dolje sredina i dolje desno. Te se regije koriste prilikom generiranja značajki. Svaka se regija šalje na obradu pomoću Gaborovih filtera koji vraća vektor značajki za tu regiju. Nakon obrade svih regija, generira se jedan vektor značajki koji reprezentira ulaznu sliku.
2. Klasifikator– Ovo je klasa koja je zadužena za pronalaženje lica na slici. Koristi se vanjski *framework* EmguCV i Haarov klasifikator.
3. Klasa filter– Prima ulaznu sliku. Generira šesnaest Gaborovih filtera različitih orijentacija i valnih dužina. Ti filteri se potom primjenjuju na sliku koja onda predstavlja korisne značajke. Kako bi se smanjio broj značajki, koristi se PCA. Poziva se metoda koja prima sliku i pretvara ju u 2D polje to jest matricu. Na matricu se primjenjuje PCA i onda se umjesto cijele slike, koristi samo jedan dio. Na kraju metoda vraća vektor značajki.
4. Pomoćne klase– Služe za pretvaranje slike iz *Bitmap* formata u 2D polje i obrnuto, pretvaranje slike u sliku sivih tonova, rezanje slike, promjenu veličine slike…

* Projekt.Weka– Biblioteka klasa. Služi za treniranje algoritma i klasifikaciju. Sastoji se od dvije klase: klasifikator i klasificiraj. Koristi posebnu Wekinu biblioteku kako bi se mogli koristiti klasifikatori.

1. Klasifikator- Klasa koja služi za treniranje i ocjenu klasifikatora. Korišteni klasifikator je SVM.
2. Klasificiraj- Klasa koja omogućuje klasifikaciju novih podataka. Kreira novi vektor, koji je prigodan za korištenje od strane klasifikatora, na temelju dobivenog vektora značajki. Kao rezultat vraća predviđenu klasu.

* Projekt.Form- *WindowsForms* aplikacija. Omogućuje korisniku da koristi kameru kako bi predvidio emociju i omogućuje učitavanje slike sa diska. Nakon dohvaćanja slike poziva metodu za obradu slike i metodu za klasifikaciju te na temelju dobivenih značajki ispisuje korisniku rezultat u obliku predviđene emocije.

Dijagram toka aplikacije prikazan je na slici 3.1.

|  |
| --- |
|  |
| Sl. 3.1. Dijagram toka |

* 1. Korišteni alati i tehnologije

Prilikom izrade aplikacije korišteno je nekoliko vanjskih *frameworka* kao pomoć pri izradi. To su: OpenCV, EmguCV, Accord.NET, Weka i AForge.NET. Svaki od korištenih *frameworka* je slobodan za korištenje.

OpenCV

OpenCV (*Open Source Computer Vision Library*) je *framework* otvorenog koda koja se koristi za računalni vid i strojno učenje [16]. Može se besplatno koristiti u nekomercijalne, ali i komercijalne svrhe. Sadrži više od 2500 optimiziranih algoritama koji se mogu koristiti u svrhu detekcije i prepoznavanja lica, praćenje pokreta, 3D obrada slike i videa, spajanje slika u svrhu proizvodnje slike visoke rezolucije i još mnogo toga [16]. OpenCV koriste svi od osobnih korisnika, manjih tvrtki pa sve do velikih kompanija poput Google-a, Yahoo-a, Intela-a, IBM-a, Microsoft-a… Pisan je u programskom jeziku C++ i postoji mnogo *frameworka* pomoću kojih se može koristiti i u drugim programskim jezicima.

EmguCV

EmguCV je *framework* koji služi kako bi se metode iz OpenCV-a mogli koristiti u .NET kompatibilnim programskim jezicima kao što su C#, Python, VB… Pisan je isključivo u C#. Slobodan je za korištenje ako je rađena aplikacija otvorenog koda [17].

Accord.NET

Accord.NET je *framework* otvorenog koda koji se koristi za strojno učenje, obradu slike i zvuka, obradu signala, statističke primjene, prepoznavanje uzoraka… [18]… Pisan je u programskom jeziku C#. Podijeljen je u biblioteke:

* Znanstveno računanje- matematičke funkcije, statistika, strojno učenje neuronske mreže…
* Obrada slike i signala- razni filteri za slike, spajanje više slika u jednu, kreiranje integralnih slika, filteri za zvuk, detekcija lica u stvarnom vremenu…
* Biblioteke potpore- histogrami, grafovi, komponente za prikaz specifičnih slika i zvukova…

Kao i prethodne biblioteke, i ova je slobodna za korištenje.

AForge.NET

AForge.NET je *framework* otvorenog koda razvijen u C#-u. Dizajniran je za rad na područjima računalnog vida, umjetne inteligencije – obrada slike, neuronskih mreža, genetskih algoritama, strojnog učenja, robotike i slično [19]. Konstantno se održava i omogućuju mu se nove funkcionalnosti. Postoji mnogo primjera aplikacija koje su rađene u ovom *frameworku*.

Weka

Weka je alat za rudarenje podatcima koji omogućuje korištenje i analizu performansi postupkom strojnog učenja [20]. Weka je razvijena u programskom jeziku Java. Sastoji se nekoliko načina rada: istraživačkog, pokusnog. toka podataka i naredbenog retka. Prikaz početnog korisničkog sučelja prikazan je na slici 3.2.

|  |
| --- |
|  |
| Sl. 3.2. Wekino početno sučelje |

Način rada u weki:

1. Učitavanje podataka u .arff formatu.
2. Pred obrada podataka.
3. Odabir željene akcija.
4. Klasifikacija- biranje klasifikatora i podešavanje opcija za treniranje.
5. Prepoznavanje nakupina podataka- odabir algoritma i specificiranje načina rada.
6. Učenje asocijacijskih pravila- odabir algoritma i podešavanje te prikaz rezultata.
7. Odabir atributa- algoritmi za izdvajanje relevantnih od irelevantnih atributa u učitanom skupu podataka.
8. Vizualizacija- Vizualni prikaz ovisnosti klasa o atributima
9. Prikaz rezultata. Rezultati se prikazuju pomoću matrice pogrešaka. Matrica pogrešaka je matrica koja na glavnoj dijagonali ima brojeve koji kazuju koliko je točno klasificiranih podataka za svaku od klasa, a ostali koliko je kojih klasa zamijenjeno za tu. Primjer matrice pogrešaka prikazan je na slici 3.3.

Weka omogućuje i uključivanje njezinih funkcionalnosti u neki od programskih jezika. Izvorno je to Java, ali se može prevesti da radi i u drugim jezicima poput C#-a.

|  |
| --- |
|  |
| Sl. 3.3. Matrica pogrešaka |

ARFF format

ARFF je oblik zapisa podataka koji se koristi u Weki. Sastoji se od dva djela. Zaglavlje u kojem se definiraju imena i vrsta atributa te dio gdje su podatci. Zaglavlje počinje s ključnom riječi @RELATION nakon čega se specificira ime relacije i onda slijedi nabrajanje atributa kao što je prikazano na slici 3.4.

|  |
| --- |
|  |
| Sl. 3.4. ARFF zaglavlje |

Podatci se odvajaju od zaglavlja ključnom riječi @DATA. Atributi su odvojeni zarezom, a podatci su svaki u svom redu, slika 3.5.

|  |
| --- |
|  |
| Sl. 3.5. ARFF podatci |

Komentari se pišu tako da se na početak reda u kojemu je komentar stavi znak „%“.

* 1. Prikaz ključnih elemenata rješenja

Aplikacija se sastoji iz dva dijela. Prvi dio je za stvaranje podataka koji se koriste pri treniranju klasifikatora. Drugi dio služi za prepoznavanje emocije preko istreniranog klasifikatora na temelju dobivenih podataka. Program za stvaranje podataka radi na način da iz direktorija učitava jednu po jednu sliku. Na učitanoj slici traži se lice te ako iz nekog razloga ne uspije pronaći, program preskače tu sliku i nastavlja dalje. Dio koda koji je zaslužan za traženje lica na slici je prikazan na slici 3.6.

|  |
| --- |
|  |
| Sl. 3.6. Pronalaženje lica na slici |

Pomoću imena slike određuje se pripadna emocija što se zapisuje uz pripadni vektor značajki i služi prilikom testiranja. Poziva se metoda iz projekta servisa koja prima sliku lica, a vraća pripadajući vektor značajki. Zadnje što je potrebno napraviti je zapisati značajke u neku datoteku. Ako je prvi prolazak kroz petlju, onda se prvo dodaje pripadajuće zaglavlje, ovisno o formatu zapisa. Značajke se zapisuju u dvije datoteke. Jedna je formata .tsv (engl. *Tab seperated value*), a druga je .arff. Drugi dio aplikacije, koji služi za stvarnu klasifikaciju i korištenje dobivenih podataka radi na način da se prilikom pokretanja trenira klasifikator koji će se potom koristiti. Ovaj proces traje od prilike sedamdesetak sekundi i pokreće se samo jednom. Nakon treniranja, moguće je koristiti kameru za dohvaćanje slika ili učitati sliku sa nekog drugog mjesta. Program koristi usluge biblioteke servisa za dohvaćanje značajke učitane slike koje kasnije prosljeđuje klasifikatoru. Program servis je dio program koji obavlja većinu posla. Taj dio posla je odvojen u poseban program tako da se mogu koristiti njegove usluge sa različitih mjesta, a prilikom promjene implementacije potrebno je na samo jednom mjestu napraviti izmjenu i sve će raditi. Program dijeli sliku na pet regija lica i na njima primjenjuje Gaborove filtere, slika 3.7. Na svaku novu sliku primjenjuje se PCA radi smanjivanje dimenzionalnosti podataka i vraća se vektor značajki u osnovni program.

|  |
| --- |
|  |
| Sl. 3.7. Primjenjivanje Gaborovih filtera |

Zadnji dio aplikacije služi za klasifikaciju. Prostoje dvije klase od kojih jedna služi za treniranje klasifikatora, a druga je napravljena kao jedinstveni objekt (engl. *Singleton*) i služi za klasifikaciju. Jedinstveni objekt je način pisanja koda tako da se iz jedne klase može instancirati samo jedan objekt i on se onda koristi kroz cijeli program. Takav je način pisanja koda dobar za probleme tipa generatora različitih brojeva jer će se uvijek koristi isti objekt za generiranje koji će davati svaki put druge brojeve, a u slučaju da postoji više objekata tog tipa moglo bi doći do slučaja kada bi oba dali isti broj. Način funkcioniranja nalazi se na slici 3.8.

|  |
| --- |
|  |
| Sl. 3.8. Način rada jedinstvenog objekta |

Dio programa zaslužan za treniranje i testiranje klasifikatora nalazi se na slici 3.9. Koristi se SVM klasifikator iz Weke. Treniranje se odvija na način da se podatci podjele na k dijelova. U svakom prolasku se samo jedan dio koristi za testiranje, a svi ostali za trening. Takav se algoritam naziva K-unakrsna provjera (engl. *K-fld cross validation*) čiji je rad prikazan na slici 3.10.

|  |
| --- |
|  |
| Sl. 3.9. Dio koda za treniranje klasifikatora |

|  |
| --- |
|  |
| Sl. 3.10. K-unakrsna provjera |

* 1. Prikaz izgleda, rad i uporaba rješenja

Da bi se aplikacija mogla koristiti prvo je potrebno stvoriti set vektora značajki iz odgovarajuće baze podataka. Kako bi program koji služi stvaranje baze vektora značajki radio, potrebno je bazu podataka imena „KDEF – DATABASE“ imati na radnoj površini. Svi podatci nastali spremaju se u „Data“ direktorij i biti će potrebni za rad programa. Izgled programa koji stvara bazu vektor značajki prikazan je na slici 3.11.

|  |
| --- |
|  |
| Sl. 3.11. Izgled programa za stvaranje baze vektora značajki |

Izgleda tek pokrenutog programa nalazi se na slici 3.12.

|  |
| --- |
|  |
| Sl. 3.12. Izgled tek pokrenutog programa |

Sa slike 3.12. vide se mogućnosti programa. Moguće je učitati sliku s diska i na njoj izvršiti klasifikaciju pomoću gumba „Učitaj sliku“. Prilikom prve klasifikacije potrebno je pričekati sedamdesetak sekundi da se klasifikator istrenira. Taj proces obavlja se samo jednom. Također je moguće odabrati koji se klasifikator želi koristiti prilikom prepoznavanja lica i vrijeme okidanja za automatsko prepoznavanje. Zadnja opcija testiranja je na drugoj bazi podataka (predefinirana baza koju je potrebno imati na radnoj površini). O bazi podataka za testiranje više u dijelu testiranja. Na slici 3.13. prikazan je izgled programa prilikom prve klasifikacije gdje se prikaže dodatan prozor koji izvještava korisnika da treba pričekati dok se klasifikator ne istrenira.

|  |
| --- |
|  |
| Sl. 3.13. Izgled programa prilikom prve klasifikacije |

Nakon što se treniranje klasifikatora završi, pojavljuje se rezultat u obliku teksta koji govori koja je emocija prepoznata na slici, slika 3.14.

|  |
| --- |
|  |
| Sl. 3.14. Rezultat klasifikacije |

Gumb „Info“ otvara novi prozor koji daje informacije o klasifikatoru. Prikazana je sveukupna preciznost klasifikacije, matrica pogrešaka i preciznost klasifikacije po pojedinoj emociji, slika 3.15.

|  |
| --- |
|  |
| Sl. 3.15. Informacije o klasifikatoru |

Jedna od opcija je klasifikacija slike sa kamere uređaja. Pomoću padajućeg izbornika odabire se kamera koja se želi koristiti i pritiskom na gumb „Pokreni“ kamera se pokreće i prikazuje sliku u sredini programa. Klasifikacija se može vršiti „ručno“ ili „automatski“ Ručna klasifikacija odnosi se na to da korisnik pritisne gumb „Prepoznaj“ i u tome trenutku se uzima trenutna slika i šalje na klasifikaciju. Automatska klasifikacija odnosi se na to da će program svaki interval vremena uzimati trenutnu sliku i slati je na klasifikaciju te vraćati rezultat. Primjer korištenja kamere prilikom dohvaćanja slike za klasifikaciju prikazan je na slici 3.16.

|  |
| --- |
|  |
| Sl. 3.16. Klasifikacije pomoću slike sa kamere |

Slika 3.17. prikazuje rezultate testiranja klasifikatora na drugoj bazi podataka. Prikazana je matrica pogrešaka i ukupna preciznost klasifikacije.

|  |
| --- |
|  |
| Sl. 3.17. Testiranje klasifikacije na drugoj bazi podataka |

* 1. Analiza ponašanja i performansi

U ovom poglavlju prikazani su rezultati uspješnosti klasifikacije u poznatim okolnostima i sa nepoznatim podatcima. Jedan od parametara koji se uzima u obzir je vrijeme potrebno za treniranje i testiranje klasifikatora. Uz samo jedan klasifikator i 1856 značajki, ne uključujući oznaku klase, vrijeme potrebno za treniranje i testiranje klasifikatora iznosi oko 70 sekundi. Program se izvodio na računalu čije su osnovne specifikacije dane u tablici 3.1. Testiranje klasifikatora vršilo se k-unakrsnom provjerom gdje je broj ponavljanja bio postavljen na deset.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | Procesor | Intel(R) Core(TM) i5-5200U CPU @ 2,20GHz, 2195 MHz | | Broj stvarnih jezgri | 2 | | Broj logičkih jezgri | 4 | | Radna memorija | 6 GB DDR3 L | | Tvrdi disk | 1 Disk - WDC WD10JPVX-22JC3T0 (931 GB) | | Matična ploča | Acer Aspire E5-573G | |
| Tablica 3.1. Osnovne specifikacije računala na kojoj je provedena analiza performansi |

Prednosti ovog načina pristupa problemu su u tome da se koristi jedan klasifikator. Kada bi se koristio po jedan klasifikator za svaku akcijsku jedinicu, vrijeme potrebno za testiranje i treniranje svih tih klasifikatora bi bilo značajno veće. Uz manji broj klasifikatora i samo programsko rješenje je puno jednostavnije i stoga lakše za održavati i nadograđivati, ali dolazi s određenim žrtvovanjem preciznosti klasifikacije.

Analiza rezultata na trening podatcima

Što se tiče prosječne preciznosti klasifikacije na trening podatcima, ona iznosi 82,06%. Prilikom treninga korišteno je 797 različitih vektora značajki koji predstavljaju različite izraze lica. Neke emocije su jako dobro prepoznate, a neke su malo slabije što je mana korištenja jednog klasifikatora jer su za neke emocije slični izrazi lica i dosta ih je teže prepoznati nego druge. Uspješnost klasifikacije po pojedinoj emociji dana je u tablici 3.2.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | Emocija | Uspješnost klasifikacije [%] | | Strah | 71,68 | | Srdžba | 75,86 | | Gađenje | 88,79 | | Radost | 96,49 | | Neutralno | 75,38 | | Tuga | 77,36 | | Iznenađenje | 90,09 | |
| Tablica 3.2. Uspješnost klasifikacije po pojedinoj emociji |

Iz tablice 3.2. se vidi da su emocije: strah, srdžba, tuga i neutralno ispod prosjeka po uspješnosti, a ostale odstupaju od njih za minimalno deset posto. To je možda iz razloga jer izrazi lica za te emocije odstupaju od drugih, a i ljudi ih lakše prepoznaju. S obzirom da se strojno učenje i način klasifikacije nastoji što više približiti načinu rada čovjeka, ovi rezultati imaju smisla. Cijela matrica pogrešaka s detaljima klasifikacije prikazana je u tablici 3.3.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | |  | Strah | Srdžba | Gađenje | Radost | Neutralno | Tuga | Iznenađenje | | Strah | 81 | 5 | 2 | 2 | 7 | 8 | 9 | | Srdžba | 7 | 88 | 8 | 1 | 6 | 4 | 0 | | Gađenje | 5 | 9 | 95 | 1 | 1 | 3 | 0 | | Radost | 2 | 0 | 0 | 110 | 3 | 0 | 0 | | Neutralno | 3 | 4 | 0 | 0 | 98 | 9 | 1 | | Tuga | 9 | 10 | 2 | 0 | 10 | 82 | 1 | | Iznenađenje | 6 | 0 | 0 | 0 | 5 | 0 | 100 | |
| Tablica 3.3. Matrica pogrešaka |

Iz matrice pogrešaka vidi se da je srdžba najviše miješana sa tugom i strahom, a neutralno s tugom. Tuga je najviše zamijenjena sa neutralnim, a potom sa strahom, dok je iznenađenje daleko najviše mijenjano sa strahom. Najbolje klasificirana emocija je radost koja skoro pa i nije pogrešno klasificirana.

Analiza rezultata na nepoznatim podatcima

Kao testni podatci korištene su slike iz Cohn-Kanade baze podataka [21], [22]. Primjer slika nalazi se na slici 3.18. Baza sadrži frontalne slike lica sa emocijama. Za svaku emociju postoji više slika, to jest od neke neutralne emocije pa postupno kako dolazi do zadane emocije.

|  |
| --- |
|  |
| Sl. 3.18. primjer slika iz Cohn-Kanade baze podataka (©Jeffrey Cohn) |

Prilikom testiranja korišteno je 299 slika lica. Prosječna točnost klasifikacije iznosi 66,22%, a točnost klasifikacije po emociji dana je u tablici 3.4.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | Emocija | Uspješnost klasifikacije [%] | | Strah | 26,47 | | Srdžba | 48,78 | | Gađenje | 90,00 | | Radost | 91,67 | | Neutralno | 0,00 | | Tuga | 55,17 | | Iznenađenje | 100,00 | |
| Tablica 3.4. Uspješnost klasifikacije po pojedinoj emociji |

Najlošije prepoznata emocija je strah, a potom slijede srdžba i tuga što se podudara s rezultatima testiranja koji su provedeni na trening podatcima. Ostale tri emocije su jako dobro prepoznate, a iznenađenje je prepoznato sa čak 100%. Neutralnih izraza nije bilo u bazi (u završnom okviru skupa slika od kojih svaki okvir predstavlja postupno dolaženje do određene emocije). Ako bi se izuzeo strah, rezultati su dosta dobri. Matrica pogrešaka prikazana je u tablici 3.5 i prikazuje detaljniji prikaz klasifikacije u smislu koja emocija je koliko puta klasificirana ispravno, a koliko puta kao neka druga (i koja druga).

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | |  | Strah | Srdžba | Gađenje | Radost | Neutralno | Tuga | Iznenađenje | | Strah | 18 | 1 | 0 | 3 | 1 | 2 | 0 | | Srdžba | 12 | 20 | 2 | 2 | 2 | 7 | 0 | | Gađenje | 19 | 15 | 18 | 1 | 2 | 3 | 0 | | Radost | 1 | 0 | 0 | 66 | 0 | 1 | 0 | | Neutralno | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | Tuga | 5 | 4 | 0 | 0 | 2 | 16 | 0 | | Iznenađenje | 13 | 1 | 0 | 0 | 2 | 0 | 60 | |
| Tablica 3.5. Matrica pogrešaka |

* 1. Osvrt i mogućnost unaprjeđenja

Prilikom prepoznavanja emocija koristili su se Gaborovi filteri u kombinaciji s PCA za smanjivanje dimenzionalnosti, a kao klasifikator SVM. Koristi se jedna baza podataka za treniranje klasifikatora sadosta kvalitetnim slikama koje su napravljene u kontroliranim uvjetima te su sve emocije odglumljene. Program je rađen da prepoznaje emocije na osnovu frontalnih slika lica. Rezultati klasifikacije na trening podatcima su dosta visoki, a na drugim bazama podataka oko 20% manji. Preciznost klasifikacije srozavaju određene emocije koje program jako teško prepoznaje, a to su strah, srdžba i tuga. Strah je daleko najlošije prepoznata emocija. Ako bi se samo preciznost klasifikacije straha dovela na razinu druge najlošije prepoznate emocije (srdžba, 48,78%) tada bi se ukupna preciznost popravila za više od 7% i bila bi 73.53%. Ako bi se zanemarilo i prepoznavanje neutralne emocije, koje ni nema, tada bi se ukupna preciznost popravila za skoro 10% i bila bi 76,05%. Glavni problem i najveća mogućnost unaprjeđivanja preciznosti bi bila traženje rješenja za veću preciznost tri najlošije prepoznate emocije. To bi se moglo raditi promjenom ili unaprjeđenjem kvalitete značajki koji se dobivaju iz slika. Još jedan način unaprjeđenja cjelokupnog rješenja bi bio korištenje više različitih baza slika prilikom konstruiranja klasifikatora koji bi bili u različitim okruženjima i različite kvalitete što bi dalo realniju situaciju u kojoj bi se primjenjivala aplikacija. Ako bi se koristile kamere s korisničkih računala tu kvaliteta može varirati od jako niske pa do jako visoke te bi u tom slučaju pomogle različite baze podataka sa različitim kvalitetama. Ako bi se stvarno htjela poboljšati preciznost te se koristiti jedna ovakva aplikacija u poslovnom svijetu, bilo bi potrebno obratiti pažnju i na druge aspekte ljudske prirode preko kojih se mogu prepoznati emocije. Na primjer način govora često može odati o kojoj se emociji radi ili barem suziti izbor i pomoći pri klasifikaciji. Pokreti tijela bi se trebali pratiti, a na kraju i kontekst u kojoj se osoba nalazi uvelike pomažu prilikom prepoznavanja emocija. Kada bi se svi ti aspekti objedinili i koristili za prepoznavanje emocija, tada bi se takvo rješenje moglo koristiti za neka ozbiljnija rješenja. Na kraju i uz sve to, neka određena pogreška je neizbježna.

1. ZAKLJUČAK

Prepoznavanje emocija osoba je zamršen proces koji je često kompliciran za ljude, a kamoli za računala. Čovjek prepoznaje emocije na određenoj osobi pomoću jako puno značajki. Na to utječe izraz lica, ton, kontekst, prijašnja iskustva s određenom osobom ili kulturom i uz sve to često se zna pogriješiti. Ovaj se rad bazira samo na slikama lica što je dovoljno za neke privatne upotrebe i za prepoznavanje glumljenih emocija koji se jako manifestiraju na licu. S obzirom na to da se koriste samo slike, rezultati su zadovoljavajući. Rezultati variraju za određene emocije. Neke je lakše prepoznati na licu, a neke teže. Sva testiranja i treniranja rađena su samo s glumljenim emocijama te bi bila dosta veća pogreška kada bi se pokušavale prepoznati stvarne emocije u nekoj realnoj situaciji. Ovakvo rješenje bi se moglo koristiti jedino na mjestima gdje bi se emocije jako izražavale na licu, a to je prilikom gledanja nekih videa ili slika koji su ili izrazito šaljivi, tužni ili imaju zastrašujuće scene. Na takvim mjestima bi se moglo koristiti za mjerenje uspješnosti neke aktivnosti (na primjer: nasmijavanje ljudi, pokušaja preplašivanja).

1. LITERATURA

[1] C. Darwin, P. Ekman, i P. Prodger, The expression of the emotions in man and animals, Oxford University Press, USA, 1998.

[2] Y. L. Tian, T. Kanade i J. F. Cohn, Facial expression analysis. In Handbook of face recognition (pp. 247-275). Springer, New York, 2005.

[3] Johnson-Laird, P. Nicholas, and K. Oatley, The language of emotions: An analysis of a semantic field, Cognition and emotion, Vol. 3, No. 2, pp. 81-123, 1989.

[4] <http://www.humintell.com/2010/06/the-seven-basic-emotions-do-you-know-them/>, pristupljeno 26.11.2016.

[5] D. Lundqvist, A. Flykt, & A. Öhman, The Karolinska Directed Emotional Faces - KDEF, CD ROM from Department of Clinical Neuroscience, Psychology section, Karolinska Institutet, ISBN 91-630-7164-9, 1998

[6]S. Theodoridis i K. Koutroumbas, Pattern Recognition (Fourth Edition), Elsevier Inc., SAD, 2009.

[7]M. N. Murty i V. S. Devi, Pattern Recognition An Algorithmic Approach, Springer, SAD, 2011.

[8] P. Viola, M. Jones, Rapid object detection using boosted cascade of simple features, In Computer Vision and Pattern Recognition, Proceedings of the 2001 IEEE Computer Society Conference, Vol. 1, pp. I-511-I-518, 2001.

[9] Y. Freund & R. E. Schapire, A desicion-theoretic generalization of on-line learning and an application to boosting. In European conference on computational learning theory, pp. 23-37 Springer Berlin Heidelberg, 1995.

[10] S. Padmavathi, P. Prem i D. Praveenn, Locating Fabric Defects Using Gabor Filters, International Jurnal of Scientific Research Engineering & Technology, ISSN 2278 – 0882, Vol. 2 Issue 8, pp. 472-478, Studeni 2013.

[11] S. Wold, K. Esbensen, and P. Geladi, Principal component analysis, Chemometrics and intelligent laboratory systems, Elsevier Science Publishers B.V, pp. 37-52, Nizozemska, 1987.

[12] <http://www.affectiva.com/>, pristupljeno 26.11.2016.

[13] <http://emovu.com/>, pristupljeno 26.11.2016.

[14] <https://www.kairos.com/>, pristupljeno 26.11.2016.

[15] <https://www.microsoft.com/cognitive-services/>, pristupljeno 26.11.2016.

[16] <http://opencv.org/>, pristupljeno 11.12.2016.

[17] <http://www.emgu.com/>, pristupljeno 11.12.2016.

[18] <http://accord-framework.net/>, pristupljeno 11.12.2016.

[19] <http://www.aforgenet.com/framework/>, pristupljeno 11.12.2016.

[20] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann i I. H. Witten, The WEKA Data Mining Software: An Update, SIGKDD Explorations, Vol. 11, pp. 10-18, 2009

[21] Kanade, T., Cohn, J. F., & Tian, Y. (2000). Comprehensive database for facial expression analysis. Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition (FG'00), Grenoble, France, 46-53.

[22] Lucey, P., Cohn, J. F., Kanade, T., Saragih, J., Ambadar, Z., & Matthews, I. (2010). The Extended Cohn-Kanade Dataset (CK+): A complete expression dataset for action unit and emotion-specified expression. Proceedings of the Third International Workshop on CVPR for Human Communicative Behavior Analysis (CVPR4HB 2010), San Francisco, USA, 94-101.

1. SAŽETAK

U radu je implementirana metoda za prepoznavanje emocija na slikama lica. Korišteni su Gaborovi filteri kako bi se dobile crte na licu i potom PCA kako bi se smanjila dimenzionalnost. Kao klasifikator koristi se SVM. U teorijskom dijelu objašnjeni su pojmovi kao što su emocija i afekt. Opisan je klasifikator i korištene metode. Praktični dio se sastoji od programa za dohvaćanje značajki iz baze slike i od dijela koji koristi te značajke kako bi se istrenirao klasifikator te kasnije koristio prilikom prepoznavanja emocija, što s korisnikove kamere, što sa učitane slike. Ovakvo rješenje je dovoljno dobro za korištenje u osobne svrhe.

**Ključne riječi:** Emocije, Gaborovi filteri, PCA, SVM, izrazi lica

**ABSTRACT**

A method for recognizing emotions in images of faces is implemented in the paper. Gabor filters are used in order to extract facial features and then PCA afterwards is applied in order to reduce dimensionality. SVM is used as a classifier. In the theoretical part, terms such as emotion and affect are explained. The classifier is described, as well as the applied methods. The practical part consists of an application which extracts features from a base of images and from a part which is using those features to train the classifier in order to use it later on when recognizing emotions in an image from a personal camera or from an uploaded one. This sort of solution is good enough for personal use.

**Key words:** Emotions, Gabor filters, PCA, SVM, facial expression

1. ŽIVOTOPIS

Josip Baketarić, rođen u Požegi 10. kolovoza 1993. Osnovnu školu završio u Pleternici u razdoblju od 2000. do 2008. godine. 2008. godine upisuje srednju tehničku školu u Požegi, smjer tehničar za računalstvo koju završava 2012. godine. Nakon toga upisuje Elektrotehnički fakultet u Osijeku, preddiplomski studij, smjer računarstvo kojega završava 2015. godine. 2016. godine upisuje diplomski studij na prijašnje spomenutom fakultetu, sada Fakultet elektrotehnike, računarstva i informacijskih tehnologija, smjer računarstvo, izborni blok programsko inženjerstvo kojega trenutno pohađa.

1. PRILOZI

Na CD-u:

1. Diplomski rad „Klasifikacija emocija na osnovi slike lica.docx“
2. Diplomski rad „Klasifikacija emocija na osnovi slike lica.pdf“
3. Izvorni kod programa
4. Instalacija programa