Zbornik radova Fakulteta tehničkih nauka, Novi Sad

UDK: 4.9
DOI: https://doi.org/10.24867/30BE22Ristic

RAZVOJ SISTEMA ZASNOVANOG NA DUBOKOM UČENJU ZA PREPOZNAVANJE LICA I GESTOVA ŠAKE

DEVELOPMENT OF A SYSTEM BASED ON DEEP LEARNING FOR FACE AND HAND GESTURE RECOGNITION

Branislav Ristić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – *U ovom radu će biti predstavljena implementacija sistema za prepoznavanje lica i gestova šake implementiranog u Python programskom jeziku. Rad uključuje teorijske osnove, kao i implementacione detalje.*

Ključne reči: *veštačka inteligencija, mašinsko učenje, neuronske mreže, duboko učenje, prepoznavcanje lica, prepoznavanje gesta šake, računarski vid*

Abstract – *This paper will present the implementation of a face and hand gesture recognition system implemented in the Python programming language. The paper includes theoretical foundations as well as implementation details.*

Keywords: *artificial intelligence, machine learning, neural networks, deep learning, face recognition, gesture recognition, computer vision*

1. UVOD

Razvoj sistema za prepoznavanje lica i gestova šake predstavlja jednu od ključnih oblasti u istraživanju računarskog vida i interaktivnih tehnologija. Ovakvi sistemi omogućavaju primenu u širokom spektru oblasti, uključujući bezbednost, automatizaciju i korisničke interfejse. Prepoznavanje lica je postalo standard u različitim aplikacijama, od automatskog otključavanja uređaja do nadzornih sistema, dok se prepoznavanje gestova šake sve više koristi u razvoju prirodnih i intuitivnih metoda interakcije.

U ovom radu predstavljen je sistem za prepoznavanje lica i gestova šake, koji koristi savremene metode računarskog vida i mašinskog učenja. Osnovna ideja rada je razvoj rešenja koje omogućava interakciju korisnika sa okruženjem pomoću prirodnih metoda komunikacije, kao što su izrazi lica i pokreti šake.

Prepoznavanje lica uključuje detekciju i identifikaciju lica, dok se prepoznavanje gestova fokusira na analizu pokreta šake. Razvoj jednog ovakvog sistema podrazumeva primenu algoritama dubokog učenja za ekstrakciju i analizu relevantnih informacija iz video-snimaka. Pored toga, ispituje se efikasnost i robusnost predloženog pristupa u različitim scenarijima. Rad takođe uključuje pregled postojećih rešenja, kao i analizu predloženog modela sa aktuelnim metodama koje se koriste u ovoj oblasti.

NAPOMENA:
Ovaj rad proistekao je iz master rada čiji mentor je bio dr Dušan Gajić, vanr. prof.

2. OSNOVE MAŠINSKOG UČENjA I NEURONSKIH MREŽA

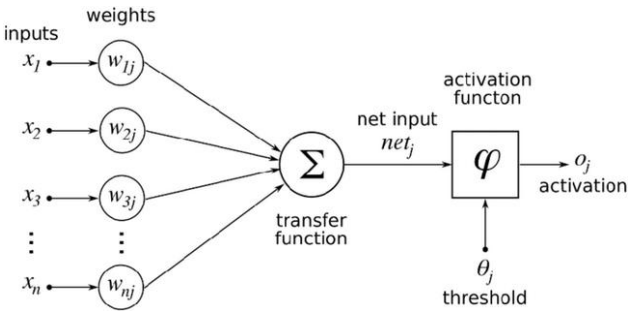
Veštačka inteligencija, u širem smislu, predstavlja mogućnost posedovanja inteligencije od strane mašina, konkretno, računarskih sistema [8].

Algoritam (ili model) mašinskog učenja jeste algoritam koji poseduje mogućnost da uči iz podataka koji mu se pruže. Mašinsko učenje predstavlja podksup veštačke inteligencije. U okviru mašinskog učenja veliku oblast predstavlja duboko učenje [2, p. 99].

Algoritmi mašinskog učenja se ugrubo mogu podeliti u nadgledane i nenadgledane. Nadgledani algoritmi mašinskog učenja poseduju obeležja koja predstavljaju ulaz u algoritam (trening skup), ali uz njih i ciljno obeležje. Nenadgledani algoritmi imaju za cilj da među podacima sami pronađu šablone [2, pp. 104-105].

2.1. Neuronske mreže

Veštački neuron jeste matematička funkcija koja je nastala po ugledu na neuron u ljudskom mozgu. Predstavlja elementarnu jedinicu veštačkih neuronskih mreža, koje su sastavljene od slojeva neurona. Veštački neuron na ulazu ima listu ulaza (x_i), odnosno dendrite, svakom od ulaza ima pridruženu težinu (w_{ij}). Sumiranjem proizvoda ulaza i težina, potom nadodavanjem praga (θ_j) i prosleđivanjem tog rezultata kao ulaz aktivacionu funkciju (φ) dobija se aktivacija, odnosno akson, O_j, j-tog neurona u sloju neuronske mreže. Ilustracija se nalazi na Slici 1 [9].



Slika 1. *Prikaz veštačkog neurona (preuzeto sa [14])*

Unapred propagirajuće (engl. *feed forward*) neuronske mreže su klasa unutar neuronskih mreža koju karakteriše nepostojanje povratnih sprega u mreži. Duboke neuronske

mreže obuhvataju mreže koje imaju velik broj slojeva neurona [1, p. 75].

2.2. Konvolucione neuronske mreže

Konvolucione neuronske mreže predstavljaju posebnu vrstu neuronskih mreža koje obrađuju podatke sa rešetkastom topologijom. Međutim, konvolucija se zapravo ne koristi u konvolucionim mrežama, već se najčešće koristi kros-korelacija.

$$S(i,j) = (I * K)(i,j) = \sum_{m} \sum_{n} I(i+m, j+n) K(m,n) \quad (1)$$

Kros-korelacija prikazana u (1) suštinski predstavlja konvoluciju fotografije *I*, bez rotacije kernel matrice *K*. Rezultat operacije kros-korelacije naziva se mapa svojstava (engl. feature map) [2, pp. 330-333].

2.3. Rezidualne neuronske mreže

Problem obuke veoma dubokih neuronskih mreža je rešiv upotrebom prečica između slojev. Odnosno, ulaz u blok i izlaz iz bloka se kombinuju kako bi se dobio jedan rezultat. Na ovaj način, pošto se inicijalno težine i pragovi inicijalizuju na nasumične vrednosti, se omogućava u početku manji faktor šuma pri propagaciji unapred i unazad. Ovo dalje dozvoljava kreiranje i obuku rezidualnih neuronskih mreža sa do nekoliko stotina ili hiljada slojeva. Koriste se između ostalog i za generisanje vektorske reprezentacije objekata [3].

3. OPIS TEHNOLOGIJA

U ovom poglavlju će biti predstavljene korišćene tehnologije, sa fokusom na alate i metode koji su primenjeni u istraživanju i razvoju.

3.1. *Linux*

Linux je familija operativnih sistema otvorenog koda koja je bazirana na Linux kernelu (kernel u ovom kontekstu označava jezgro operativnog sistema, ne kernel matricu). Otvorenost koda omogućava visoku podesivost sistema prema različitim potrebama, stoga postoji veliki broj distribucija Linuxa [11].

3.2. *Python*

Python je programski jezik visokog nivoa apstrakcije koji pripada grupi interpretiranih jezika. Poznat je po čitljivosti i svestranosti. Zbog svojih osobina popularan je odabir od strane početnika, a i iskusnih programera. Podržava više programskih paradigmi uključujući i proceduralnu, objektnu i funkcionalnu [13].

3.3. *MediaPipe*

MediaPipe predstavlja skup biblioteka i alata koje omogućuju programerima brzu upotrebu veštačke inteligencije. Produkt je kompanije Google. Programerima je omogućeno stvaranje modela po meri, ali i korišćenje već istreniranih modela, koje Gugl zvanično čini dostupnim.

Detekcija ključnih tačaka šake (engl. Hand landmarks detection) dozvoljava detekciju ključnih tačaka svih šaka koje se nalaze u kadru.

Model prepoznavanja gesta šake (engl. Gesture recognition) je paket koji koristi rezultate prethodno opisanog modela za ključne tačke šake kao ulaze u model za klasifikaciju. Kategorije gestova šake: nepoznat gest, oznaka: *Unknown*; zatvorena šaka, oznaka: *Closed_Fist*; otvoreni dlan, oznaka: *Open_Palm*; pokazivanje nagore, oznaka: *Pointing_Up*; palac dole, oznaka: *Thumb_Down*; palac gore, oznaka: *Thumb_Up*; pobednička gestikulacija, oznaka: *Victory*; ljubav, oznaka: *ILoveYou* [4, 5, 6].

3.4. *DeepFace*

Sefik Serengil je softverski inženjer i kreator biblioteke *DeepFace* koja omogućava podršku mašinskog učenja vezano za lica. *DeepFace* predstavlja omotač oko raznih modela poput *VGG-Face, Google Facenet, Facebook DeepFace, DeepID, OpenCV, Dlib* [7].

3.5. *OpenCV*

OpenCV je softverska biblioteka otvorenog koda koja pruža podršku kompjuterskog vida u realnom vremenu. Nastala je u okviru kompanije Intel 2000. godine, a 2011. godine počinje da pruža podršku GPU akceleracije. Napisana je u jeziku C++, dok su dostupna vezivanja za Python, MATLAB, Java programske jezike [12]. Biblioteka poseduje model koji omogućava analizu sentimenata lica. Rezultat analize sentimenata predstavlja klasu koja može imati sledeće vrednosti: ljuta, oznaka: angry; gađenje, oznaka: disgust; strah, oznaka: fear; sreća, oznaka: happy; tuga, oznaka: sad; iznenađenje, oznaka: surprise; neutralna, oznaka: neutral. [7]

3.6. *Dlib*

Dlib predstavlja biblioteku opšte namene napisane u jeziku C++. Od početka razvoja godine 2002. implementirani su razni alati, te u 2016. skup alata je posedovao podršku za: niti, strukture podataka, linearnu algebru, mašinsko učenje, obradu slike, s tim da se poslednjih godina fokus usmerava na mašinsko učenje [10].

4. OPIS IMPLEMENTACIJE

U ovom poglavlju razmatraće se implementacija sistema koja objedinjuje prepoznavanje lica i gesta šake. Osim toga, analiziraće se tehničke prepreke i rešenja za optimizaciju performansi i tačnosti prepoznavanja.

Cilj sistema je da podrži prepoznavanje gestova šake, povezivanje šake sa odgovarajućim licem, prepoznavanje emocije na licu i da li je lice već poznato od ranije (da li je korisnik već interagovao sa sistemom), i najzad, beleženje rezultata gesta. Interakcija sa sistemom se svodi na pokazivanje gesta šake, gledajući u kameru.

Sistem koristi postojeće algoritme mašinskog učenja, među kojima su:

- prepoznavanje gesta šake - MediaPipe v0.10.14,
- prepoznavanje emocije - DeepFace v0.0.92 (OpenCV v4.10.0.84),
- prepoznavanje izgleda lica - DeepFace v0.0.92 (Dlib v19.24.4).

Za prepoznavanje gesta šake korišćen je HandGestureClassifier, odnosno gesture_recognizer.task iz MediaPipe softverskog paketa [4].

Za prepoznavanje emocija korišćen je facial_expression_model_weights.h5 iz *OpenCV*, dok je za prepoznavanje izgleda lica korišćen dlib_face_recognition_resnet_model_v1.dat iz Dlib paketa.

Sistem podržava vizuelizaciju u pogledu obojenih granica (kutija) oko detektovanih šaka i lica, kao i tekstualno beleženje rada sistema na standardnom izlazu.

4.1. Podaci

Podaci koji predstavljaju rezultate rada sistema su:

- FaceHash,
- emocija,
- tip gesta,
- vreme (Unix timestamp)

FaceHash predstavlja base64 kodovanu vrednost SHA256 funkcije nad vektorskom reprezentacijom lica. Emocija predstavlja klasu čije su vrednosti nabrojane u poglavlju 3.5. Tip gesta podrazumeva vrednosti koje su nabrojane u poglavlju 3.3.2. Vreme prestavlja standardni Unix timestamp izražen u sekundama.

4.2. Memorija

U ovom poglavlju će biti obrađena tri nivoa memorije u kojima se skladište podaci u sistemu.

4.2.1 Pret-keš

Pret-keš se koristi kako bi se poboljšala tačnost algoritma za prepoznavanje lica. Uprkos velikoj tačnosti algoritma, može se dogoditi da pri računanju vektorske reprezentacije lica dobije netačno podudaranje. Iz tog razloga se rezultati obrade kadrova koji su vremenski blizu, uzimaju kao jedna celina. Algoritam čeka, i ukoliko se ne pojavi ni jedno lice u okviru od 30 kadrova (podesiv parametar), pokreće se obrada. Nad blokom koji se obrađuje se većinski odlučuje koje je vrste lice (novo lice ili neko staro). U pret-kešu se pored gorenavedenih podataka čuva i vektorska reprezentacija lica. Ovo služi kao kratkotrajna memorija, rezultat obrade pret-keša se smešta u keš kao jedan podatak.

4.2.2 Keš

Vrednosti sa kojima se upoređuju potencijalna nova lica u kadru se nalaze u kešu. Keš je mesto u kojem se skladište rezultati dobijeni obradom bloka rezultata u pret-kešu. Ukoliko se desi da je rezultat zabeležen za neko staro lice, vrednosti gesta vremena i emocije se ažuriraju. U kešu se, takođe, pored gorenavedenih podataka čuva i vektorska reprezentacija lica. Ovo služi kao srednjetrajna memorija.

4.2.2 Masovna memorija

Ukoliko za dato lice u kešu prođe vremenski period od 30 minuta, lice se memoriše na disk (baza podataka) i potom uklanja iz keša.

Takođe, svakih 15 minuta se pokreće kopiranje podataka iz keša na disk (snapshotting, bez brisanja iz keša).

Oba načina čuvanja podataka na disku se ogledaju korišćenjem .csv datoteke. Kao baza podataka se koristi jedna datoteka, dok se za stanja (snapshot) sistema koristi iznova nova datoteka, koja u svom nazivu sadrži i vreme kreiranja iste.

4.3. Algoritam

Algoritam je u stanju da detektuje maksimalno 5 šaka u kadru od jednom. Kao mera udaljenosti lica korišćeno je euklidsko rastojanje.

Aplikacija prima signal sa veb-kamere u vidu nizova slika. Ukoliko algoritam u kadru uspešno prepozna gest šake koji nije "*None*", pokreće se detekcija lica i analiza sentimenata. Ukoliko je ta operacija uspešna, vrši se računanje vektorske reprezentacije lica. Ukoliko je ta operacija uspešna, povezuje se šaka sa licem i skladišti kao međurezultat. Zatim se proverava da li je lice već interagovalo sa sistemom (već je u kešu). Ukoliko jeste, dodeljuje mu se odgovarajuća SHA256 vrednost, ukoliko nije, neophodno je da se izračuna. Taj rezultat se skladišti u pret-kešu.

Svakih 30 kadrova se obrađuje pret-keš. Ovo je način da se poboljša tačnost algoritama mašinskog učenja u slučaju da dođe do netačnog poklapanja ili nepoklapanja.

Svaki treći kadar se zapravo obrađuje, dok se za ostale kadrove korisniku prikazuju prethodno izračunati rezultati (granice lica i šaka). Ovo je način da se unapredi odziv sistema budući da metoda koja je odgovorna za prepoznavanje gestova lica vrši obradu sinhrono.

Paralelno sa niti glavnog algoritma se izvršava i nit pražnjenja keša na disk i beleženja stanja keša (engl. *snapshotting*). Ona započinje svoj rad kada i glavni algoritam.

4.4. Detaljnija razmatranja

Obrada svakog trećeg kadra je posledica nestabilne asinhrone obrade u okviru MediaPipe biblioteke. Ovaj parametar je svakako podesiv prema računarskoj moći mašine na kojoj se aplikacija izvršava. Algoritam detektuje maksimalno 5 šaka u kadru. Ovo je omogućeno kako bi se eventualne nepravilnosti rada algoritma prepoznavanja šaka ublažile.

Računanje aritmetičke sredine nad pret-kešom se ispostavlja da značajno poboljšava tačnost modela mašinskog učenja za prepoznavanje lica, iako je reklamirana tačnost 97,53%. Za poređenje sličnosti dva lica se koriste isključivo vektorske reprezentacije lica. U masovnoj memoriji se ne skladišti vektorska reprezentacija lica, već samo SHA256 vrednost vektora. Mogućnost poravnavanja lica pri detekciji se svodi na rotaciju za korake od po $\pi/2$, stoga je neophodno što ispravnije (potpuno vertikalno) postaviti kameru. Performanse aplikacije variraju u odnosu na osvetljenje.

Za najbolje performanse obezbediti dobro, ali ne previše, osvetljenu prostoriju. Preterano pomeranje i pričanje za vreme interacije sa sistemom takođe utiče na performanse rada algoritma. Pozadina bi, idealno, trebalo da bude jednobojno platno.

5. ZAKLJUČAK

Predloženo rešenje u ovom radu omogućava interakciju sa sistemom isključivo putem prirodne, neverbalne komunikacije, koristeći gestove i vizuelne signale. Ovaj pristup je omogućen primenom relativno jednostavnih i lako dostupnih tehnologija koje ne zahtevaju velike hardverske resurse, čineći sistem pogodnim za širok spektar aplikacija. Osnovna ideja je da se korisnička interakcija obavlja bez verbalne komunikacije, čime se omogućava intuitivniji i prirodniji način rada sa sistemom.

Sistem nudi visoku fleksibilnost kroz podešavanje parametara kao što su broj kadrova koji se preskaču, maksimalan broj detektovanih šaka, kao i izbor različitih modela mašinskog učenja. Ova fleksibilnost omogućava da se sistem lako prilagodi različitim okruženjima i hardverskim konfiguracijama, što je korisno u raznovrsnim scenarijima primene. Pored toga, upotreba pret-keša i keša povećava efikasnost obrade podataka, omogućavajući postepenu obradu i unapređenu tačnost sistema.

Sistem takođe delimično garantuje privatnost korisnika, jer trajno ne skladišti osetljive podatke kao što su vektorske reprezentacije lica, osim u RAM-u tokom procesa uparivanja lica.

Dodatna poboljšanja mogu uključivati razvoj naprednijih algoritama za prepoznavanje lica koji bi mogli bolje da se nose sa različitim uglovima i uslovima osvetljenja. Takođe, uvođenje podrške za složenije gestove i korišćenje više ruku istovremeno moglo bi da proširi primenu sistema u složenijim scenarijima. Razmatranje ovih poboljšanja moglo bi značajno povećati primenu i efikasnost ovog sistema u različitim oblastima.

6. LITERATURA

[1] Andriy Burkov. (2019). THE HUNDRED-PAGE MACHINE LEARNING BOOK. Andriy Burkov.

[2] Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. The MIT Press

[3] He, K., Zhang, X., Ren, S., & Sun, J. (2015, December 10). Deep Residual Learning for Image Recognition. ArXiv.org; arXiv. https://arxiv.org/abs/1512.03385

[4] Google. (n.d.-a). Gesture recognition task guide | Google AI Edge. Google for Developers. Retrieved September 2, 2024, from https://ai.google.dev/edge/mediapipe/solutions/vision/gesture_recognizer

[5] Google. (n.d.-b). Hand landmarks detection guide | Edge. Google for Developers. Retrieved September 2, 2024, from https://ai.google.dev/edge/mediapipe/solutions/vision/hand_landmarker

[6] Google. (n.d.-c). SOLUTION DETAILS. https://storage.googleapis.com/mediapipe-assets/Model%20Card%20Hand%20Tracking%20(Lite_Full)%20with%20Fairness%20Oct%202021.pdf

[7] Serengil, S. I. (2020, August 30). serengil/deepface. GitHub. https://github.com/serengil/deepface

[8] Wikipedia. “Artificial Intelligence.” Wikipedia, Wikimedia Foundation, 18 Feb. 2019, en.wikipedia.org/wiki/Artificial_intelligence. Accessed 2 Sept. 2024.

[9] Wikipedia Contributors. “Artificial Neuron.” Wikipedia, Wikimedia Foundation, 4 Oct. 2018, en.wikipedia.org/wiki/Artificial_neuron. Accessed 2 Sept. 2024.

[10] Wikipedia Contributors. “Dlib.” Wikipedia, Wikimedia Foundation, 22 Sept. 2019, en.wikipedia.org/wiki/Dlib. Accessed 2 Sept. 2024.

[11] Wikipedia Contributors. “Linux.” Wikipedia, Wikimedia Foundation, 14 Sept. 2019, en.wikipedia.org/wiki/Linux. Accessed 2 Sept. 2024.

[12] Wikipedia Contributors. “OpenCV.” Wikipedia, Wikimedia Foundation, 29 Aug. 2019, en.wikipedia.org/wiki/OpenCV. Accessed 2 Sept. 2024.

[13] Wikipedia Contributors. “Python (Programming Language).” Wikipedia, Wikimedia Foundation, 4 May 2019, en.wikipedia.org/wiki/Python_(programming_language). Accessed 2 Sept. 2024.

[14] Polat, Ali. (2017). Hidden bank charges amidst ethics, transparency, and regulation: Case of rebate programs of international banks.

Kratka biografija:

Branislav Ristić je rođen 30. maja 2000. godine u Novom Sadu. Osnovne akademske studije završio je školske 2022/2023. godine, smer Informacioni inženjering. Planira da nastavi sa svojim usavršavanjem kroz doktorske akademske studije i rad na fakultetu.

kontakt: branislav.ristic@uns.ac.rs