

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 6354

Sustav za raspoznavanje znakovnog jezika

Nikola Tomažin

Zagreb, lipanj 2019.

Zagreb, 14. ožujka 2019.

ZAVRŠNI ZADATAK br. 6354

Pristupnik: **Nikola Tomažin (0036498789)**
Studij: Računarstvo
Modul: Računalno inženjerstvo

Zadatak: **Sustav za raspoznavanje znakovnog jezika**

Opis zadatka:

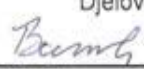
Jedna od zanimljivih primjena računalnog vida je automatsko raspoznavanje znakovnog jezika, čime bi se olakšala komunikacija između gluhoonijemih osoba i osoba zdravog sluha. U okviru ovog završnog rada potrebno je proučiti značajnije pristupe računalnom raspoznavanju znakovnog jezika opisane u literaturi, s naglaskom na pristupe temeljene na dubokom učenju te predložiti i programski ostvariti takav sustav temeljen na prikladnom modelu. Pripremiti bazu slika za učenje i ispitivanje sustava, analizirati ponašanje implementiranog sustava te prikazati i ocijeniti ostvarene rezultate. Radu priložiti izvorni i izvršni kod razvijenih postupaka, ispitne slike i rezultate, uz potrebna objašnjenja i dokumentaciju te navesti korištenu literaturu.

Zadatak uručen pristupniku: 15. ožujka 2019.
Rok za predaju rada: 14. lipnja 2019.


Mentor:


Izv. prof. dr. sc. Tomislav Hrkać

Djelovođa:


Prof. dr. sc. Danko Basch

Predsjednik odbora za
završni rad modula:


Prof. dr. sc. Mario Kovač

Sadržaj

1. Uvod	1
2. Podaci	4
2.1. Prikupljanje podataka	4
2.2. Izdvajanje ruke	5
2.2.1. Histogram boje ruke	5
2.2.2. Oduzimanje pozadine	6
2.3. Nadopunjavanje skupa podataka	7
3. Implementacija sustava	10
3.1. Općenito o strojnom učenju	10
3.1.1. Neuron	10
3.1.2. Neuronska mreže	12
3.1.3. Konvolucijska neuronska mreža (<i>Convolutional Neural Network - CNN</i>)	13
3.1.4. Povratna neuronska mreža (<i>Recurrent Neural Network – RNN</i>)	14
3.2. Arhitektura mreže	14
3.2.1. Prvi pristup - Konvolucijska neuronska mreža	14
3.2.2. Drugi pristup – 3D konvolucijska neuronska mreža i povratna neuronska mreža(RNN)	16
3.3. Treniranje mreže	17
4. Rezultati	19
4.1. Problemi	22
5. Instalacija i upute za korištenje	24
5.1. Potrebne instalacije (<i>Requirements</i>)	24
5.2. Pokretanje	24
5.2.1. Dodavanje geste	25
5.2.2. Treniranje mreže	26
5.2.3. Prepoznavanje geste	26
6. Zaključak	28
Literatura	29

1. Uvod

U Republici Hrvatskoj službeni je jezik hrvatski, a službeno pismo latinica. Gluhih i nagluhih osoba u Republici Hrvatskoj uvijek je bilo, no njihov način jezika, njihov način komunikacije (sa zajednicom i društvom) u zakonima pojavljuje se tek 2015. godine. Hrvatski znakovni jezik (HZJ) sustav je vizualnih znakova koji, uz pomoć posebnog položaja (oblika šake), orijentacije, položaja i smjera pokreta ruke, tvore koncept odnosno smisao slova i riječi. Koristi se najčešće u obiteljima u kojima ima gluhih, zajednicama gluhih te, nešto manje u školama za gluhe.

Postoje dvije vrste hrvatske znakovne abecede:

- Jednoručna abeceda hrvatskog znakovnog jezika koja je zapravo verzija američke (internacionalne) jednoručne abecede. Proširena je onim slovima koja se koriste samo u našem jeziku. U osnovi, oponaša mala tiskana slova. Uglavnom se koristi u školama za gluha djecu jer se tako na učinkovit način prenose riječi i informacije



Slika 1: Primjer jednoručne abecede hrvatskog znakovnog jezika

Ručne abecede (Šarac Kuhn et al. 2006: 56)

- Dvoručna abeceda hrvatskog znakovnog jezika koja je posebno određena položajima prstiju obje ruku, a oponaša velika tiskana slova hrvatske abecede. Dvoručna abeceda ima dugu tradiciju u zajednici gluhih. U dvoručnoj se abecedi koriste veliki pokreti ruku pa je moguće da se dvoručnom abecedom koriste i gluhonijeme osobe sa značajnim ostatkom vida.



Slika 2: Primjer dvoručne abecede hrvatskog znakovnog jezika

Ručne abecede (Šarac Kuhn et al. 2006: 56)

Cilj ovoga završnoga rada napraviti je sustav koji bi prepoznavao geste znakovne abecede (jednoručne i dvoručne) te još neke geste/znakove u stvarnom vremenu (engl. *real-time*) te omogućio komunikaciju gluhonijeme osobe s nekime tko primjerice ne zna znakovni jezik/abecedu. Sustav je širok pojam, ovdje on obuhvaća niz procesa i radnji, počevši od prikupljanja podataka(*dataseta*), obrada tih podataka (izdvajanje bitnoga), treniranje i testiranje neuronske mreže na problemu te konačni rezultat koji preko kamere interpretira znakove koje mu pokazujemo te nam “prevodi” HZJ u slova abecede.

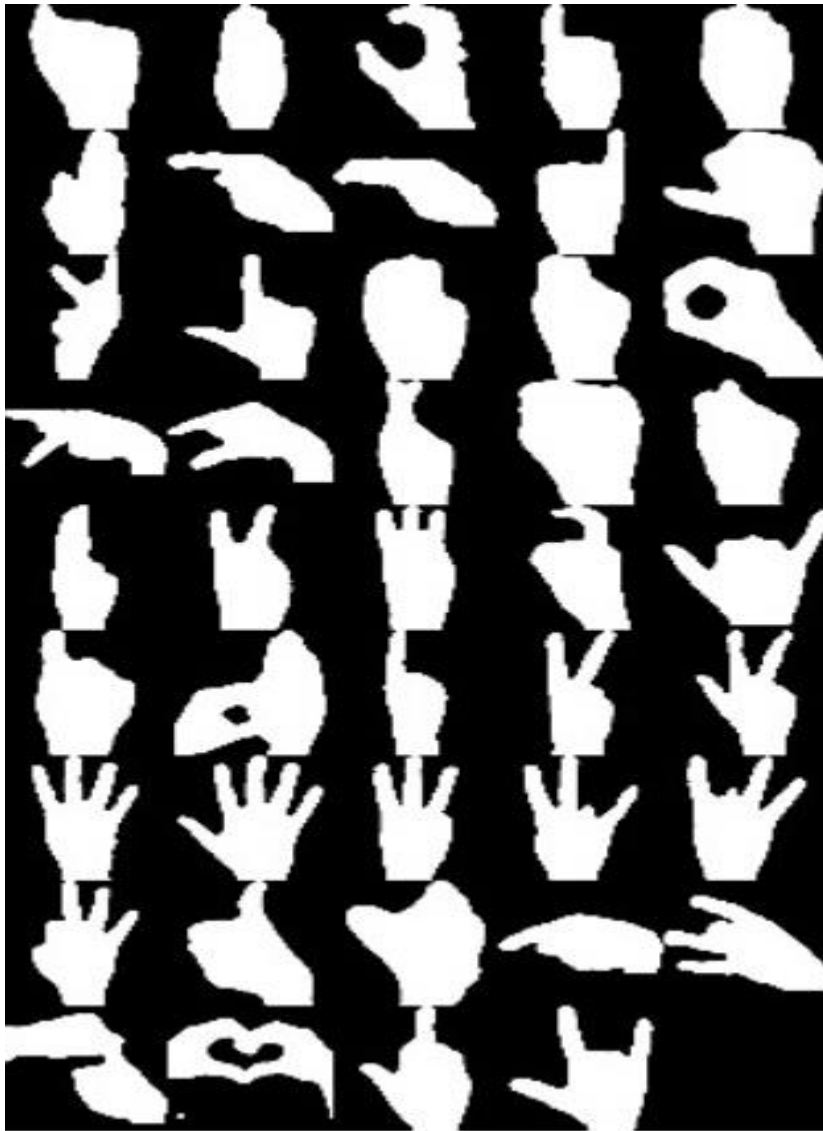
Prvi dio rada obuhvaća prikupljanje podataka, obradu istih te način oblikovanja (kako ih pripremit za mrežu). Drugi dio odnosi se na neuronske mreže te teoriju u njihovoj pozadini. Bit će opisana arhitektura konvolucijske neuronske mreže(CNN - *Convolutional Neural Network*) te pokušaj upotrebe arhitekture povratne neuronske mreže(RNN - *Recurrent Neural Network*) za analiziranje dinamičkih pokreta. Treći dio obuhvaća analizu rezultata, probleme kod sličnih slova, probleme dinamičkih slova te instalaciju i uputstva za uporabu.

2. Podaci

Najbitnija stvar kod strojnog učenja je imati dovoljno veliki i smislen skup podataka (engl. *dataset*). Bez kvalitetnih podataka rezultati mogu biti neprecizni te nema koristi od takve mreže. Kod odabira podataka mora se mnogo faktora uzeti u obzir, primjerice veličina i kvaliteta slike, osvjetljenje, uzima li se cijela slika ili dio nje, pozicija/poza objekta kojeg promatramo... Na sreću, postoje već gotovi skupovi podataka koji su dizajnirani za određene zadatke i probleme, kao što su praćenje objekata, detekcija, procjena poze, prepoznavanje akcija itd. Veličina skupova podataka također može biti raznolika, od nekoliko stotina slika, za jednostavne probleme do više stotina tisuća podataka kod kompleksnih i širokih problema.

2.1. Prikupljanje podataka

Pri prvotnom prikupljanju podataka za raspoznavanje znakovnog jezika uzet je skup podataka^[1] koji sadrži 40 različitih gesti. Skup je sadržavao znakove američke jednoručne znakovne abecede, brojeve od 1 do 9 te neke dodatne znakove. Svaka gesta bila je prikazana kroz 2400 slika: 1200 polaznih slika te njihova zrcalna varijanta kako bi se dobila mogućnost prikazivanja geste neovisno gleda li se sa prednje ili stražnje strane što omogućava promatranje pomoću prednje i stražnje kamere. Slike su bile crno-bijele (kontrast je napravljen bijelim znakom šake / dlana na crnoj podlozi), dimenzija 50x50 piksela.



Slika 3: Prikaz skupa podataka

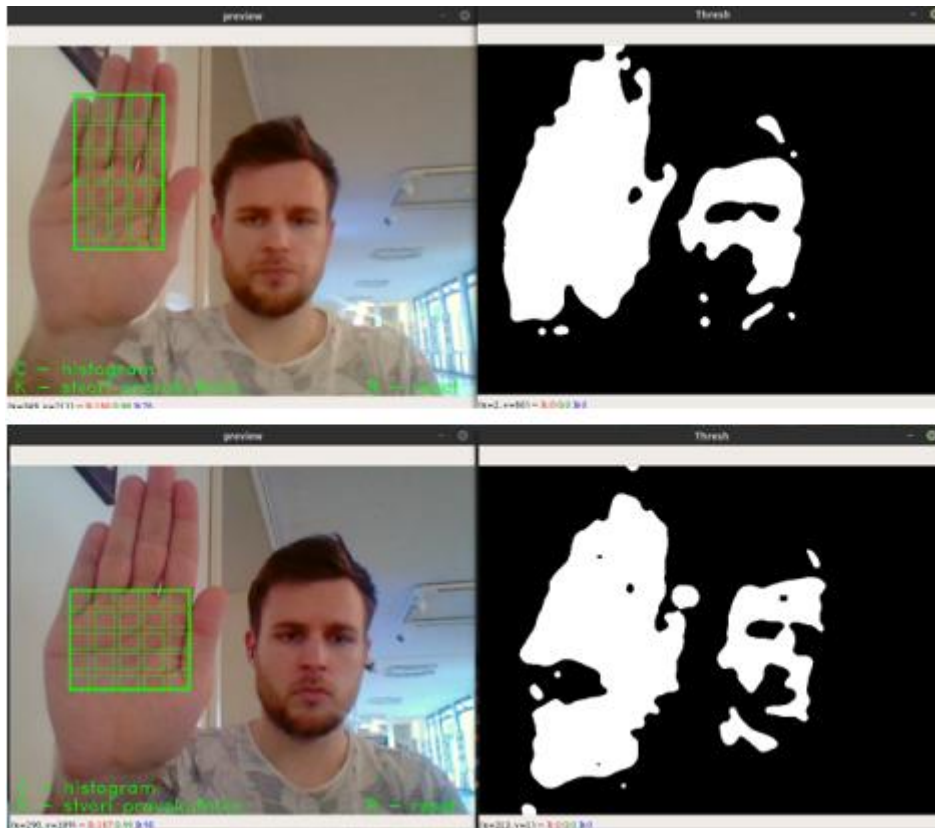
2.2. Izdvajanje ruke

Budući da je skup podataka crno bijela slika dlana/šake, treba tome prilagoditi ulaz (prikaz geste).

2.2.1. Histogram boje ruke

Prvi pristup tom problemu je bila metoda histograma boje ruke(engl. *Hand hist*). U ovom pristupu se prilikom početku snimanja(?) postavi dlan na određen prostor prikaza kamere, te se onda uzme spektar boja koji se nalazi u tom određenom prostoru. Sav spektar boje kože postavi se u bijelo, a sve izvan toga spektra u

crno. Nakon što dobijemo zadovoljavajući histogram možemo ga spremi te dalje koristiti kod prepoznavanja. Problem u ovom pristupu je da uvjeti snimanja moraju biti "savršeni" kako bi se dobio 'čist' histogram. Svaka promjena u osvjetljenju, nastajanje sjene te ne uzimanje u spektar sve nijanse boje kože dovodile su do nepotpunog histograma (Slika 4). Ova metoda nije davala zadovoljavajuće rezultate jer se nije poklapala sa već postojećim skupom podataka, a i nije bila precizna, ovisila je o previše parametara.



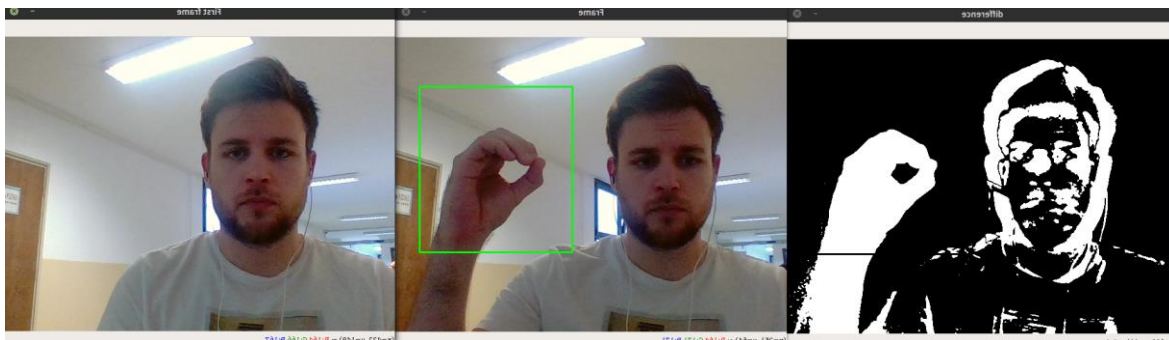
Slika 4: Prikaz dohvata histograma ruke

2.2.2. Oduzimanje pozadine

Drugi pristup je preko "oduzimanja pozadine" (engl. background subtraction). Oduzimanje pozadine je široko korištena metoda koja se uglavnom koristi za dobivanje maske prvog plana (engl. foreground mask), odnosno objekata koji se pomiču u sceni koristeći statične kamere. Sastoji se od dva dijela, od inicijalizacije pozadine te od ažuriranja pozadine. U inicijalizaciji pozadine uzima se početni model pozadine, koji u sebi sadrži samo statične elemente dok se u drugom

koraku svaki kadar (engl. frame) uspoređuje s inicijalnim modelom pozadine te se detektiraju razlike.

U ovom pristupu na ovaj problem inicijalizacija pozadine obavlja se pri paljenju kamere, gdje se uzima prvi kadar, a ažuriranje pozadine se gleda samo na određenom dijelu prikaza kamere, omeđenom pravokutnikom, jer taj dio nam predstavlja ulaz u mrežu, odnosno dio koji uspoređujemo i gledamo. Ova metoda se pokazala zadovoljavajućom jer ne ovisi o vanjskim uvjetima, ovisi jedino o pretpostavci da je kamera statična. Kod dinamične kamere pozadina mora biti jednolična (npr. Jednobojni zid) kako bi ova metoda funkcionirala



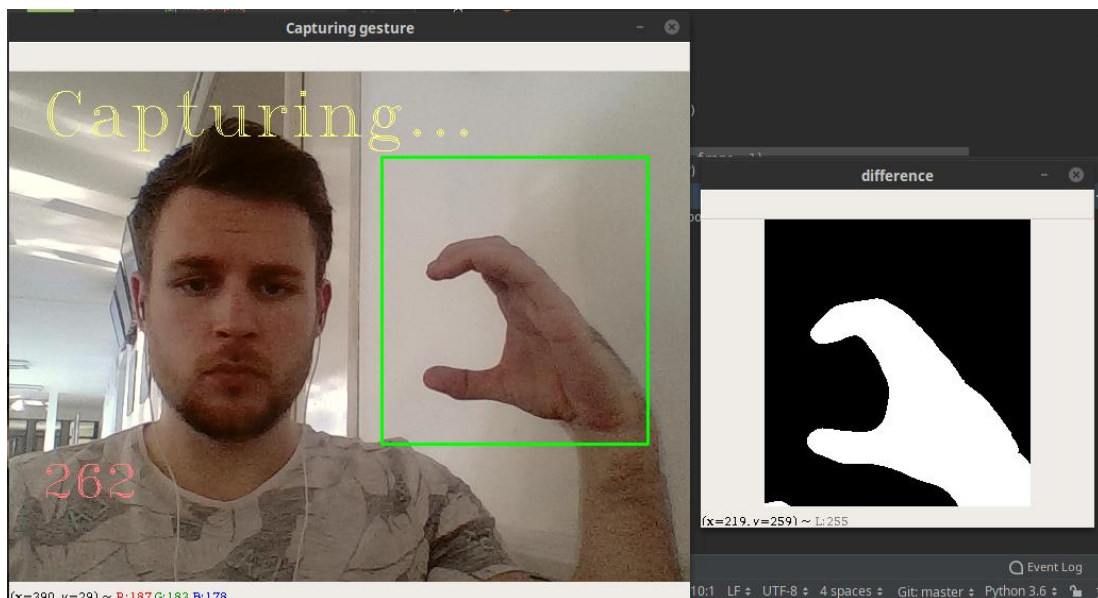
Slika 5: Prikaz metode oduzimanja pozadine

Prvi dio slike predstavlja prvi snimljen kadar, drugi dio predstavlja prikaz kamere i treći dio predstavlja izlaz metode oduzimanja pozadine na temelju prve dvije slike.

2.3. Nadopunjavanje skupa podataka

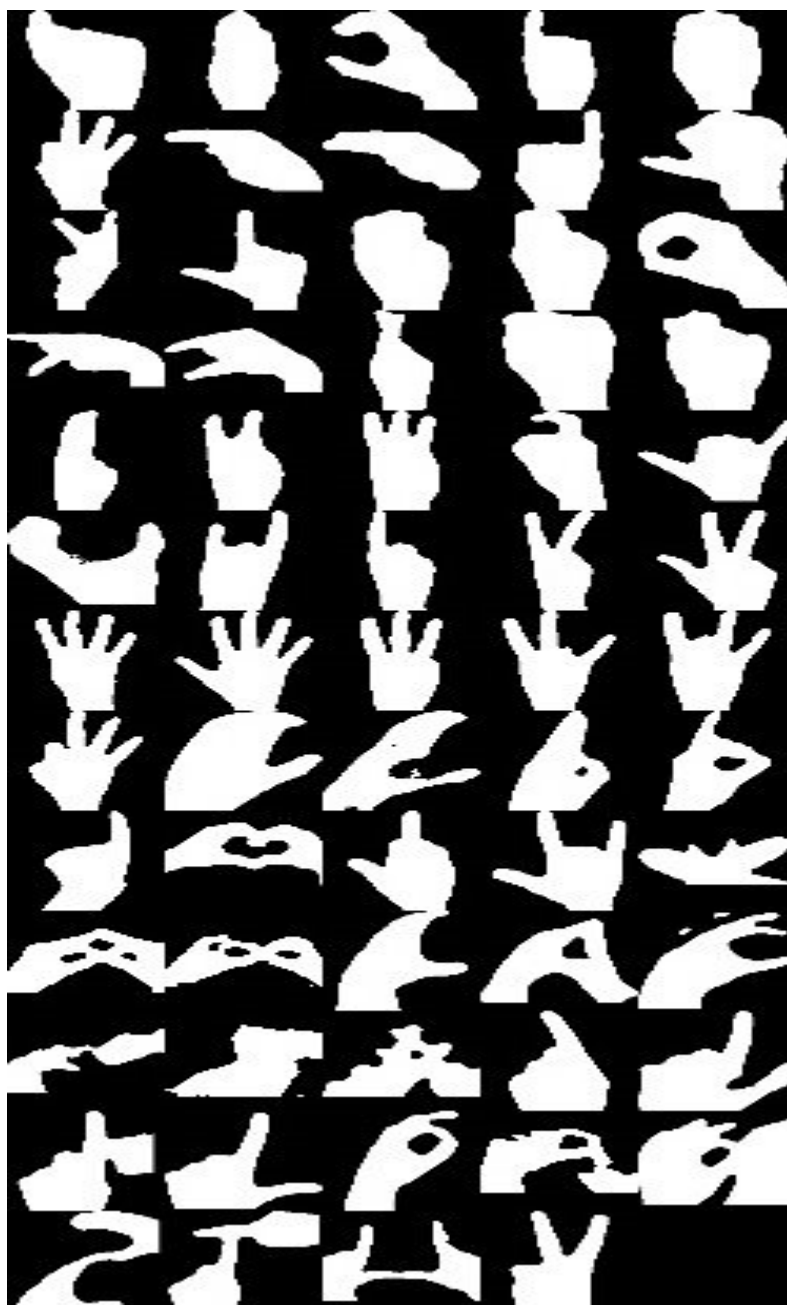
Kako se hrvatska i američka jednoručna abeceda imaju neke razlike te kako postoji i dvoručna hrvatska znakovna abeceda trebalo je povećati skup podataka. Program “add_gesture.py” dopunjen je skup podataka. Program kreira direktorij gdje će se spremati geste te uključuje kameru sa standardnim zelenim pravokutnikom kao pokazivačem koji dio se promatra i sa prozorom koji prikazuje što računalo „vidi”. Pritiskom na tipku „c”, uz uvjet da je površina geste zadovoljavajuća, pokreće se snimanje gesti. Program snima 1200 slika i njegov

napredak se može pratiti u kutu prikaza kamere. Korisnik svakog trenutka može pauzirati/nastaviti snimanje ponovnim pritiskom na tipku „c”, a snimanje se ujedno i pauzira čim je površina geste manja od dozvoljene. Snimanje može potrajati do minute. Nakon snimanja slika se zrcali pomoću metode „flip_images” te se konačno dobije 2400 slika nove geste.



Slika 6: Prikaz nadopunjavanja skupa podataka

Nakon nadopunjavanja, skup podataka trenutno sadrži 64 geste (Slika 7), gdje su sadržane geste jednoručne abecede, dvoručne abecede, brojevi te još neki dodatni znakovi/geste.



Slika 7: Prikaz svih znakova

3. Implementacija sustava

3.1. Općenito o strojnom učenju

„Strojno učenje jest programiranje računala na način da optimiziraju neki kriterij uspješnosti temeljem podatkovnih primjera ili prethodnih iskustava.“ – (Alpaydin 2009).

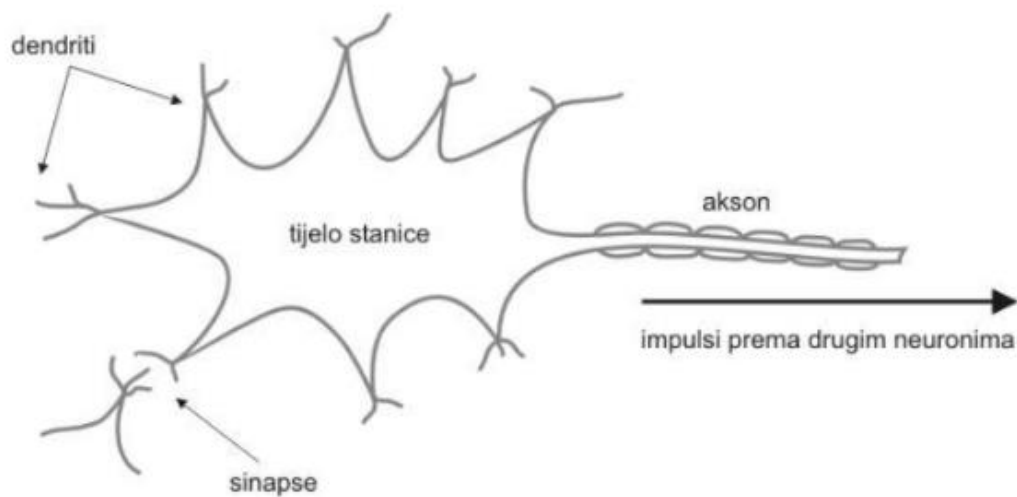
Strojno učenje se koristi iz raznih razloga, kao što su rješavanje složenih problema gdje ne postoji ljudsko znanje o procesu ili ljudi ne mogu dati objašnjenje o procesu (npr. Raspoznavanje govora) u što spadaju i problemi koje nije moguće riješiti na klasičan algoritamski način, kod problema s ogromnim količinama podataka te kod sustava koji se dinamički mijenjaju gdje je potrebna prilagodba. Strojno učenje grana je umjetne inteligencije koja se bavi oblikovanjem algoritama koji svoju učinkovitost poboljšavaju na temelju empirijskih podataka. Strojno učenje jedno je od danas najaktivnijih i najuzbudljivijih područja računarke znanosti, ponajviše zbog brojnih mogućnosti primjene koje se protežu od raspoznavanja uzoraka i dubinske analize podataka do robotike, računalnog vida, bioinformatike i računalne lingvistike.

Strojno učenje ima dva osnovna pristupa: nadzirano učenje (klasifikacija i regresija) i nenadzirano učenje (grupiranje i smanjenje dimenzionalnosti). Jedno od najboljih rješenje za strojno učenje su neuronske mreže.

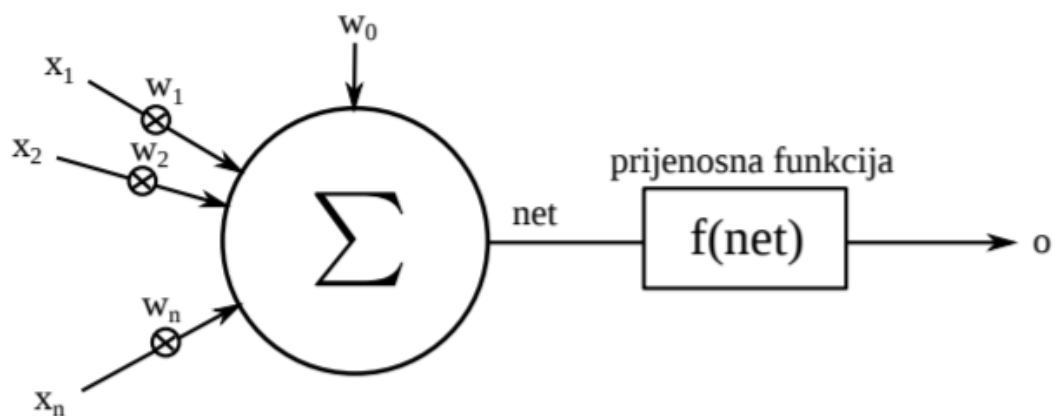
Umjetna neuronska mreža skup je međusobno povezanih jednostavnih procesnih elemenata (neurona) čija se funkcionalnost temelji na biološkom neuronu i koji služe distribuiranoj paralelnoj obradi podataka.

3.1.1. Neuron

Glavna jedinica neuronske mreže je neuron koji oponaša biološki neuron. Biološki neuron (Slika 8) sastoji se od tijela (soma), dendrita, aksona te završnih članaka te je u prosjeku, u ljudskom mozgu, svaki neuron povezan s 1000 do 10000 drugih neurona.



Slika 8: Biološki neuron^[2]



Slika 9: Računalni neuron^[3]

Vrijednost sa svakog ulaza x_i množi se s osjetljivošću (engl. *weight*) tog ulaza w_i i akumulira u tijelu. Ukupnoj sumi dodaje se i pomak w_i (engl. *bias*) te se time definira akumulirana vrijednost „net“.

$$\text{net} = \left(\sum_{i=1}^n x_i w_i \right) + w_0$$

Ta se vrijednost propušta kroz prijenosnu (aktivacijsku) funkciju „f“ čime nastaje izlazna vrijednost „o“.

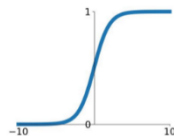
$$o = f(\text{net})$$

Česte prijenosne funkcije su Identitet (ADALINE-neuron), funkcija skoka (TLU-perceptron), sigmoidalna funkcija (sigmoidalni neuron), zglobnica (Rectified Linear Unit, ReLU)... Prijenosna funkcija ovisi o problemu, kod linearnih problema koriste se linearne funkcije, dok se kod nelinearnih problema koriste nelinearne funkcije.

Activation Functions

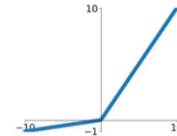
Sigmoid

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



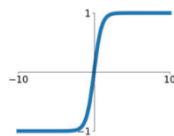
Leaky ReLU

$$\max(0.1x, x)$$



tanh

$$\tanh(x)$$

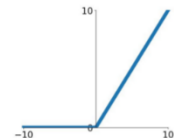


Maxout

$$\max(w_1^T x + b_1, w_2^T x + b_2)$$

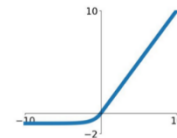
ReLU

$$\max(0, x)$$



ELU

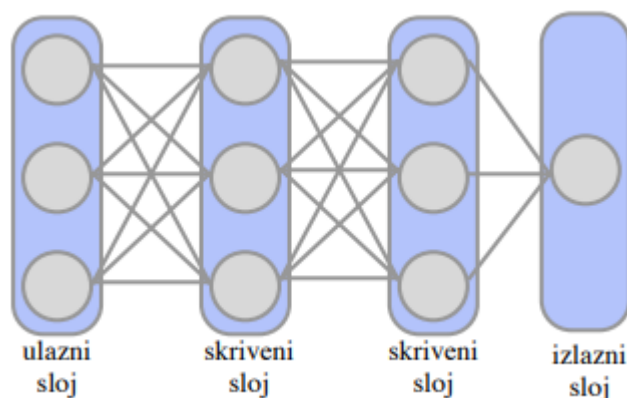
$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



Slika 10: Prijenosne (aktivacijske) funkcije^[4]

3.1.2. Neuronska mreže

Kako bi omogućili modeliranje složenijih odnosa u postupcima klasificiranja te regresije, koristi se više neurona. Višestruki neuroni povezani su u acikličkim grafovima, iako je jedna od najčešćih arhitektura ona višeslojnog neurona u kojima su neuroni organizirani u slojevima. Postoje ulazni i izlazni slojevi i dodatni skriveni slojevi koji povećaju veličinu i složenost mreže.



Slika 11: Model neuronske mreže

Znanje mreže pohranjeno je implicitno u težinama veze između neurona. Te težine je potrebno učiti ili trenirati da bi ispravno okarakterizirale podatke s ulaza mreže. Kada se kaže učenje mreže misli se na iterativno predočavanje ulaznih podataka i eventualno očekivanih vrijednosti na izlazu. Metoda učenja kod koje se uz ulaz predočava i očekivani izlazi naziva se nadzirano učenje.

3.1.3. Konvolucijska neuronska mreža (*Convolutional Neural Network - CNN*)

Konvolucijske neuronske mreže mogu se opisati kao nadogradnja nad neuronskim mrežama. Konvolucijska, kao i obična, neuronska mreža sastoji se od jednog ulaznog, jednog izlaznog te jednog ili više skrivenih slojeva. Specifičnost konvolucijskih neuronskih mreža su konvolucijski slojevi i slojevi sažimanja. . Konvolucijske neuronske mreže najčešće kreću s jednim ili više konvolucijskih slojeva, zatim slijedi sloj sažimanja, pa ponovo konvolucijski sloj i tako nekoliko puta. Mreža najčešće završava s jednim ili više potpuno povezanih slojeva koji služe za klasifikaciju. Arhitektura konvolucijskih neuronskih mreža pokazala se izrazito dobra u radu sa slikama i prepoznavanju značajki s istih^[5] te se zato koristi pri prepoznavanju objekata na slici.

Parametri konvolucijskog sloja su filteri koji sadrže težine koje je potrebno naučiti. Filteri obično imaju malu visinu i širinu, a dubine su jednake kao i ulaz u mrežu. Pri unaprijednom prolazu filter pomičemo s nekim korakom (engl. *stride*) po visini i širini ulaza i računamo skalarni produkt, odnosno konvoluciju između elemenata filtra i ulaza. Operacija konvolucije proizvodi dvodimenzionalne aktivacijske mape koje daju odzive filtra. Za svaki filter dobijemo po jednu aktivacijsku mapu, a izlaz iz mreže dobijemo tako da posložimo aktivacijske mape u dubinu. Konvolucija ima zanimljivo svojstvo ekvivarijantnosti s obzirom na pomak. Intuitivno to možemo tumačiti kao da se konvolucijski filteri “aktiviraju” na značajke koje su nam interesantne, a konvolucijski algoritmi uče na koje značajke 10 se filteri trebaju aktivirati. Izlaz mreže ne ovisi o tome gdje se značajke nalaze već samo o tome jesu li prisutne.

3.1.4. Povratna neuronska mreža (*Recurrent Neural Network – RNN*)

Povratna neuronska mreža može se smatrati neuronskom mrežom s pamćenjem (memorijom). Ideja povratnih neuronskih mreža je koristiti sekvencu informacija. Tradicionalne neuronske mreže pretpostavljaju da su ulazi i izlazi neovisni jedni o drugima, ali kod povratnih neuronskih mreža nije tako, kod njih su međuoavisni. To se ostvaruje dodavanjem skrivenog kratkoročno memorijskog sloja između slojeva neuronske mreže. Tako se u mreži ne gleda samo ulaz nego i izlaz prijašnje iteracije te mreža gleda na njihov međusoban odnos.

3.2. Arhitektura mreže

Ideja mreže je da za ulaz primi sliku, preko korisnikove kamere, a za izlaz vrati koja je to gesta. Ulazna slika je zapravo dvodimenzionalno (2D) polje veličine 50x50 piksela koja sadrži vrijednosti od 0-255 koji označavaju koje je boje taj određen piksel, gdje je 0 potpuno crna, a 255 potpuno bijela boja.

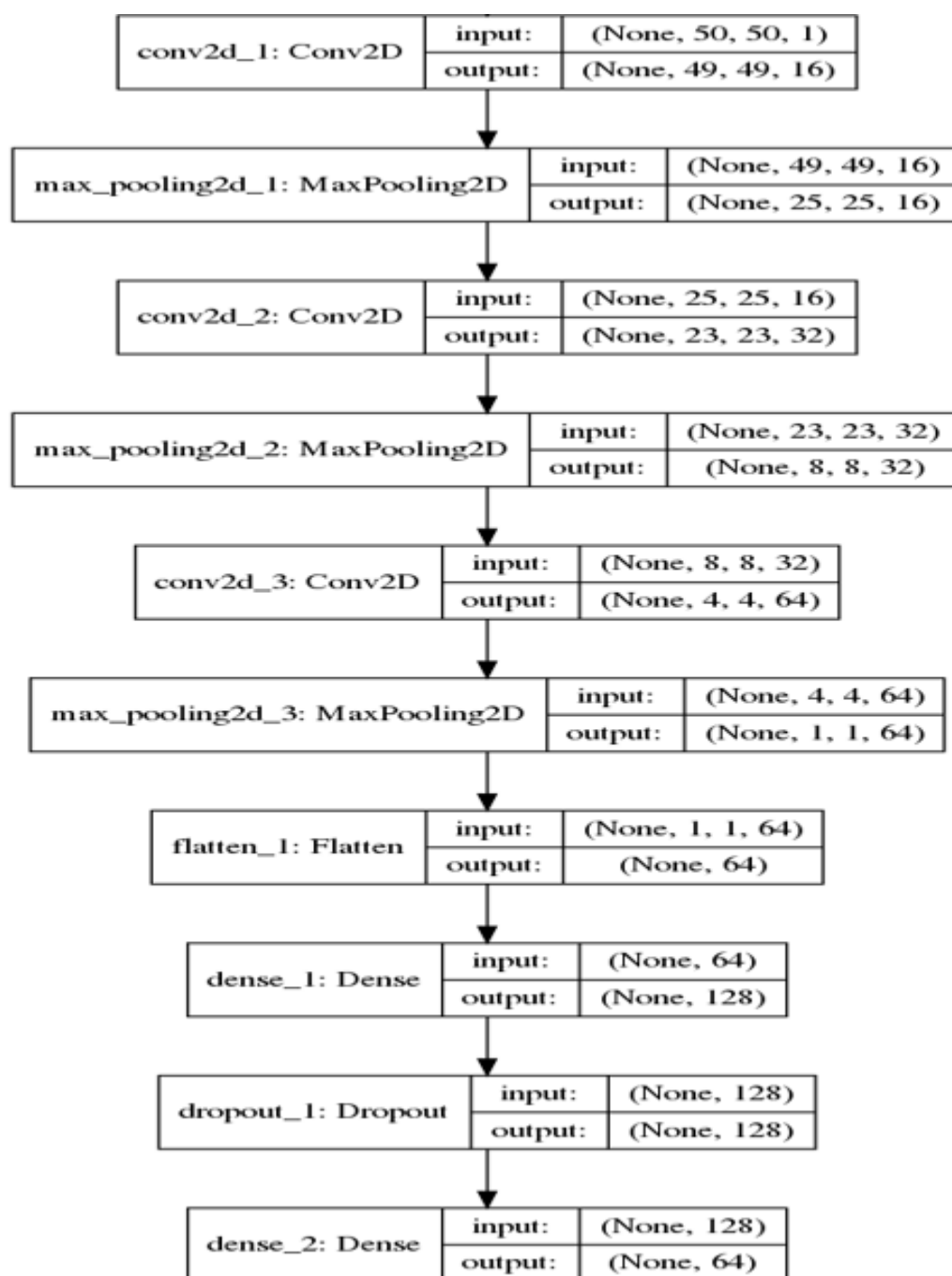
Ovom problemu pristupilo se na dva načina. Prvi pristup bio je preko konvolucijske neuronske mreže dok je drugi preko 3D konvolucijske neuronske mreže i preko povratne neuronske mreže.

3.2.1. Prvi pristup - Konvolucijska neuronska mreža

Mreža se sastoji od 3 dvodimenzionalna konvolucijska sloja (engl. *Convolutional 2D layer*) koji između sebe imaju slojeve sažimanja (engl. *Pooling layer*). Konvolucijski slojevi za aktivacijsku (prijenosnu) funkciju koriste zglobnicu (engl. *ReLU function*) te pokušavaju otkriti neke karakteristike na slikama. Izlaz trećeg sloja sažimanja povezan je sa slojem zaravnjanja (engl. *Flatten layer*) koji dvodimenzionalni izlaz pretvori u jednodimenzionalan niz. Nakon slijedi zbijen sloj (engl. *Dense layer*) koji služi kao potpuno povezan sloj (engl. *Fully connected layer*) te on svaki ulazni neuron povezuje na svaki izlazni neuron. U ovom slučaju povezuje sve ulazne neurone na 128 izlaznih neurona, također koristeći zglobnicu

kao aktivacijsku funkciju, koji su povezani dalje na sloj izbacivanja (engl. *Dropout layer*) koji služi za regulaciju prenaučenosti (engl. *overfitting*) tako da zanemari nasumično odabrane neurone. Na kraju se nalazi još jedan zbijen sloj koji povezuje neurone koji su ostali nakon izbacivanja s konačnim izlaznim slojem (koji sadrži neurona koliko imamo različitih gesti) koristeći „*softmax*” aktivacijsku funkciju koja pretvara izlaze u razdiobu vjerojatnosti.

Ova arhitektura mreže bila je dobra za statičke geste, no nije mogla raspoznavati dinamičke geste.



Slika 12: Arhitektura mreže

3.2.2. Drugi pristup – 3D konvolucijska neuronska mreža i povratna neuronska mreža(RNN)

Dinamičku gestu možemo gledati kao skup od više statičkih gesti. Za taj pristup prvo je uzeta trodimenzionalna (3D) konvolucijska mreža, koja je za razliku od uobičajene konvolucijske mreže koja je primala jednu sliku (zapravo 2D matricu

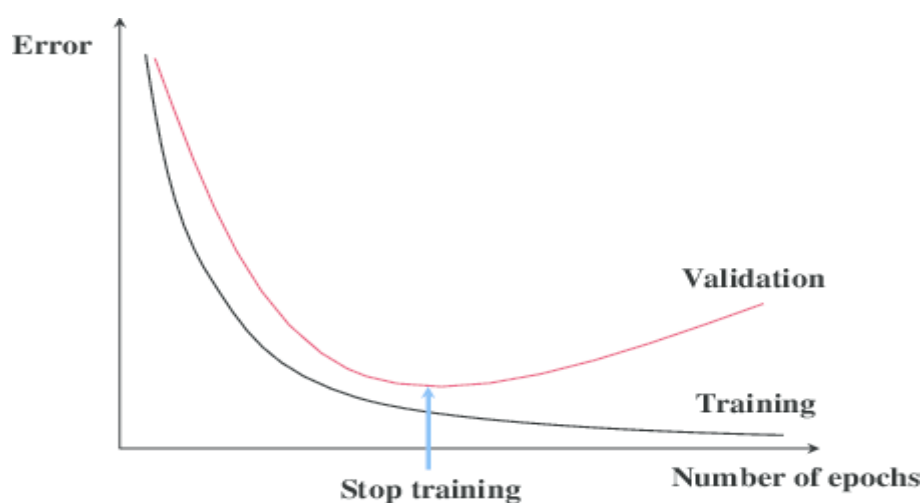
vrijednosti piksela), primala trodimenzionalnu matricu, gdje je treća dimenzija zapravo bila niz slika (2D matrica). Ako se odredi točan broj kadrova geste te se pretpostavi da će se dinamička gesta izvoditi poznatom brzinom može se uzeti slike u određenim kadrovima (primjerice prvom, srednjem i zadnjem kadru izvođenja geste) te ih spojiti u trodimenzionalni ulaz koji bi se mogao interpretirati kao vremenski slijed. Za ovaj pristup potrebna je i manipulacija skupom podataka kako bi se prilagodili ovom načinu obrade.

3.3. Treniranje mreže

Sveukupni skup podataka sadrži 64 klasifikacijske kategorije, gdje svaka kategorija sadrži 2400 slika, što rezultira 153 600 slika sveukupno. Za treniranje je uzeto 5/6 skupa podataka (128 000 slika), za validaciju 1/12 skupa (12 800 slika) i za testiranje 1/12 skupa (12 800).

Za hiperparametri mreže uzeti su broj epoha koji iznosi 50 i veličina serije (engl. *Batch size*) koja iznosi 500. Broj epoha je broj iteracija kroz cijeli skup podataka dok veličina serije predstavlja broj koliko primjeraka algoritam prođe prije nego ažurira vrijednosti težina neurona.

Kako bi se izbjegla prenaučenost uvedena je metoda ranog zaustavljanja (engl. *Early stopping*). Ona zaustavlja treniranje čim se gubitak (engl. *Loss*), odnosno greška, na validacijskom skupu krene povećavati (Slika 13).



Slika 13: Prikaz metode ranog zaustavljanja

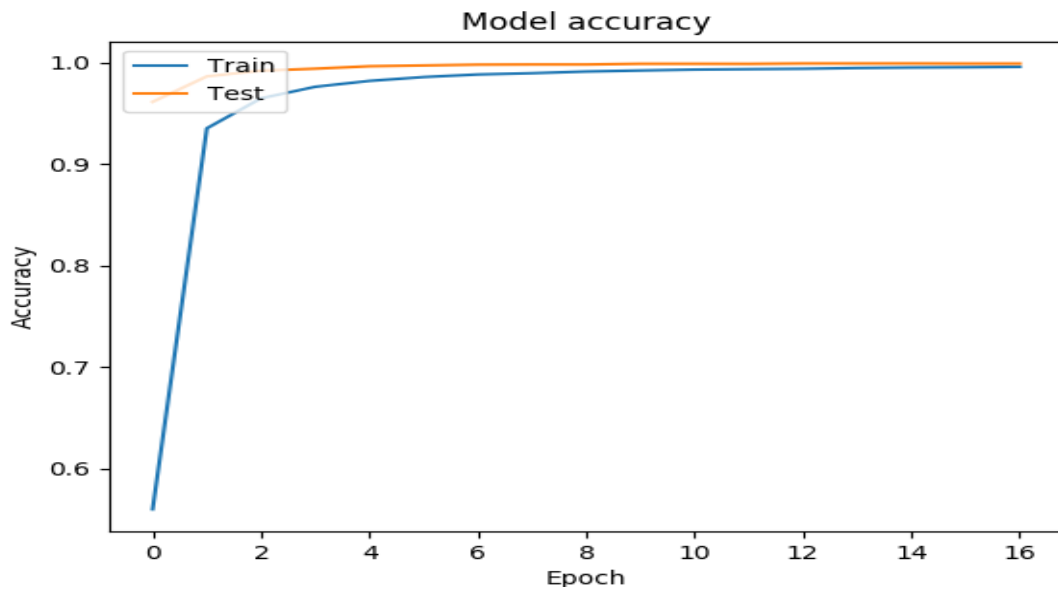
Za funkciju gubitka korištena je funkcija kategorične križne entropije (engl. Categorical Crossentropy) koja se koristi kod klasifikacijskih problema gdje samo jedan izlaz može biti točan.

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 49, 49, 16)	80
max_pooling2d_1 (MaxPooling2)	(None, 25, 25, 16)	0
conv2d_2 (Conv2D)	(None, 23, 23, 32)	4640
max_pooling2d_2 (MaxPooling2)	(None, 8, 8, 32)	0
conv2d_3 (Conv2D)	(None, 4, 4, 64)	51264
max_pooling2d_3 (MaxPooling2)	(None, 1, 1, 64)	0
flatten_1 (Flatten)	(None, 64)	0
dense_1 (Dense)	(None, 128)	8320
dropout_1 (Dropout)	(None, 128)	0
dense_2 (Dense)	(None, 64)	8256
Total params: 72,560		
Trainable params: 72,560		
Non-trainable params: 0		

Slika 14: Parametri o obliku slojeva

4. Rezultati

Budući da sustav raspolaže s jednostavnim crno-bijelim slikama veličine 50x50 piksela već nakon prve epohe treniranja dosegne preciznost od preko 90% (Slika 15).

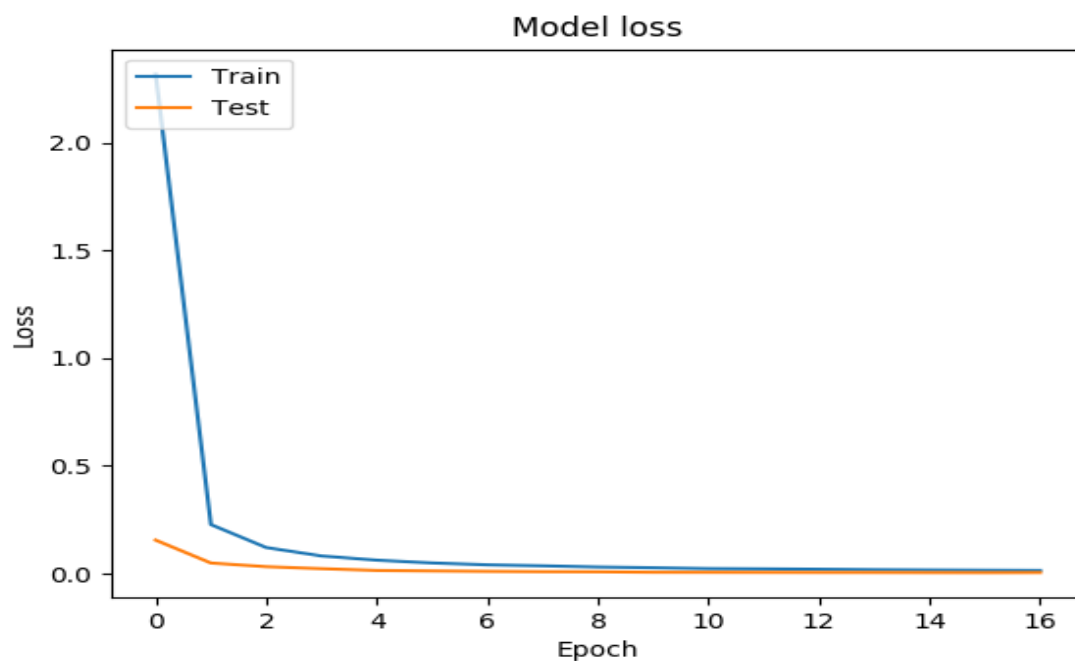


Slika 15: Prikaz preciznosti kroz broj epoha

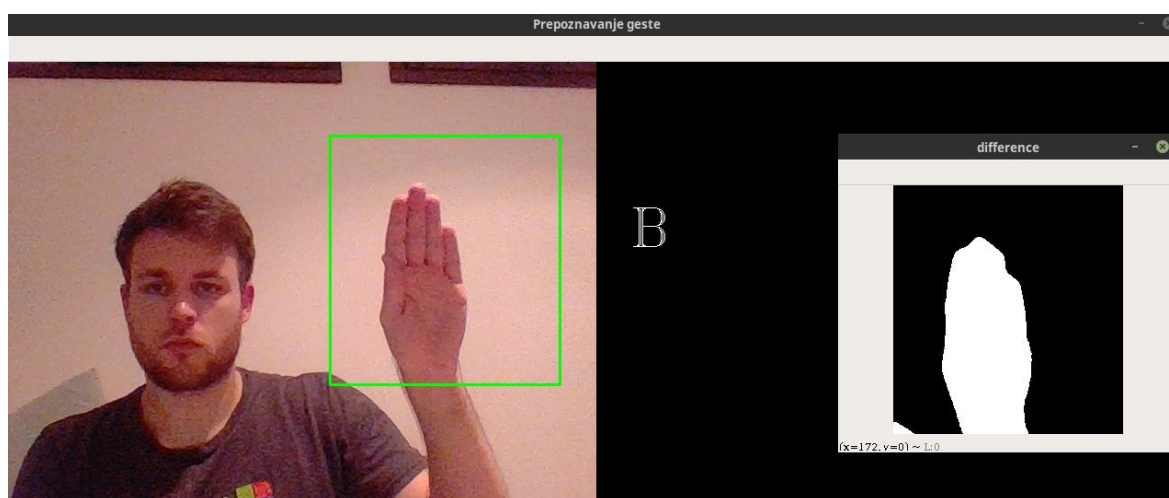
Metoda ranog zaustavljanja prekida treniranje mreže nakon 16 epoha zbog rasta gubitka te preciznost mreže tada iznosi 99.88%.

```
Epoch 15/50  
- 84s - loss: 0.0167 - acc: 0.9949 - val_loss: 0.0046 - val_acc: 0.9989  
Epoch 16/50  
- 89s - loss: 0.0156 - acc: 0.9952 - val_loss: 0.0043 - val_acc: 0.9988  
Epoch 17/50  
- 85s - loss: 0.0145 - acc: 0.9956 - val_loss: 0.0048 - val_acc: 0.9988
```

Slika 16: Ispis podataka zadnje tri epohe treniranja



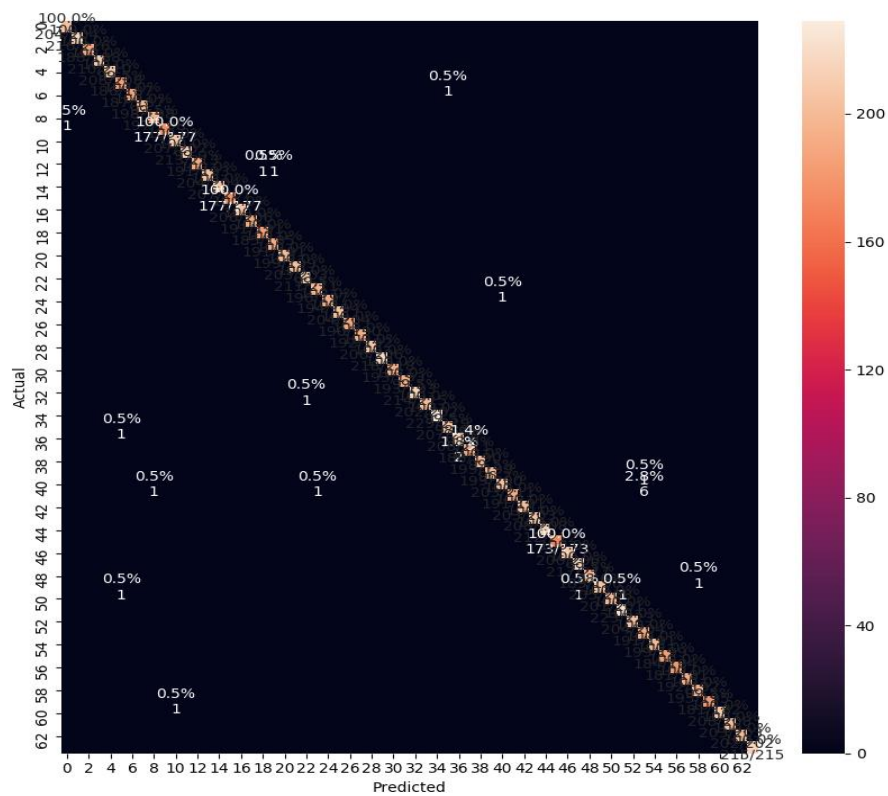
Slika 17: Prikaz gubitka kroz broj epoha



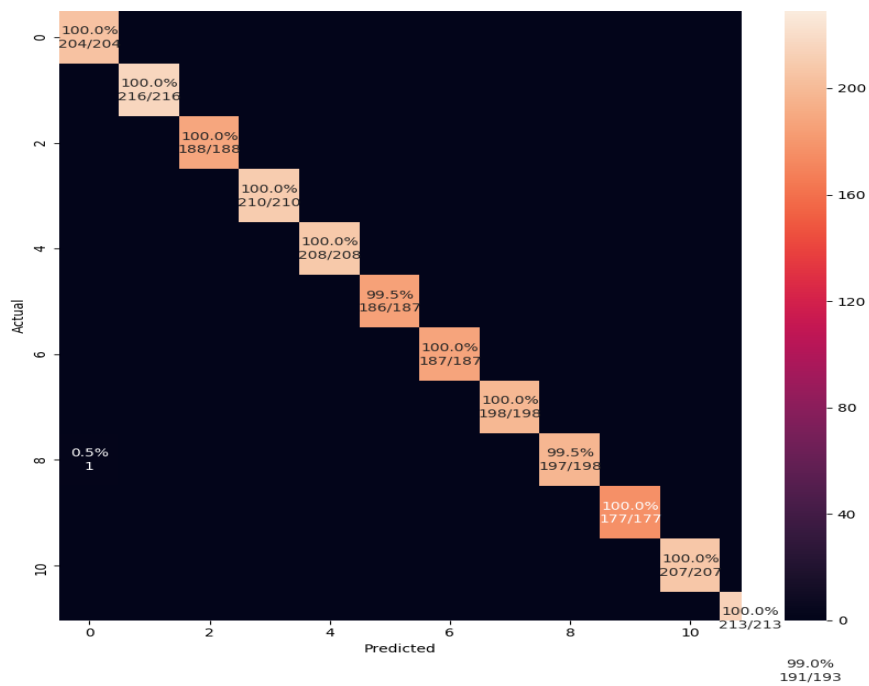
Slika 18: Primjer pravilnog rada

Dobar prikaz ispravnih i pogrešnih izlaza pruža matrica konfuzije koja može prikazati koji podaci su međusobno slični, odnosno za koju kategoriju ulaza je mreža dala krivu kategoriju izlaza.

Kako je vjerojatnost modela mreže veoma visoka, nema ni puno odstupanja.



Slika 19: Prikaz cijele matrice konfuzije



Slika 20: Prikaz dijela konfuzijske matrice

4.1. Problemi

Iako je veoma visoka preciznost mora se uzeti u obzir da je mreža trenirana u „savršenim” uvjetima gdje su slične geste vidljivo različite, kod prepoznavanja u stvarnom vremenu i stvarnim uvjetima, korisnik može nesavršeno izvesti gestu što ipak može prouzročiti krivi rezultat. Osim toga, znakovna abeceda sadrži neke veoma slične znakove i geste. Primjer toga su slova „a”, „s” i „e”, koja se mogu vidjeti na slikama.



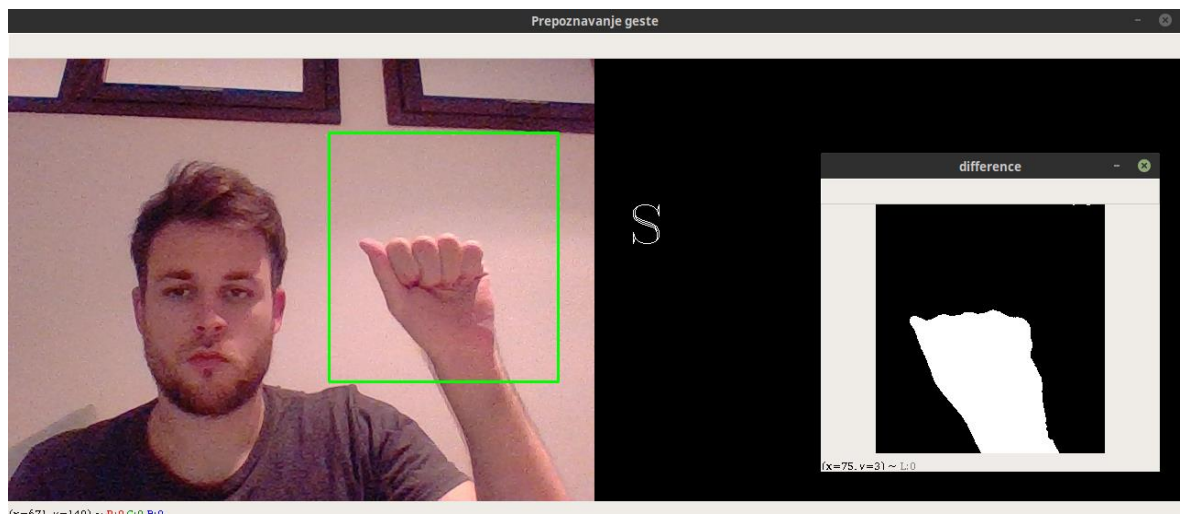
Slika 21: Prikaz slova "a", "e" i "s"

Slova su sva u obliku šake te se razlikuju samo u poziciji palca što je vidljivo na slikama, no kako mreža koristi crno-bijele slike koje nisu toliko detaljne može doći do zabune.



Slika 22: Crno-bijeli prikaz slova "a", "e" i "s"

U nastavku, na slici 23, može se vidjeti kako dolazi do zabune i mreža slovo „a” interpretira slovom „s”.



Slika 23: Primjer krive interpretacije

5. Instalacija i upute za korištenje

5.1. Potrebne instalacije (*Requirements*)

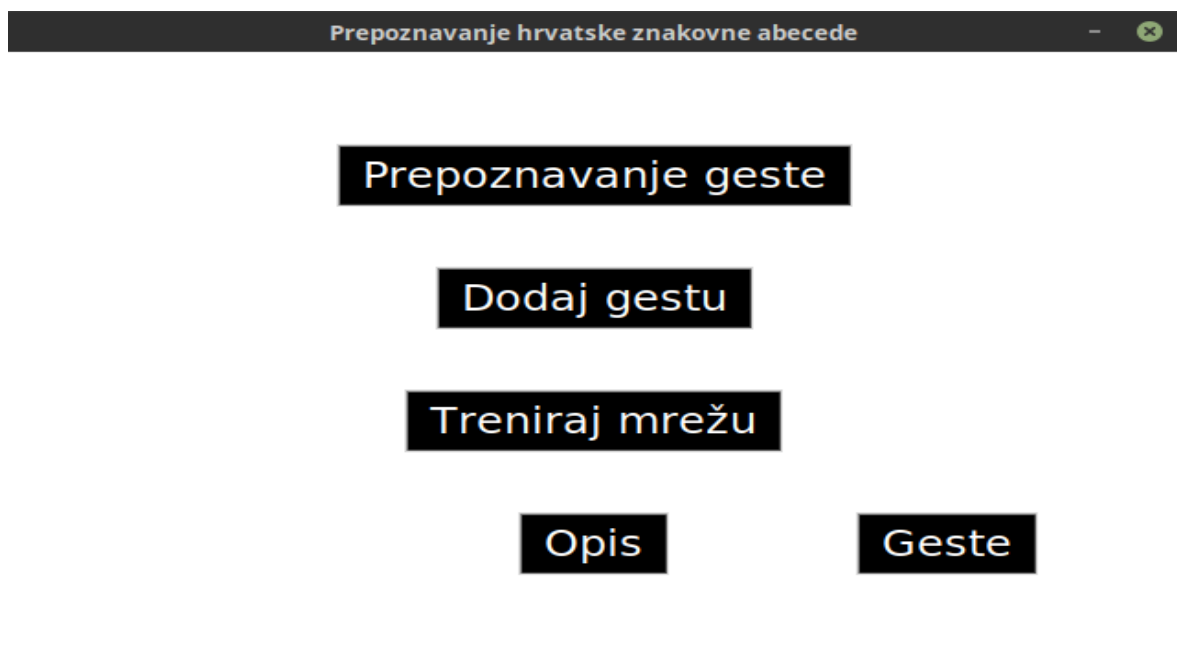
Za izvođenje programa potrebno je imati:

0. Python 3.x
1. Tensorflow 1.5
2. Keras
3. OpenCV 3.4
4. h5py
5. pyttsx3

Za instaliranje svih potrebnih alata potrebno je pozicionirati se terminalom u direktorij gdje se nalazi datoteka requirements.txt te pokrenuti naredbu (uz instaliran pip3 i Python): „pip install requirements.txt”

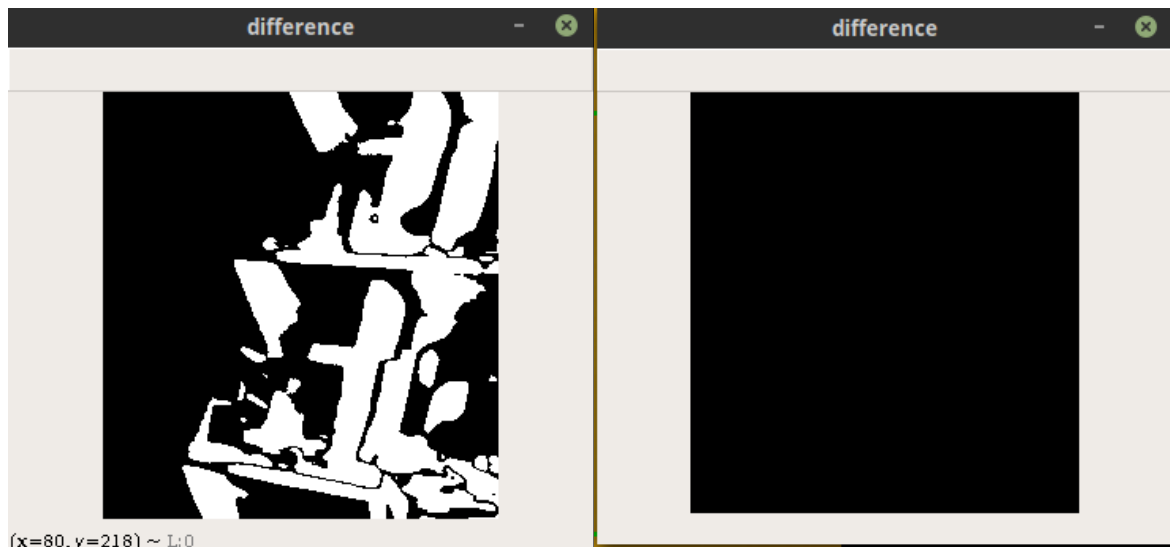
5.2. Pokretanje

Postoji jednostavno sučelje (Slika 24) koje olakšavanje služenje. Ono sadržava opcije opisa, opcija prikaza gesti(Geste), opciju dodavanja geste(Dodaj Gestu), opciju treniranja modela(treniraj mrežu) nakon dodanih gesti te opciju pokretanja glavnog programa koji prepoznaje geste(Prepoznavanje geste).



Slika 24: Prikaz početnog sučelja

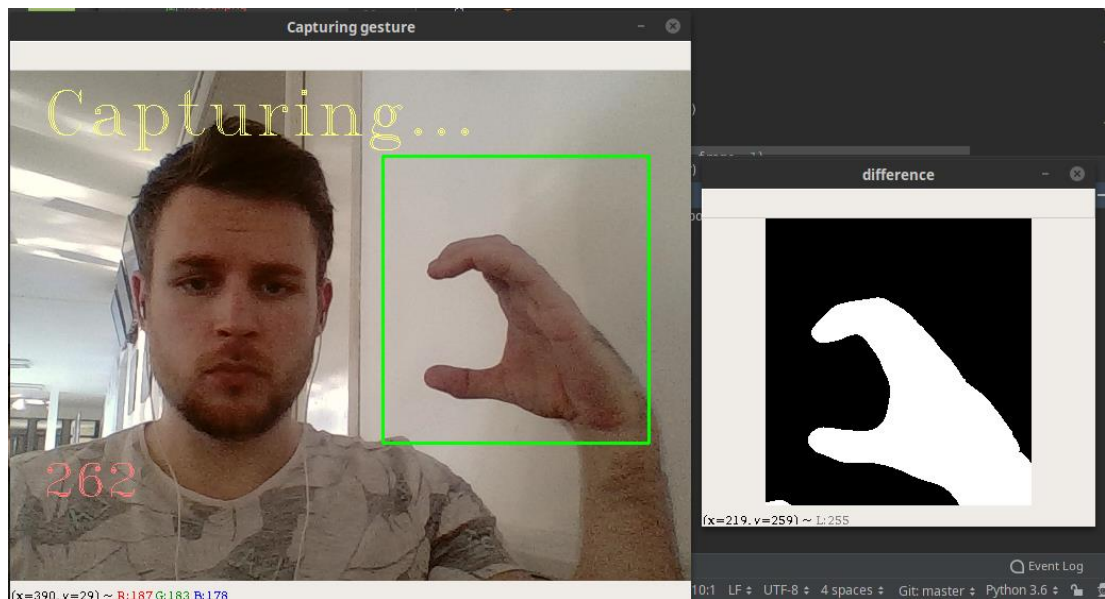
Bitno je naglasiti kako prilikom pokretanja programa bi bilo idealno kada bi površina ograničena kvadratom na kameri bila prazna, odnosno ništa se tamo ne bi trebalo nalaziti. Ako prozor „*difference*“ nije potpuno crn kod paljenja programa korisnik pritiskom na tipku „r“ može ponovo pokrenuti program. Dobar(desno) i loš(lijevo) prikaz prozora mogu se vidjeti na slici 25. Isto tako korisno je naglasiti da korisnik iz programa izlazi pritiskom na tipku „Esc“.



Slika 25: Prikazi prozora *difference*

5.2.1. Dodavanje geste

Pokretanjem programa za dodavanje geste pali se kamera te se na ekranu može vidjeti prikaz kamere uz jedan zeleni kvadrat, na njegovo mjesto treba postaviti dlan s obzirom na gestu koju želite spremiti te kliknuti tipka „c“ koja pokreće dohvaćanje (engl. *capture*) nakon kratkog vremena. Dohvaćanje se može svakog trenutka pauzirati i odpauzirati ponovnim klikom na tipku „c“. Na dodatnom prozoru koji se otvorio i koji se zove „*difference*“ može se vidjeti kakva se slika sprema, te ako to nije zadovoljavajuća gesta, na klik „r“ se može ponovo pokrenuti. Nakon otprilike minute, sustav uslika 1200 slika, koje zrcali te spremi. Nakon dodavanja geste potrebno je ponovo trenirati mrežu, kako bi se vidjeli rezultati dodavanja, što može potrajati i do 30 minuta.



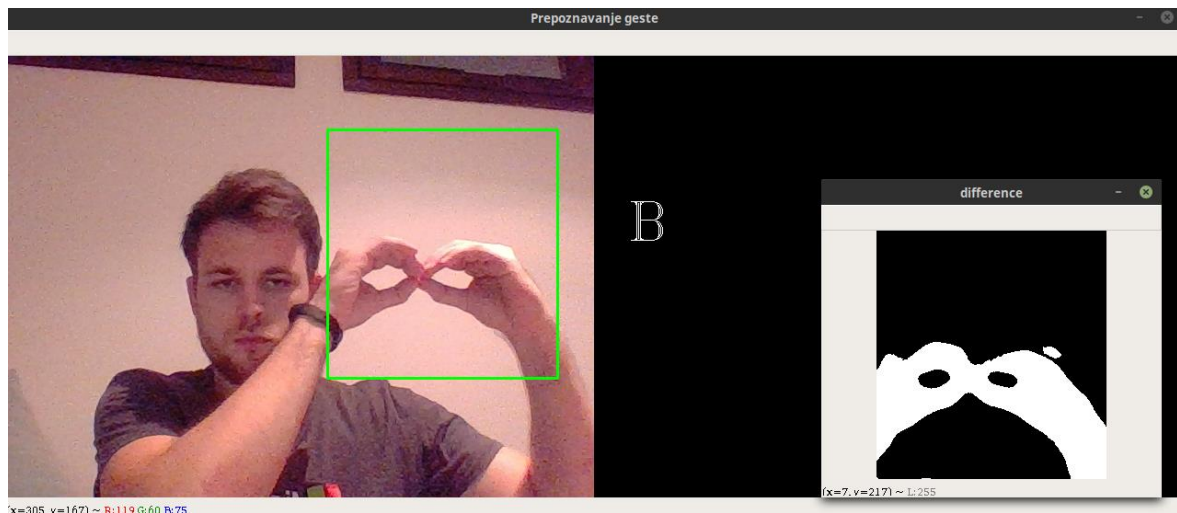
Slika 26: Primjer dodavanja geste "c"

5.2.2. Treniranje mreže

Nakon dodavanja geste potrebno je ispočetka trenirati mrežu. Pokretanjem programa za treniranje mreže nasumično se rasporede slike i podijele u skupove za treniranje, učenje i testiranje. Nakon čega slijedi „treniranje“ neuronske mreže koje može trajati i do 30 minuta te će veoma opteretiti računalo. Nakon završetka treniranja, sprema se model mreže te je sve spremno za prepoznavanje gesti.

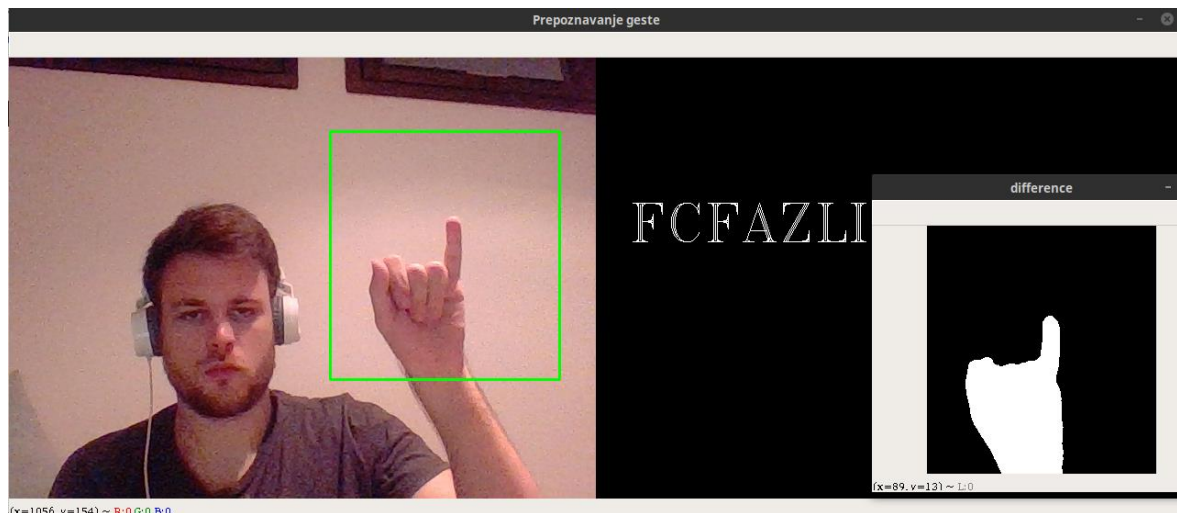
5.2.3. Prepoznavanje geste

Pokretanjem programa prepoznavanja geste pali se korisnikova kamera gdje možemo, uz uobičajeni prikaz kamere i crno-bijeli prikaz ruke „*difference*“ koji predstavlja što računalo vidi, vidjeti i nadodan dio na kameru koji predstavlja „ploču“ na kojoj se mogu iščitati geste.



Slika 27: Primjer prepoznavanja geste

Pritiskom na tipku “t” ulazimo u “textMode” koji omogućava spajanje slova, odnosno ispisivanje riječi. Za nadodavanje slova treba na ulazu biti jednaka gesta određen broj kadrova (otprilike 2 sekunde). Ponovnim pritiskom na tipku “t” vraćamo se u standardni način.



Slika 28: Primjer tvorenja riječi pomoću TextMode-a

6. Zaključak

Porastom interesa za računalni vid, raste i njegova šira upotreba i primjena. Broj ljudi koji koristi znakovni jezik je svakako značajan te bi se čitanje istog računalnim vidom moglo olakšati ljudima koji koriste taj jezik za komunikaciju ili sasvim suprotno, onima koji nisu upoznati s njime, a žele biti dio komunikacije. Prevođenje abecede korak je k tome...

Ovim radom pokazan je način rada neuronskih mreža kod problema raspoznavanja jednoručnih i dvoručnih znakova hrvatske znakovne abecede. Rezultati su pokazali dobru uspješnost klasifikacije što je uvelike ovisilo o prethodnom pretprocesiranju slike. Prikazana su i dva različita pristupa pretprocesiranja slike, odnosno izdvajanja dlana/šake. Prvi je bio histogram boje ruke koji se ovdje nije pokazao kao dobar pristup zbog loše kamere i problema osvjetljenja, ali u nekoj drugoj situaciji i drugim uvjetima bi bio veoma koristan. Drugi pristup je bio metoda oduzimanja pozadine koji je na kraju korišten za ostvarenje rada zbog svojih dobrih rezultata.

U budućnosti, ovaj rad se sigurno može nadograđivati. Prijevod abecede samo je prvi korak. Proširenjem područja promatranja sa dlana/šake na držanje tijela i glave te izraz lica moglo bi se nadograditi i na riječi što bi još više olakšalo komunikaciju.

Literatura

- [1] Kaggle URL <https://www.kaggle.com/datamunge/sign-language-mnist/>
- [2], [3], [4] Šnajder J. Čupić M., Dalbelo Bašić B. Umjetne neuronske mreže, Službeni materijali s predavanja iz predmeta Umjetna inteligencija. Fakultet elektrotehnike i računarstva, Zagreb, 2008. URL http://www.fer.hr/_download/repository/UmjetneNeuronskeMreze.pdf.
- [5] Yann LeCun, Léon Bottou, Yoshua Bengio, i Patrick Haffner. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11):2278– 2324, 1998
- D. Tran, L. Bourdev, R. Fergus, L. Torresani, M. Paluri. "Learning Spatiotemporal Features with 3D Convolutional Networks". Dartmouth College. USA. URL http://www.cvfoundation.org/openaccess/content_iccv_2015/papers/Tran_Learning_Spatiotemporal_Features_ICCV_2015_paper.pdf
- Roberto Lopez. "Artificial Neural Network". Neural Designer. URL <https://www.neuraldesigner.com/blog/perceptron-the-main-component-of-neural-networks>
- Alcoverro, M. [et al.]. Gesture control interface for immersive panoramic displays. "Multimedia tools and applications", 25 Jul 2013. URL <http://upcommons.upc.edu/bitstream/handle/2117/20565/Gesture.pdf?sequence=1&isAllowed=y>

Sustav za raspoznavanje znakovnog jezika

Sažetak

Broj ljudi koji koristi znakovni jezik je svakako značajan te bi se čitanje istog računalnim vidom moglo olakšati ljudima koji koriste taj jezik za komunikaciju ili sasvim suprotno, onima koji nisu upoznati s njime, a žele biti dio komunikacije. Prevođenje abecede korak je ka tome. U ovom radu prikazan je način rada neuronskih mreža kod problema raspoznavanja jednoručnih i dvoručnih znakova hrvatske znakovne abecede. Objašnjena su dva pristupa (histogram boje ruke, oduzimanje pozadine) problemu te postupak pripreme i nadopunjavanja skupa podataka. Na kraju su analizirana ponašanja implementiranog sustava, prikazani problemi kod prepoznavanja te opisana uputstva za korištenje programa.

Ključne riječi: strojno učenje, neuronske mreže, konvolucijske neuronske mreže, klasifikacija, znakovni jezik, abeceda, oduzimanje pozadine

System for Sign Language Recognition

Abstract

The number of people who use sign language is certainly significant and reading it using computer vision would make it easier for people who use it to communicate or, contrarily, those who are unfamiliar with it and want to be part of communication. Translating the alphabet is a step to it. This paper presents the way neural networks work in problems of recognizing letters of the Croatian sign language alphabet. Two approaches(hand histogram, background subtraction) are explained as well as the procedure for preparing and upgrading the dataset. Finally, the behaviors of the implemented system were analyzed, recognition problems were described and the instructions for use of the program were shown.

Keywords: machine learning, neural networks, convolutional neural networks, classification, sign language, alphabet, background subtraction