RAZVOJ I EVALUACIJA MODELA ZASNOVANOG NA VEŠTAČKOJ INTELIGENCIJI ZA PREPOZNAVANJE SLOVA SRPSKOG ZNAKOVNOG JEZIKA

By Mladen Radaković

WORD COUNT 31994
TIME SUBMITTED 03-APR-2024 03:07PM
PAPER ID 108037183



UNIVERZITET SINGIDUNUM

BEOGRAD

DEPARTMAN ZA POSLEDIPLOMSKE STUDIJE

DOKTORSKA DISERTACIJA

RAZVOJ I EVALUACIJA MODELA ZASNOVANOG NA VEŠTAČKOJ INTELIGENCIJI ZA PREPOZNAVANJE SLOVA SRPSKOG ZNAKOVNOG JEZIKA

MENTOR:

Prof. dr Marina Marjanović

STUDENT:

Mladen Radaković

Broj indeksa:

466013/2019

Beograd, 2024.



Singidunum University

Department for postgraduate studies

PhD Thesis

DEVELOPMENT AND EVALUATION OF MACHINE LEARNING MODEL FOR RECOGNIZING SERBIAN SIGN LANGUAGE ALPHABETS

MENTOR:

Prof. dr Marina Marjanović

STUDENT:

Mladen Radaković

Index number:

466013/2019

Beograd, 2024.

UNIVERZITET SINGIDUNUM

DEPARTMAN ZA POSLEDIPLOMSKE STUDIJE

Danijelova 32, Beograd

Kandidat: Mladen Radaković

Broj indeksa: 466013/2019

Studijski program: Inženjerski sistemi u menadžmentu

Tema:

RAZVOJ I EVALUACIJA MODELA ZASNOVANOG NA VEŠTAČKOJ INTELIGENCIJI ZA PREPOZNAVANJE SLOVA SRPSKOG ZNAKOVNOG JEZIKA

Mentor:

Prof. dr Marina Marjanović, redovni profesor

Univerzitet Singidunum, Beograd

Članovi komisije:

Prof. dr Marko Tanasković, vanredni profesor

Univerzitet Singidunum, Beograd

Prof. dr Boško Nikolić, redovni profesor

Elektrotehnički fakultet, Univerzitet u Beogradu

Beograd, 2024. godine

RAZVOJ I EVALUACIJA MODELA ZASNOVANOG NA VEŠTAČKOJ INTELIGENCIJI ZA PREPOZNAVANJE SLOVA SRPSKOG ZNAKOVNOG JEZIKA

Apstrakt: Primenom veštačke inteligencije ostvaruju se raznolike i brojne koristi koje su već implementirane u raznim sektorima. Međutim, efikasnost tih rešenja, kao i vreme potrebno za njihovu izradu, variraju. Trendovi upućuju na metode koje omogućavaju brža i preciznija rešenja. Nažalost, slušno oštećena zajednica sa srpskog govornog područja ne uživa značajnu podršku tehnološke naučne zajednice. Postojeći radovi i aktivnosti manje su usmereni ka razvoju interaktivnih alata koji bi olakšali svakodnevni život ove zajednice.

Glavni problem sa kojim se slušno oštećeni suočavaju je nedostatak efikasne komunikacije sa čujućom zajednicom, što otežava obavljanje svakodnevnih obaveza, poslovnih poduhvata i svih drugih aktivnosti za koje je komunikacija neophodna. Do sada dostupna tehnološka rešenja, kao što su internet portali i programi, nisu pružila korisnicima mogućnost interakcije korišćenjem srpskog znakovnog jezika.

Ovaj rad ima za cilj da olakša komunikaciju i razmenu informacija u ovoj zajednici kako bi se unapredio socijalni i poslovni aspekt njihovog života. Znakovni jezici su specifični, a nedostatak univerzalno priznatog jezika zahteva od regionalnih i lokalnih zajednica slušno oštećenih osoba da izazove sa kojima se susreću kako u privatnom tako i u poslovnom okruženju prevazilaze samostalno. Glavni fokus ovog istraživanja jeste unapređenje interaktivne edukacije, poboljšanje opšte komunikacije i proširenje angažovanosti ove zajednice u poslovnim aktivnostima. Korišćenjem metoda mašinskog učenja, kreirana je obimna baza podataka zasnovana na pokretima srpskog znakovnog jezika. Preko 40 učesnika je demonstriralo gestove ispred kamere, snimajući preko 10.000 video snimaka za 30 slova azbuke srpskog jezika. Specijalno kreirani Python kôd je prepoznavao ključne tačke na rukama i telu u numeričke koordinate i beležio dobijene rezultate u bazu kreiranu namenski za tu svrhu.

Na osnovu kreirane baze podataka, korišćenjem metoda mašinskog učenja, projektovano je i evaluirano više različitih modela, od kojih je za primenu odabran najprecizniji respektujući stopu tačnosti predviđanja rezultata. Funkcionisanje i

karakteristike ovog modela su predstavljene u finalnom eksperimentalnom istraživanju.

Za potrebe istraživanja, kreiran je funkcionalni računarski program koji implicira mogućnosti, tačnost i pouzdanost praktične primene modela mašinskog učenja zasnovanog na prethodno generisanoj bazi podataka. Demonstriranjem gestova slova srpskog znakovnog jezika ispred kamere, računar i model mašinskog učenja prepoznaju slova srpske azbuke i prikazuju prognozirane rezultate. Prepoznati gestovi se konvertuju u numeričke koordinate formirajući tako bazu podataka. Postavljanjem ove sveobuhvatne i kompleksne baze podataka slova srpskog znakovnog jezika na internet, omogućen je dalji razvoj modela mašinskog učenja za srpsko govorno područje. Ovaj pristup predstavlja osnovu za razvoj i implementaciju sličnih platformi drugih znakovnih jezika sa ciljem da se obezbedi podrška i drugim slušno oštećenim zajednicama.

Ključne reči: srpski znakovni jezik, mašinsko učenje, veštačka inteligencija

Naučna oblast: Tehničko-tehnološke nauke

Uža naučna oblast: Elektrotehnika i računarstvo

DEVELOPMENT AND EVALUATION OF MACHINE LEARNING MODEL FOR RECOGNIZING SERBIAN SIGN LANGUAGE ALPHABETS

Abstract: By applying artificial intelligence, diverse and numerous benefits are realized, already implemented across various sectors. However, the efficiency of these solutions, as well as the time required for their development, varies. Trends point towards methods that enable faster and more precise solutions. Unfortunately, the hearing-impaired community from the Serbian-speaking region lacks significant support from the technological scientific community. Existing efforts and activities are less directed towards the development of interactive tools that would facilitate the daily life of this community.

The main problem that the hearing-impaired face is the lack of effective communication with the hearing community, which makes it difficult to carry out daily tasks, business endeavors, and all other activities for which communication is essential. So far, the available technological solutions, such as internet portals and programs, have not provided users with the ability to interact using Serbian Sign Language.

This paper aims to facilitate communication and information exchange within this community to enhance the social and business aspects of their lives. Sign languages are specific, and the lack of a universally recognized language requires regional and local communities of the hearing impaired to independently overcome the challenges they face both in private and business environments. The primary focus of this research is to enhance interactive education, improve general communication, and expand the engagement of this community in business activities. Utilizing machine learning methods, an extensive database based on the movements of Serbian sign language has been created. Over 40 participants demonstrated gestures in front of the camera, recording over 10,000 videos for the 30 letters of the Serbian alphabet. Specially designed Python code recognized key points on the hands and body into numerical coordinates and recorded the results in a database created specifically for this purpose.

Based on this database, multiple different models were projected and evaluated using machine learning methods, selecting the most accurate and reliable

among them. The functioning and characteristics of this model are presented in the final experiment.

A functional computer program has been created to demonstrate the capabilities, accuracy, and reliability of practical application of the machine learning model based on the previously generated database. By displaying Serbian sign language gestures in front of the camera, the computer and machine learning model recognize Serbian alphabet letters and display predicted results. By placing this unique and extensive database of Serbian sign language letters on the internet, further development of machine learning models in the Serbian-speaking region is enabled.

Keywords: Serbian sign language, machine learning, artificial intelligence

Scientific field: Technical and technological sciences

Narrow scientific field: Electrical engineering and computing Science

Zahvalnica

Zahvaljujem se na podršci i saradnji svom mentoru, dr Marini Marjanović na ukazanom poverenju, savetima i uloženom trudu.

Veliku zahvalnost dugujem svojoj dragoj prijateljici dr Svetlani Dabić Ceci, na drugarstvu, motivaciji, stručnim sugestijama i nesebičnoj velikoj pomoći. Njeni saveti vezani za pisanje stručnih naučnih radova i ove disertacije su za mene bili dragoceni.

Ogromno hvala dr Ivani Ristić i njenim studentima na uloženom trudu i vremenu za snimanje video klipova sa gestovima slova srpskog znakovnog jezika. Bez njenog profesionalnog angažmana i pokretačke energije, kreirana generisana baza podataka ne bi bila u ovoj meri obimna i kvalitetna.

Hvala koleginici i drugarici dr Mileni Ilić na podršci i stručnoj pomoći pruženoj na putu do disertacije. Zajednički rad na nekim naučnim radovima, učestvovanje na konferencijama i pisanje udžbenika su me motivisali i pomogli na putu do završetka doktorata.

Hvala porodici i prijateljima koji su me razumeli i podržali moj trud.

Zauvek u mislima i srcu, moj stric Miroslav ostaje večna i snažna podrška, izvor pokretačke energije i oštri kritičar, pa je tako i deo njega ugrađen u ovu disertaciju.

Mojim najdražim roditeljima, Ljiljani i Urošu, ogromno hvala za svu energiju, strpljenje i pruženu podršku kroz celo školovanje i život. Hvala im na ljubavi, požrtvovanju, obrazovanju i vaspitanju koje su mi pružili.

Najveću zahvalnost dugujem svojoj supruzi Katarini na ideji i inspiraciji koju mi je dala za ovu disertaciju, na njenoj ljubavi, strpljenju, razumevanju i svakodnevnoj podršci.

Moje sjajne ćerke, Milena, Jovana i Dunja su mi najveći oslonac i nepresušan izvor sreće i radosti. Njihovo postojanje i ljubav me motiviše da idem napred i nastavim da napredujem, učim i radim.

Sadržaj

ZAHVALNICA 8

1. UVOD 12

1.1. PREDMET ISTRAŽIVANJA, HIPOTEZE I CILJ 12

1.1.1. PROBLEM ISTRAŽIVANJA 12

1.1.2. CILJEVI I ZADACI ISTRAŽIVANJA 13

1.1.3. HIPOTEZE U ISTRAŽIVANJU 14

1.1.4. METODE ISTRAŽIVANJA 14

1.1.5. NAUČNA I DRUŠTVENA OPRAVDANOST ISTRAŽIVANJA 15

1.2. SADRŽAJ DISERTACIJE 16

2. SRPSKI ZNAKOVNI JEZIK I MAŠINSKO UČENJE 18

2.1. ISTORIJAT NASTANKA I KARAKTERISTIKE SRPSKOG ZNAKOVNOG JEZIKA 18

2.2. PREGLED ODABRANIH ISTRAŽIVANJA IZ OBLASTI PREPOZNAVANJA ZNAKOVNIH JEZIKA 20

2.3. MOTIVACIJA ZA RAZVOJ MODELA MAŠINSKOG UČENJA U PREPOZNAVANJU SRPSKOG ZNAKOVNOG JEZIKA

30

3. KREIRANJE BAZE PODATAKA SRPSKOG ZNAKOVNOG JEZIKA 31

3.1. METODOLOGIJA KREIRANJA BAZE PODATAKA 32

3.1.1. IZBOR STRUKTURE I TIPA PODATAKA 34

3.1.2. ODABIR UREĐAJA ZA RAZVOJ PLATFORME 35

3.2. PRIKUPLJANJE I SNIMANJE PODATAKA 39

3.2.1. STRUKTURA UČESNIKA 39

3.2.2. PRIPREMA OKRUŽENJA ZA SNIMANJE 40

3.2.3. PROCEDURA SNIMANJA 41

3.2.4. SNIMANJE GESTOVA 30 SLOVA AZBUKE 43

3.3. REZULTATI SNIMANJA 44

3.3.1. SNIMLJENI VIDEO ZAPISI 45

3.3.2. MATEMATIČKA OSNOVA BAZE PODATAKA 47

3.3.3. ZAPISIVANJE KLJUČNIH TAČAKA SA VIDEO SNIMAKA U BAZU PODATAKA 51

3.3.4. KONTROLA I PROVERA PODATAKA 52
3.3.5. KONAČNI OBLIK I VELIČINA BAZE PODATAKA 59
3.4. DISKUSIJA REZULTATA SNIMANJA GESTOVA KOJI PREDSTAVLJAJU SLOVA AZBUKE SRPSKOG ZNAKOVNOG JEZIKA 60

4. KREIRANJE MODELA MAŠINSKOG UČENJA 63

4.1. TEORIJSKA OSNOVA MAŠINSKOG UČENJA 63
4.1.1. ARHITEKTURA MODELA 65
4.1.2. SPREČAVANJE PREVELIKOG PRILAGOĐAVANJA MODELA PODACIMA 65
4.1.3. OPTIMIZOVANJE HIPER-PARAMETARA MODELA 65
4.1.4. METODE NAPREDNIH TEHNIKA 66
4.1.5. PROCENA RADA MODELA 66
4.1.6. UPOREDNA ANALIZA RAZLIČITIH MODELA 67
4.1.7. PRINCIPI POREĐENJA PERFORMANSI KREIRANIH MODELA 68
4.2. SPROVOĐENJE EKSPERIMENTA KORIŠĆENJEM RAZLIČITIH MODELA MAŠINSKOG UČENJA 85
4.3. ANALIZA DVA REPREZENTATIVNA MODELA ZASNOVANA NA JEDNODIMENZIONALNIM I DVODIMENZIONALNIM PODACIMA 88
4.3.1. ANALIZA PARAMETARA PRI OBUČAVANJU MODELA KOJI SE BAZIRA NA JEDNODIMENZIONALNIMA PODACIMA 90
4.3.2. ANALIZA PARAMETARA PRI OBUČAVANJU MODELA KOJI SE BAZIRA NA JEDNODIMENZIONALNIMA PODACIMA 91

5. PRAKTIČNA PRIMENA KREIRANOG MODELA MAŠINSKOG UČENJA 94

5.1. IZBOR MODELA ZA SPROVOĐENJE EKSPERIMENTA 94
5.2. IZBOR PARAMETARA ZA OBUČAVANJE ODABRANOG DVODIMENZIONALNOG MODELA MAŠINSKOG UČENJA 99
5.2.1. BROJ ITERACIJA (EPOCHS) 99
5.2.2. VELIČINA PAKETA PODATAKA (BATCH_SIZE) 100
5.3. POTENCIJALNE IMPLEMENTACIJE MODELA MAŠINSKOG UČENJA 103
5.3.1. HARDVER ZA MODEL MAŠINSKOG UČENJA 103
5.3.2. SOFTVER ZA MODEL MAŠINSKOG UČENJA 103
5.3.3. LOKACIJE ZA POSTAVLJANJE MODELA 104
5.3.4. MOGUĆE APLIKATIVNE PRIMENE 104
5.4. PYTHON KOD ZA KORIŠĆENJE MODELA MAŠINSKOG UČENJA PRI PREPOZNAVANJU GESTOVA SLOVA SRPSKOG ZNAKOVNOG JEZIKA 106

5.5. PRAKTIČNA UPOTREBA I REZULTATI RADA APLIKACIJE ZA PREPOZNAVANJE GESTOVA SRPSKOG ZNAKOVNOG JEZIKA 109

6. ZAKLJUČAK 115

6.1. HIPOTEZE 115
6.2. KOMPLEKSNOST ISTRAŽIVAČKOG PROCESA: MOGUĆNOSTI, OGRANIČENJA, RIZICI, PREDNOSTI I NEDOSTACI 118
PRAVCI BUDUĆIH ISTRAŽIVANJA 120
6.3. STRUKTURNI ASPEKTI NAUČNOG ISTRAŽIVANJA: DESKRIPCIJA, KLASIFIKACIJA, OTKRIĆE I OBJAŠNJENJE 122

LITERATURA 123

PRILOZI 131

PRILOG A. VERZIJE INSTALIRANIH PYTHON MODULA U KREIRANOJ APLIKACIJI 131

SPISAK AKRONIMA 132

SPISAK SLIKA 133

SPISAK TABELA 136

1. Uvod

Slušno oštećena zajednica sa srpskog govornog područja suočava se sa nedostatkom podrške tehnološke naučne zajednice, čiji radovi i aktivnosti su manje usmereni ka razvoju interaktivnih alata koji bi olakšali svakodnevni život ove zajednice. Glavni problem s kojim se suočavaju je nedostatak mogućnosti za efikasnu komunikaciju, što im umanjuje kvalitet života. Postojeća tehnološka rešenja nisu pružila korisnicima alat za interakciju korišćenjem srpskog znakovnog jezika.

Ova disertacija za jedan od primarnih ciljeva ima olakšanje komunikacije i razmene informacija između slušno oštećene zajednice koja koristi znakovni jezik i dela zajednice koji je ne koriste. Ovakvo tehnološko rešenje može značajno da unapredi socijalni i poslovni aspekt života slušno oštećene zajednice (Radaković M. et al. 2024). Doprinos unapređenju interaktivne edukacije, poboljšanja opšte komunikacije i proširenja učešća ove zajednice u poslovnim aktivnostima postavljeni su kao dodatni ciljevi. Angažovanjem oko 45 studenata, profesora i nastavnika specijalne škole, kreirana je obimna baza podataka zasnovana na pokretima srpskog znakovnog jezika.

Evaluacijom različitih modela mašinskog učenja, izvršena je selekcija najpreciznijih i najpouzdanijih, poređenjem rezultata koje generišu. Funkcionalnost i karakteristike usvojenog i analiziranog modela su detaljno predstavljene u finalnoj fazi sprovođenja eksperimenta. Kreiran je računarski program koji demonstrira praktičnu primenu modela mašinskog učenja, omogućavajući prepoznavanje gestova srpskog znakovnog jezika i prikazivanje prognoziranih rezultata. Kreirani kod predstavlja ilustraciju mnogobrojnih mogućnosti daljeg razvoja aplikacija koje se mogu bazirati na generisanoj bazi podataka. Na osnovu izloženog, rad predstavljen u ovoj disertaciji obezbeđuje unapređenje mogućnosti slušno oštećenih da usavrše svoje veštine u komunikaciji, omogući bolju komunikaciju sa čujućom zajednicom i uključi ih ravnopravno u svakodnevne i poslovne aktivnosti.

1.1. Predmet istraživanja, hipoteze i cilj

1.1.1. Problem istraživanja

Realni problemi slušno oštećene zajednice počinju od njihovog rođenja, nastavljaju se kroz školovanje i prate ih do kraja života (Lilly, J.P., 2021). Jedan od

osnovnih načina komunikacije slušno oštećene zajednice sa okolinom je pomoću znakovnog jezika. Da bi komunikacija mogla biti uspostavljena, obe strane (i čujući i slušno oštećeni) treba da razumeju prikazane gestove (Munoz-Baell, I. M., & Ruiz, M. T., 2000). Dodatni problem predstavlja i to što znakovni jezik nije standardizovan na internacionalnom nivou, pa svaka država na svojoj teritoriji razvija sopstvene standarde. Čak i u okviru jedne države dolazi do manjih ili većih razlika u gestovima kojima se opisuju reči ili pojedinačna slova. Na srpskom govornom području nije posvećeno dovoljno pažnje razvoju interaktivnih programa ili baza podataka koji bi za cilj imali tumačenje ili prepoznavanje pokazanih gestova. Gestovi mogu predstavljati reči ili samo slova azbuke. U standardnom korišćenju znakovnog jezika, slova azbuke se ne koriste, već se gestovima koji predstavljaju reči kreiraju rečenice. Ukupan broj reči u srpskom znakovnom jeziku je između 10.000 i 20.000. Slova azbuke se koriste u situacijama kada je potrebno precizno prikazivanje neke reči – ime, inicijale ili slično.

Razvoj veštačke inteligencije u širem smislu, kao i povećana dostupnost gotovih programskih modela za prepoznavanje ključnih tačaka na telu i rukama, omogućuje lakši razvoj i implementaciju sistema za prepoznavanje gestova znakovnog jezika. Ipak, rešavanje problema slušno oštećene zajednice počinje sa kreiranjem obimne, kvalitetne, relevantne i klasifikovane baze podataka koja sadrži pokrete znakovnog jezika.

1.1.2. Ciljevi i zadaci istraživanja

Ovaj rad ima za cilj da pomogne slušno oštećenoj zajednici da omogući efikasniju komunikaciju i razmenu informacija i time unapredi socijalni aspekt održivog života i poslovanja. Slušno oštećena zajednica se suočava sa mnogim problemima, a jedan od najvećih je korišćenje tehnologije i komunikacija sa drugima (Aldalur A., 2021). Znakovni jezici su specifični i zbog nepostojanja univerzalnog, svetski priznatog jezika, regionalne i lokalne zajednice slušno oštećenih ljudi su prinuđene da se snalaze na različite načine. Ključna svrha ovog rada je da obezbedi odgovarajuće podloge za povećanje stepena uključenosti ove grupe ljudi u što širi, po mogućstvu neograničen spektar poslovanja.

Osnovni cilj je kreiranje modela baziranog na mašinskom učenju za prepoznavanje srpskog i njemu sličnih znakovnog jezika. Pored toga, dodatni ciljevi

usmereni su na korišćenje razvijenog modela za različite namene: na mobilnim telefonima, u računarima, konferencijskim salama, itd. Poseban cilj je da se omogući modifikacija i nadgradnja razvijenog modela za implementaciju u slične i druge svrhe.

Zadatak je da se dođe do funkcionalnog računarskog programa koji će biti sposoban da prepoznaje gestove slova azbuke srpskog znakovnog jezika, upotrebljivog u praksi. Ovakav program treba da bude fleksibilan, kompatibilan sa većinom današnjih računara i primenljiv u realnim svakodnevnim okolnostima.

1.1.3. Hipoteze u istraživanju

Opšta hipoteza od koje se krenulo u istraživanje u disertaciji je: „korišćenjem alata baziranim na veštačkoj inteligenciji može se unaprediti i olakšati komunikacija ljudi oštećenog sluha“.

Posebna hipoteza koja proizilazi iz opšte je: „modeli mašinskog učenja mogu značajno da pomognu u razvoju algoritama za prepoznavanje znakovnog jezika“.

Pojedinačne hipoteze koje su korištene u disertaciji su:

1. Smanjuje se mogućnost greške u komunikaciji sa slušno oštećenim osobama
2. Unapređuje se položaj slušno oštećenih osoba u društvu
3. Proširuje se spektar delatnosti u kojima se mogu efikasno angažovati ljudi oštećenog sluha

1.1.4. Metode istraživanja

U toku naučnog i istraživačkog rada upotrebljene su različite metode kako bi bili zadovoljeni osnovni metodološki zahtevi – objektivnost, pouzdanost, opštost i sistematičnost. U skladu sa izabranom problematikom, definisanim ciljevima istraživanja i postavljenim naučnim hipotezama radi definisanja naučnih i stručnih zaključaka i iznalaženja mogućih rešenja upotrebljeno je mašinsko učenje uz korišćenje rezultata istraživanja iz savremene naučno-stručne literature, odnosno saznanja naučnika i drugih istraživača koji su analizirali problematiku kojom se bavi i ovaj rad.

Kandidat će u Disertaciji istraživati naučno-teorijska saznanja, relevantnu literaturu i savremenu poslovnu praksu korišćenjem većeg broja metoda, i to:

deskriptivne, induktivne i deduktivne metode, metode analize i sinteze, metode generalizacije i specijalizacije, metode dokazivanja i opovrgavanja, metode kompleksnog posmatranja i analize sadržaja, kao i metod studije slučaja.

Primenom eksperimentalnog metoda analizirana je preciznost predviđanja kreiranog modela. Nakon utvrđivanja preciznosti, sprovedene su mere za koje je pretpostavljeno da mogu povećati tačnost prepoznavanja znakovnog jezika. Nakon primene ovih mera ponovo je procenjivana tačnost modela u prepoznavanju znakovnog jezika sa slika. Na ovaj način primenjene su metoda analize i sinteze, kao i metoda dokazivanja i opovrgavanja.

Pošto su primećene značajne greške u tumačenju znakovnog jezika osoba oštećenog sluha, razvoj i implementacija odgovarajućih modela baziranih na veštačkoj inteligenciji bila bi od izuzetnog značaja za unapređenje položaja ovih osoba u široj društvenoj zajednici. Između ostalog, jedan od ključnih benefita u razvoju ovih i sličnih modela, bio bi omogućavanje ljudima sa oštećenim sluhom da učestvuju u video konferencijama, tako što bi se njihovi gestovi automatski prevodili u tekst. S tim u vezi, značaj sprovedene studije slučaja je da ukaže na efikasnost korišćenja znakovnog jezika uz podršku odgovarajućeg razvijenog modela.

1.1.5. Naučna i društvena opravdanost istraživanja

U svetlu naprednih tehnologija industrije 4.0, razvijene zemlje su se u poslednje vreme naročito fokusirale na socijalni aspekt razvijenog poslovanja. U nastojanju da humanizuju uslove rada i obezbede ekološki održivo okruženje, vodeće globalne sile, poput Amerike i Kine su fokus stavile na intenziviranje uključenja osoba sa posebnim potrebama u sve sfere održivog poslovanja. Imajući u vidu da su navedeni i drugi razvijeni regioni uveli čak i takmičenja u kreiranju modela za prepoznavanje znakovnog jezika, zemlje u razvoju još su uvek u fazi razvoja svojih lokalizovanih sistema za prepoznavanje znakovnih jezika. Kako globalizacija održivog poslovanja nameće inkluziju osoba sa posebnim potrebama u skoro sve poslovne sfere, razvoj računarskih sistema koji im omogućavaju ovu inkluziju je od nesagledivog socijalnog značaja.

U ovom istraživanju, razvijeni model služi da pretvori slike u čitljiv znakovni jezik. Razvijeni model je baziran na jedinstvenom skupu podataka srpskog

znakovnog jezika. Ovaj skup prethodno prikupljenih i obrađenih podataka se može naknadno koristiti i u druge namene.

Korišćena baza podataka je univerzalna osnova za kreiranje različitih programa i aplikacija za edukaciju i obučavanje svih uzrasta osoba sa oštećenim sluhom. Takođe, može poslužiti i kao osnova drugim slovenskim jezicima, koji bi uz manje modifikacije mogli da ga koriste za svoje potrebe. Jedan od značaja razvijenog modela je mogućnost i njegove integracije sa sličnim modelima namenjenim da unaprede socijalni položaj osoba oštećenog sluha.

Razvoj mobilnih aplikacija za prepoznavanje znakovnog jezika za različite korisničke aplikacije. Razvoj računarskih programa i aplikacija za pomoć u obrazovanju i učenje svih starosnih struktura znakovnom jeziku.

Razvoj adekvatnog hardvera koji bi uz pomoć ovog modela mogao nezavisno da prepoznaje znakovni jezik i omogući ostvarivanje individualne komunikacije između korisnika i osoba oštećenog sluha. Jedan od tipičnih primera su pametne naočare.

Tema doktorske disertacije je aktuelna, a doprinos predloženog istraživanja je u ukazivanju na značaj korišćenja savremenih tehnologija u unapređenju položaja osoba sa posebnim potrebama u širu društvenu zajednicu.

U skladu sa predmetom i ciljem, postavljenim pretpostavkama i metodama istraživanja, očekuje se da će rezultati ovog naučno-istraživačkog rada dati doprinos napretku Srbije u ovoj oblasti, jer se značajni rezultati u ovoj oblasti tek očekuju. Očekivani rezultat, odnosno ishod koji će se dobiti nakon sprovedenog istraživanja jeste da će se postavljene hipoteze potvrditi i dokazati sa najmanje 80% tačnosti.

1.2. Sadržaj disertacije

Tokom rada, prateći predmet i ciljeve disertacije, korišćene su osnovne naučne istraživačke metode: tehnike prikupljanja podataka, metoda analize i sinteze, analitička, kao i induktivno-deduktivna. Istraživanje je sastavljeno iz nekoliko ključnih delova.

Početni deo rada detaljno opisuje postupak kreiranja baze podataka koja sadrži gestove slova srpskog znakovnog jezika. Baza predstavlja ključni i najvažniji deo disertacije, pošto se nad njom može nadgraditi veliki broj drugih modela

mašinskog učenja i primeniti u različite svrhe. U okviru toga, objašnjeno je koji su okviri za razvoj korišćeni, na koji način su slike za obučavanje modela prikupljene, obrađene i prilagođene za dalje korišćenje. Na osnovu pripremljene baze podataka sa slikama, definisane su i prikupljene ključne tačke delova ruke (šake). Sve prikupljene slike su potom klasifikovane prema značenju i obeležene, pri čemu je pripremljena osnova za obučavanje modela mašinskog učenja.

Centralni deo rada, detaljno opisuje algoritam izrade modela baziranog na mašinskom učenju za prepoznavanje znakovnog jezika iz slika i njihovu transformaciju u slova. Podaci iz baze su podeljeni u grupe za obučavanje i testiranje, na osnovu kojih je kreiran model. Nakon obučavanja modela, sprovedeno je testiranje u cilju verifikacije dobijenih rezultata. S obzirom da je ovaj deo rada važan za finalno funkcionisanje programa, poseban akcenat je stavljen na razvoj modela. Performanse i karakteristike analiziranih modela se upoređuju i sprovodi se selekcija.

Poslednji deo rada je vezan za testiranje modela na realnoj studiji slučaja slušno oštećenih osoba. Kreiran je računarski program koji praktično koristi prethodno projektovan model mašinskog učenja. Predstavljena je i analiza performansi ovakvog programa u praksi.

U zaključku su date mogućnosti razvijenog modela sa pravcima njegovog potencijalnog unapređenja. Takođe, predstavljena je i mogućnost proširenja baze podataka, tako da pored slova može da prepoznaje i druge gestove. Zbog univerzalnosti celokupnog postupka, korišćenjem prikazanog istraživanja, moguće je primeniti postupak na bilo koji drugi znakovni jezik.

2. Srpski znakovni jezik i mašinsko učenje

U ovom poglavlju opisuje se srpski znakovni jezik kao sredstvo komunikacije gluvih, kao i moguća primena računara i mašinskog učenja u svrhu prepoznavanja gestova. Takođe se prikazuje i pregled dosadašnjih aktivnosti i dostignuća naučne zajednice koja se bavila tom temom.

2.1. Istorijat nastanka i karakteristike srpskog znakovnog jezika

Znakovni jezik predstavlja vrstu neverbalne komunikacije u kojoj se pomoću ruku, tela i lica izražavaju reči i pojmovi (Mijatović, S, 2019). U Parizu je 1984. godine, na skupštini UNESCO precizno definisan znakovni jezik: „Znakovni jezik je vid komunikacije, kojim se reči i pojmovi iskazuju rukama, kojom prosečan pojedinac lako izražava reči, misli i celine, zavisno od konteksta ili složene serije ideja" (Dimić N., 2018). Definicija znakovnog jezika po Savić i Ivanović (1996) glasi: „Znakovni jezik je sistem znakova koji se ostvaruje uslovnim pokretima ruku koji su obilno propraćeni mimikom u cilju razgovora gluvog sa svojom okolinom“.

Gestovi znakovnog jezika su nosioci i činioci pojedinih misli i reči, a uklopljeni u neprekidan pokret, oni predstavljaju rečenice različitih tipova i stepena kompleksnosti. Znakovni jezik čine svi pokreti tela, lica i ruku pomoću kojih osoba iskazuje svoje misli. Svi ti pokreti zajedno čine znakovni jezik i ne mogu se razdvajati (Dimić, D. N. 2002). Pored samih reči i rečenica, znakovni jezik u sebi nosi i emociju, afektivno stanje, radnu operaciju ili predmet. Za termin znakovni jezik se zbog svojih dinamičkih fizičkih karakteristika neretko koristi i izraz „kinetički govor“.

Prolaznost je karakteristika znakovnog jezika, ali se za razliku od muzike, koja je takođe prolazna, ovaj jezik ne može sačuvati u obliku nota, pisma ili slika. Moguće je samo snimiti ga u obliku video zapisa, pa ga naknadno reprodukovati. Znakovni jezik je glavno sredstvo za komunikaciju među gluvim osobama, pored govornog i pisanog. Smatra se legitimnim i primarnim lingvističkim sredstvom za komunikaciju kod gluvih i osoba sa oštećenjem sluha. Zbog velikog uticaja na razvoj kognitivnog sistema kod gluvih, znakovni jezik je činjenično od velikog značaja za intelektualni i socijalni razvoj osoba oštećenog sluha. Gluve osobe su vizuelni tipovi, pa najveći deo ličnih saznanja iz spoljnog sveta doživljavaju putem vida. Kao primarni sistem komunikacije, od najveće je važnosti da bude dobro razvijen i što je više moguće na standardnim osnovama postavljen. Pri komunikaciji odraslih osoba sa decom, važno

je da sve verbalne poruke odgovaraju i budu usklađene sa neverbalnim postupcima (Colić, V., 2008).

Naučna zajednica koja se bavi istraživanjima u oblasti znakovnog jezika stavlja fokus na ispitivanje odnosa govornog jezika sa znakovnim. Neke od važnijih tema su i uticaj i posledice po jezički razvoj dece usled boljeg ili lošijeg poznavanja znakovnog jezika. Istraživači se bave i analizom kako stepen i obim znanja znakovnog jezika utiče na opšte obrazovanje i razvoj deteta (Gatalica, A., 2011).

Srpski znakovni jezik, kao i drugi slični svetski jezici su u neprekidnom razvoju. Pošto je sačinjen od sistematski struktuiranih i organizovanih gestova, uporediv je sa prirodnim jezicima. Naučna i zajednica gluvih kontinuirano ulažu napore u standardizaciju znakovnog jezika, ali i pored velikog uloženog truda, postoje primetne razlike unutar države, oblasti, grada ili porodice. Tako su gestovi često obojeni lokalizmima, pa se razlikuju u zavisnosti od područja gde se primenjuju (Savić, Lj. 2002).

Znakovni jezici su u svojoj osnovi prirodni, pa omogućavaju sveobuhvatnu komunikaciju među ljudima. Posebnu važnost ovakva komunikacija nosi kod dece u razvoju, zbog njihovog emocionalnog, socijalnog i kognitivnog razvoja. Kod gluve dece, znakovni jezik mora biti što ranije usvojen kao prvi ili eventualno drugi jezik, pošto tako omogućava kompletnu komunikaciju i razvoj (Humphries T., 2022). Omogućava komunikaciju deteta sa roditeljima od najranijeg uzrasta i igra značajnu ulogu u socijalnom i emocionalnom razvoju. Ukoliko je dete savladalo znakovni jezik na pravi način i može ga koristiti za komunikaciju u potpunosti, to će mu omogućiti uklapanje u svet gluvih, ali i mogućnost da usvoji govorni jezik u pisanom ili čak govornom obliku (Kovačević, T., 2020). Kad gluvo dete usvoji jezik znakova i razvije dobru lingvističku osnovu kroz njegovo prirodno usvajanje, to znanje se može iskoristiti za učenje drugog, teže savladivog, govornog jezika (Kovačević, T. et al., 2019).

Jedan od značajnijih doprinosa ovog naučnog rada se ogleda u razvoju sistema za pomoć u učenju srpskog znakovnog jezika. Takođe, kreirana je kvalitetna baza podataka sa zabeleženim gestovima slova srpske Azbuke koja se može koristiti kao osnova za razvoj modela mašinskog učenja različitih namena. Na osnovu nje, mogu se kreirati obrazovni interaktivni sadržaji koji pomažu učenicima da prepoznaju ili reprodukuju odgovarajuće pokrete. Pored klasičnih obrazovnih programa, baza se

može iskoristiti i za kreiranje interaktivnih računarskih igara koje bi se koristile za obučavanje gluve dece da pokazuju gestove srpske Azbuke. Obučavanje gluvih se time ne prenosi iz škole kući, već se učenicima nudi mogućnost da na standardnom laptop računaru vežbaju gestove srpskog znakovnog jezika i u vanškolskom okruženju i unaprede svoje znanje i sposobnosti. Primene ovakve baze podataka i sistema mašinskog učenja su raznovrsne i pored obrazovnih, mogu se koristiti i u poslovne namene, kao i u programima za unapređenje komunikacije. Moguće je inkorporirati sistem zasnovan na prepoznavanju gestova znakovnog jezika u komunikacione programe kao što su Viber, WhatsApp, Webex, Teams, itd. Pored toga, ovakvi sistemi se mogu ugraditi i u video konferencijske uređaje koje proizvode velike svetske firme kao što su Cisco, Polycom, Microsoft, Logitech, itd.

2.2. Pregled odabranih istraživanja iz oblasti prepoznavanja znakovnih jezika

U ovom poglavlju su tabelarno prikazani odabrani radovi koji odgovaraju predmetu naučnog istraživanja u ovoj doktorskoj disertaciji. Odabrani i analizirani radovi u osnovi imaju kreiranje baza podataka i razvoj i implementaciju aplikacija za različite znakovne jezike. Pored radova kojima je u fokusu kreiranje, opisivanje i analiza baza podataka, u tabeli Tabela 1 prikazani su i radovi koji se bave kreiranjem modela mašinskog učenja koji za svoju osnovu koriste pomenute baze. U pitanju su različiti svetski jezici, veće i manje baze podataka, sa različitim brojem učesnika i metodologija prikupljanja snimaka i gestova. Radovi iz tabele Tabela 1 i drugi razmatrani radovi, korišćena su kao osnova za ovo istraživanje.

Tabela 1. Prikaz naučno istraživačkih radova iz oblasti baza podataka znakovnih jezika

Albanie, S. Et al. (2021) *BBC-Oxford British Sign Language Dataset*	**Opis istraživanja:** U ovom radu je prikazan opis, razvoj i evaluacija baze podataka za britanski znakovni jezik. Osnova za kreiranje ovog seta podataka su emisije sa televizijskog programa emitovanog na BBC-u. Snimci su se koristili zajedno sa titlovima, koji odgovaraju zvučnom sadržaju obrađenih epizoda. Postupak pripreme podataka počinje podelom video materijala na pojedinačne epizode, njihovo grupisanje, a potom i kategorizacija po tipovima emisija. U delu video materijala se pojavljuje demonstrator koji prikazuje

	gestove za izgovoreni tekst uz prethodno obezbeđen titl. Naknadno se iz video zapisa izdvaja samo demonstrator čiji se pokreti koriste za kreiranje baze podataka. Istraživači su koristili *OpenPose* sistem za prepoznavanje položaja lica, tela i ruku (Z. Cao, 2019). **Rezultati istraživanja:** Pošto interpretator prikazuje prevod izgovorenih reči, prikazani gestovi ne odgovaraju direktno tekstu, već se beleži kontekst i smisao rečenice. Na osnovu toga, zaključuje se da prikazani gestovi ne predstavljaju transkripciju teksta već približni prevod smisla rečenica. Korišćena rezolucija od 444 × 444 tačaka za interpretatora znakovnog jezika ima ograničenja u smislu kvaliteta prepoznavanja manjih pokreta, ali zbog dužine snimaka nije bilo moguće efikasno koristiti veće rezolucije. Rečenice znakovnog jezika se polu-automatski (potrebna je korekcija istraživača) usklađuju sa izgovorenim i titlovanim tekstom. Jedan od većih izazova su istraživači imali u sinhronizaciji znakovnog jezika sa izgovorenim rečenicama i tekstom koji im odgovara. Zbog toga su pribegli manuelnoj anotaciji video zapisa, sa mogućnošću pomeranja i usklađivanja sa odgovarajućim audio i tekstualnim zapisom. Izvučeno je preko 1.2 miliona rečenica iz video snimaka sa titlovima na engleskom jeziku, što je omogućilo obimno istraživanje. Sa oko 1467 sati video materijala u kojima je učestvovalo 39 tumača znakovnog jezika, ova baza predstavlja jednu od najvećih sa kontinuiranim snimcima znakovnog jezika.
G. Akhil Krishna & G. Sandeep (2022) *Sign language recognitionusing deep cnn*	**Opis istraživanja:** Istraživanje opisano u dokumentu fokusira se na prepoznavanje znakovnog jezika korišćenjem dubokih konvolucijskih neuronskih mreža (CNN), istražujući kako se CNN može koristiti za identifikaciju gestova rukom. Posebno, rad istražuje prepoznavanje američkog znakovnog jezika (ASL) koristeći slike gestova rukom koje su snimljene web kamerom. Istraživači su koristili tehniku segmentacije boje kože u HSV postavci boja u prostoru za izdvajanje ruku na slikama, a zatim primenili različite tehnike računarskog vida

	za obradu slika pre klasifikacije pomoću CNN. Istraživanje koje je predstavljeno u priloženom radu koristilo je video klipove za prepoznavanje znakovnog jezika. Tokom faze pripreme snimaka, ulazni video materijal je pretvoren u niz RGB slika. Da bi se prilagodili originalnom modelu za video tokove od po 16 slika, korišćeno je linearno uzimanje uzoraka kako bi svaki video imao potpuno istu dužinu. **Rezultati istraživanja:** Pokazali su visok nivo tačnosti prepoznavanja, premašujući 90% tačnost na korišćenoj bazi podataka, što ukazuje na visoku efikasnost predložene metode. Ovakvi rezultati naglašavaju potencijal primene dubokog učenja u prepoznavanju znakovnog jezika, čime se otvara put ka razvoju naprednih sistema za komunikaciju između osoba sa oštećenjem sluha i šire zajednice.
V. Athitsos, et al. (2008) *The American Sign Language lexicon video dataset*	**Opis istraživanja:** Jedan od prvih video setova podataka koji se sastoje od zapisa koji prikazuju gestove američkog znakovnog jezika. Zbog svoje dostupnosti na internetu, vremenom se razvijao i proširivao, povećavajući svoju funkcionalnost i upotrebljivost. Baza podataka služi kao osnova za razvoj sveobuhvatnijeg sistema za prepoznavanje znakova američkog znakovnog jezika. **Rezultati istraživanja:** Pored svoje primarne namene, za prepoznavanje gestova američkog znakovnog jezika, okvir ovog istraživanja omogućava naučnoj zajednici da ga upotrebe i za druge primene vezane za računarsku viziju i prepoznavanje video zapisa. Baza podataka je posebno namenjena za analiziranje gestova i komunikaciju među ljudima.
H. R. V. Joze, O. Koller (2019) *MS-ASL: A large-scale data set and benchmark for understanding American Sign Language*	**Opis istraživanja:** Ovaj rad dva autora iz Microsoft-a predstavlja veliki skup podataka i osnovu za razumevanje američkog znakovnog jezika (ASL). Rad takođe predstavlja I3D arhitekturu kao efikasnu za prepoznavanje znakovnog jezika, nadmašujući u rezultatima veliki broj trenutno korišćenih metoda. I3D metoda je često korišćena za video klasifikaciju, a poznata je po moćnoj i stabilnoj arhitekturi za

	prepoznavanje znakovnih jezika. **Rezultati istraživanja**: Ovaj skup podataka obuhvata preko 25.000 anotiranih video snimaka i omogućava istraživanje funkcionisanja ove metode sa podacima koji nisu direktno vezani za demonstratore. Ovo istraživanje se može vršiti u realističnim uslovima. Angažovano je preko 200 pojedinačnih učesnika – pokazivača gestova američkog znakovnog jezika. Kreirana baza podataka je javno dostupna svima. Obrađeno je 1000 različitih gestova znakovnog jezika u uslovima koji su univerzalni i koji se lako mogu ponoviti.
O. M. Sincan and H. Keles (2020) *A large scale multi-modal Turkish Sign Language dataset and baseline methods*	**Opis istraživanja:** Cilj ovog istraživanja je kreiranje velike baze podataka gestova turskog znakovnog jezika, koji bi predstavljao osnovu za ralji razvoj sistema za njihovo prepoznavanje. Snimljeni znakovi pored ruku i tela pokazivača sadrže i veliki broj raznovrsnih pozadina i okruženja, čime se pokriva veliki broj različitih scenarija, a modeli zasnovani na ovoj bazi postaju otporniji na raznolikosti. Korišćena je kamera Microsoft Kinect v2, zbog čega snimljeni zapisi sadrže pored informacije o boji i informacije o dubini. **Rezultati istraživanja**: Kreirana je baza koja sadrži 226 različitih gestova, a ona je kreirana uz pomoć 43 različita demonstratora. Ukupno je kreirano 38.336 različitih, pojedinačnih snimaka. Pošto je korišćena specijalna kamera, koja može da izmeri udaljenost snimljenih objekata, kreirana baza sadrži precizne podatke, ali je mana u tome što se za korišćenje ovakve baze, takođe mora koristiti isti uređaj. Korišćenjem ove baze podataka, pomoću kreiranog modela mašinskog učenja, ostvarena je maksimalna preciznost od oko 96% tačno prepoznatih gestova turskog znakovnog jezika. Dodatno, uočen je i problem pogrešne klasifikacije nekih gestova koji imaju sličnosti sa drugim gestovima.
Kagirov, I. et al. (2020) TheRuSLan: Database of	**Opis Istraživanja:** Istraživanje predstavlja novu multimedijsku bazu podataka ruskih znakovnih jezika TheRuSLan. Baza sadrži leksičke jedinice (pojedinačne reči i

Russian Sign Language	fraze) iz oblasti "prehrambeni proizvodi u supermarketu" i prikupljena je pomoću Microsoft Kinect uređaja sa korišćenjem režima dubinske mape. Ruski znakovni jezik ima zvaničan status u Rusiji i koristi ga preko 120.000 gluvih osoba u Rusiji i susednim zemljama. Cilj baze podataka je stvaranje automatskog sistema za prepoznavanje ruskih znakovnih jezika. **Rezultati Istraživanja:** Baza podataka može biti proširena da obuhvati više leksičkih jedinica a eksplicitno je kreirana za zadatke prepoznavanja ruskih znakovnih jezika. Autori su formulisali osnovne principe anotacije znakovnih reči na osnovu prikupljenih podataka i otkrili sadržaj prikupljene baze podataka. Baza pruža nove mogućnosti za leksikografski opis vokabulara ruskih znakovnih jezika i unapređuje istraživanja u oblasti automatskog prepoznavanja gestova.

Tabela 2. Prikaz naučno istraživačkih radova iz oblasti mašinskog učenja i znakovnih jezika

Papastratis, I. et al. (2021) *Artificial Intelligence Technologies for Sign Language*	**Opis istraživanja:** U okviru ovog preglednog istraživanja, prikazan je pregled velikog broja radova koji se bave različitim oblastima vezanim za znakovne jezike i mašinsko učenje. Kreiranjem bazama podataka znakovnih jezika, uređajima za snimanje pokreta, razlikama u načinu prepoznavanja gestova znakovnih jezika (kontinualno prepoznavanje gestova ili izolovano prepoznavanje pojedinačnih gestova), tumačenju i prevođenju gestova u govorni jezik (modeliranjem gramatike i strukture), vizuelnim predstavljanjem i prikazivanjem gestova pomoću trodimenzionalnih avatara, realističnih ljudskih figura, itd. **Rezultati istraživanja:** Prikazana su rešenja vezana za različite znakovne jezike, kao što su: Britanski, Indijski, Korejski, Američki engleski, Grčki, Tajlanđanski, itd. Pomenuti su i korišćeni principi i metode za prikupljanje podataka vezanih za znakovne jezike, na način na koji će odgovarati za upotrebu u sistemima veštačke inteligencije i mašinskog učenja u užem smislu. Opisani u i metodi i uređaji

	za prikupljanje informacija pomoću različitih vrsta nosivih senzora, ali i pomoću računarskih i kamera na mobilnim uređajima. Analizirani su i principi razvoja sistema mašinskog učenja i različiti primenjeni pristupi. Zaključeno je i da većina sistema koji rade prevođenje i prepoznavanje znakovnih jezika je zasnovano na neuronskim mrežama i računarskim kamerama. Animacije i trodimenzionalni avatari se koriste za prezentaciju gestova znakovnih jezika, ali se skreće pažnja i na veliki broj izazova sa kojima se istraživači suočavaju, a vezani su za vreme, računarske performanse i krajnje rezultate koji imaju svoja ograničenja. Primetan je i značajan napredak u ovoj oblasti u drugoj polovini 2023 godine. Prikazan je veliki broj gotovih računarskih rešenja za primenu u prevođenju, učenju ili dijagnostici korisnika, a najčešće na mobilnim uređajima ili računarima.
Shin, J. et al. (2021) *Alphabet Recognition by Extracting Features from Hand Pose Estimation*	**Opis istraživanja:** Istraživanje se bavi prepoznavanjem alfabeta američkog znakovnog jezika korišćenjem slika ruku dobijenih preko web kamere. Koristi se algoritam „MediaPipe Hands“ za procenu položaja prstiju i zglobova ruke iz RGB slika, te se na osnovu tih koordinata generišu dva tipa karakteristika: distance između tačaka zglobova i uglovi između vektora i 3D osa. Za klasifikaciju (grupisanje) gestova su korišćene metode mašinskog učenja: Support Vector Machine (SVM) i Light Gradient Boosting Machine (GBM). **Rezultati istraživanja:** Rezultati istraživanja pokazuju visoku tačnost prepoznavanja znakova, i to: 99.39% za Massey bazu podataka, 87.60% za ASL Alphabet bazu i 98.45% za Finger Spelling A podatke. Predloženi sistem za automatsko prepoznavanje američkog znakovnog jezika je efikasan, pošto ne zahteva posebne senzore ili dodatne uređaje, dok pokazuje bolje performanse u poređenju sa drugim sličnim radovima.
J. Huang et al. (2019) *Attention-based 3D-CNNs for large-vocabulary sign*	**Opis istraživanja:** Istraživanje se bavi prepoznavanjem kineskog znakovnog jezika koristeći 3D konvolucijske neuronske mreže (3D-CNN) zasnovane na mehanizmu

language recognition	pažnje. Tehnika uči prostorno-vremenske karakteristike direktno iz sirovih video materijala bez „prethodnog znanja“, dok korišćeni mehanizam pomaže u selekciji ključnih elemenata. Prostorna pažnja (*attention*) integrisana je u mrežu za fokusiranje na značajne oblasti, a temporalna se koristi za izbor bitnih pokreta za klasifikaciju. **Rezultati istraživanja:** Rezultati su zasnovani na dve velike grupe podataka za znakovni jezik. Pokazuju visoku efikasnost primenjene metode u poređenju sa drugim savremenim algoritmima. Testiranje i provera rada ove metode je izvršena primenom na dve velike baze kineskog znakovnog jezika: CSL sa 500 kategorija i ChaLearn14.
Danielle, B. et al.(2019) *Sign Language Recognition, Generation, and Translation: An Interdisciplinary Perspective*	**Opis istraživanja:** U ovom radu se autori bave razvojem sistema za prepoznavanje, generisanje i prevod znakovnog jezika. Oni insistiraju na inkluziji ekspertize iz različitih oblasti poput računarske vizije, grafike, obrade prirodnog jezika, interakcije čovek-računar, lingvistike i kulture slušno oštećenih osoba. I pored potrebe za dubokim interdisciplinarnim znanjem, postojeća istraživanja su izolovana u različitim disciplinarnim oblastima i bave se odvojenim delovima ovakvih platformi za prepoznavanje znakovnog jezika. Cilj ovog rada je da premosti ove razlike i da odgovori na tri ključna pitanja koja se tiču trenutne situacije u naučnoj oblasti, najvećih izazova s kojima se oblast suočava i poziva na akciju za ljude koji rade u ovoj oblasti. Za postizanje ovih ciljeva, organizovana je dvodnevna radionica koja je okupila raznovrsnu grupu stručnjaka. **Rezultati** radionice su prikupljeni i prikazani u ovom radu, pružajući ključne informacije koje su često zanemarene od strane računara i naučnika. Prikazan je pregled aktuelnog stanja tehnike, skup najvećih izazova i poziv na akciju za istraživačku zajednicu. Glavni doprinosi ovog rada se ogledaju u pružanju orijentacije, informacija i uvida za istraživače, isticanje potreba i prilika za interdisciplinarnu

	saradnju kao i dalji smer razvoja zarad prevazilaženja problema. Posebno se naglašava potreba za uključivanjem gluve zajednice u sve faze istraživanja i razvoja kako bi se osiguralo da kreirani sistemi zaista odgovaraju potrebama korisnika.
Amangeldy, N. et al. (2022) *Sign Language Recognition Method Based on Palm Definition Model and Multiple Classification*	**Opis Istraživanja:** Istraživanje opisano u radu se fokusira na prepoznavanje znakovnog jezika koristeći model definicije dlana i linearne modele za prepoznavanje brojeva i slova kazahstanskog znakovnog jezika. Glavna inovacija predloženog algoritma je potpuno prepoznavanje 41 slova od 42 u kazahstanskom znakovnom alfabetu, što je prethodno bilo ograničeno samo na ruska slova u kazahstanskom alfabetu. Integrisana je i unificirana funkcija za podešavanje moda dubinske mape okvira, što je poboljšalo performanse prepoznavanja. **Rezultati Istraživanja:** Rezultati istraživanja pokazuju efikasnost predloženog metoda sa tačnošću prepoznavanja od oko 97%. Prednost predloženog algoritma leži u njegovoj sposobnosti da prepoznaje i jednoručne i dvoručne gestove u realnom vremenu, obezbeđujući pouzdane rezultate. Ovi rezultati potvrđuju izvodljivost pristupa i otvaraju put ka razvoju efikasnih automatskih sistema za tumačenje znakovnog jezika, koji mogu podržati efikasnu interakciju između ljudi i mašina za gluve i nagluve osobe.
Sharma, P.et. al., 2022. *Translating Speech to Indian Sign Language Using Natural Language Processing*	**Opis Istraživanja:** Istraživanje se bavi razvojem sistema za prevod govora u indijski znakovni jezik koristeći metode obrade prirodnog jezika. Cilj je omogućiti osobama s oštećenjem sluha lakšu komunikaciju sa okolinom tako što će govor biti preveden u znakovni jezik u realnom vremenu. Sistem prihvata audio i tekstualne ulaze, obrađuje ih pomoću alata za obradu prirodnog jezika, a zatim usklađuje sa video snimcima u bazi podataka koji predstavljaju odgovarajuće gestove u indijskom znakovnom jeziku. **Rezultati Istraživanja:** Sistem je pokazao visoku efikasnost u prevodu govora u znakovni jezik, sa sposobnošću da

	prepozna i pretvori složene rečenice u odgovarajuće gestove. Osim toga, uključen je i obrnuti prevod iz znakovnog jezika u govorni jezik, čime se omogućava dvosmerna komunikacija. Testiranje sistema među korisnicima pokazalo je visok nivo upotrebne vrednosti, ističući potencijal sistema da znatno poboljša interakciju između osoba sa oštećenjem sluha i šire društvene zajednice.
Kumari, D. & Anand, R.S. (2024) *Isolated Video-Based Sign Language Recognition Using a Hybrid CNN-LSTM Framework Based on Attention Mechanism*	**Opis Istraživanja:** Istraživanje je usmereno na prepoznavanje izolovanih reči znakovnog jezika iz video zapisa koristeći hibridni okvir koji kombinuje konvolucionu neuronsku mrežu (CNN) i neuronsku mrežu dugoročne kratkoročne memorije (LSTM) zasnovanu na mehanizmu pažnje. Autori su koristili MobileNetV2 kao osnovni model zbog njegove lagane strukture koja smanjuje kompleksnost arhitekture modela za izvlačenje značajnih karakteristika iz niza video frejmova. Prostorne karakteristike su obrađene LSTM neuronskom mrežom kako bi se izdvojili gestovi iz video frejmova. **Rezultati Istraživanja:** Predloženi metod je zasnovan na referentnom postojećem skupu podataka WLASL sa 100 klasa. Metodologija je postigla prosečnu tačnost od 84,65%. Rezultati eksperimenta ilustruju da je model računski efikasan u poređenju sa drugim metodama. Ovi rezultati potvrđuju da je ovakav pristup upotrebljiv u praksi i otvara put ka razvoju efikasnih automatskih sistema za tumačenje znakovnog jezika.
Teran-Quezada, A.A. et al. (2024) *Sign-to-Text Translation from Panamanian Sign Language to Spanish in Continuous Capture Mode with Deep Neural Networks*	**Opis Istraživanja:** Istraživanje je fokusirano na razvoj sistema za prevođenje znakova iz panamskog znakovnog jezika u španski tekst koristeći duboke neuronske mreže u režimu kontinuiranog snimanja. Glavni cilj rada je da omogući efikasnu komunikaciju između osoba sa oštećenjem sluha i onih koji ne koriste znakovni jezik. Sistem koristi LSTM model za rad sa dinamičkim znakovima kao sekvencijalnim podacima i uključuje detekciju akcija, što podrazumeva preciznu obradu frejmova u kojima se izvodi gest znakovnog

	jezika. Pored traženja ruku govornika, sistem uzima u obzir i lice i položaj tela jer pri komunikaciji znakovnim jezicima, osim gestova ruku, važne su i druge vizuelne karakteristike. **Rezultati Istraživanja:** Model je testiran sa preciznošću od 98,8%, što čini ovaj sistem vrednom osnovom za efikasnu komunikaciju između korisnika panamskog znakovnog jezika i korisnika španskog jezika. Rad pruža unapređenje u odnosu na trenutno stanje tehnike za prevod PSL-španskog jezika koristeći mogućnosti prevodivih znakova putem dubokog učenja. Osim toga, rad ističe važnost interdisciplinarnog pristupa koji kombinuje duboko učenje, računarsku viziju i lingvistiku za razvoj inovativnih rešenja u oblasti asistivnih tehnologija.
Gu, Y. et al. (2024) *American Sign Language Recognition and Translation Using Perception Neuron Wearable Inertial Motion Capture System*	**Opis Istraživanja:** Istraživanje se fokusira na prepoznavanje i prevod američkog znakovnog jezika (ASL) koristeći nosivi inercijalni sistem za detekciju pokreta, „Perception Neuron“. Cilj rada je bio sakupiti dataset od 300 često korišćenih ASL rečenica od tri volontera i realizovati prepoznavanje i prevod ASL rečenica u tekst, koristeći duboke neuronske mreže. Glavne komponente modela za prepoznavanje uključuju konvolucijsku neuronsku mrežu (CNN), dvosmernu dugoročnu kratkoročnu memoriju (bi-directional LSTM) i povezivanje vremenske klasifikacije (CTC). Za prevod, korišćen je enkoder-dekoder model zasnovan na LSTM. **Rezultati Istraživanja:** Model za prepoznavanje postigao je tačnost od 99,07% na nivou reči i 97,34% na nivou rečenica. Model za prevod, koristeći globalnu pažnju, ostvario je stopu greške u rečima (WER) od 16,63%. Rezultati ukazuju na visoku efikasnost predloženih metoda za prepoznavanje i prevod znakovnog jezika, ističući potencijal za razvoj pouzdanih sistema za interakciju između ljudi i mašina za gluve i nagluve osobe.
Vashisth, H.K. et al. (2023)	**Opis Istraživanja:** Istraživanje je usmereno na prepoznavanje gestova ruku u indijskom znakovnom jeziku koristeći duboke neuronske mreže, posebno CNN

Hand Gesture Recognition in Indian Sign Language Using Deep Learning	(konvolucijske neuronske mreže). Cilj rada je omogućiti komunikaciju između osoba sa oštećenjem sluha koje koriste indijski znakovni jezik i onih koji ga ne razumeju, čime se poboljšava dostupnost i komunikacija za korisnike znakovnog jezika. Sistem koristi poseban skup podataka generisan za tu namenu, koji sadrži 7800 slika, a fokusira se na razvoj mašinskog modela za prepoznavanje znakovnog jezika u realnom vremenu. Autori su prilagodili model dubokog učenja kako bi postigli maksimalne performanse, istražujući pritom i mane modela i načine njihovog prevazilaženja. **Rezultati Istraživanja:** Model je postigao gubitak od 0,0178 i tačnost od 99%, što ukazuje na visoku efikasnost predloženog pristupa u prepoznavanju gestova ruku u indijskom znakovnom jeziku. Eksperimentisanje s različitim parametrima i različitim računarima pokazalo je da slike u HSV formatu daju najbolje rezultate. Ova istraživanja su korisna i za buduća istraživanja na temu prepoznavanja znakovnog jezika.

2.3. Motivacija za razvoj modela mašinskog učenja u prepoznavanju srpskog znakovnog jezika

Veliki problemi i ograničenja u komunikaciji sa kojima se slušno oštećena zajednica suočava, sprečavaju ih u obavljanju svakodnevnih aktivnosti i poslovnih delatnosti. Nedostatak interaktivnih programa i baze podataka za srpski znakovni jezik je ukazao na potrebu razvoja obimne i kvalitetne baze podataka, kao i programa koji bi započeo proces razvoja i skrenuo pažnju tehničkog dela naučne zajednice na ovaj problem i potrebu.

Razvijanje alata i usluga koji mogu pomoći uključivanju osoba sa oštećenjem sluha u rad i svakodnevni život može doprineti njihovom povećanju kvaliteta života i smanjenju prepreka u komunikaciji sa drugima (Kothadiya D. et al., 2022). Prema podacima Svetske zdravstvene organizacije iz 2020. godine, više od 450 miliona ljudi ima oštećenje sluha (World Health Organization, 2020). Ova organizacija predviđa da će do 2050. godine oko 2.5 milijarde ljudi imati neki vid oštećenja sluha, a da će za oko 700 miliona osoba biti neophodan neki vid rehabilitacije sluha (World Health

Organization, 2024). Posledično, veliki broj ljudi širom sveta koristi znakovni jezik kao osnovno sredstvo komunikacije sa drugima. Zbog otežane komunikacije, osobe sa oštećenjem sluha suočavaju se sa brojnim preprekama kada je reč o zapošljavanju, druženju i uklapanju u širu društvenu zajednicu, i kao rezultat toga, često su isključene iz društva. Zbog toga, kao imperativ društvene održivosti i odgovornosti, ističe se potreba da se ove osobe uključe u sve segmente društvenog okruženja. Nakon što je Narodna skupština Republike Srbije usvojila "Zakon o upotrebi znakovnog jezika" 2015. godine (Zakon o upotrebi znakovnog jezika, 2015), nije bilo dovoljno pokušaja da se adresiraju pitanja koja se tiču jezičkih procesa, kao što su standardizacija srpskog znakovnog jezika, kreiranje novih udžbenika, rečnika, gramatičkih priručnika, itd (Bajić, 2021). Iz tog razloga, ukazuje se potreba za razvijanje velikog broja računarskih alata i aplikacija - kako bi se prevazišle postojeće prepreke i stvorila osnova za poboljšanje položaja osoba sa oštećenjem sluha u široj društvenoj zajednici. Prema Markoviću M. (2014) 2% populacije u Srbiji prijavilo je probleme sa sluhom bilo kog stepena. Na osnovu najnovijih dostupnih podataka iz popisa stanovništva iz 2011. godine, u Srbiji je bilo 144.648 osoba sa oštećenjem sluha. To predstavlja više od 25% populacije koja pati od bilo kog oblika invaliditeta. Prema prethodnim saznanjima, gotovo 30% stanovnika Srbije mlađih od 65 godina spada u grupu osoba sa oštećenjem sluha, što predstavlja ozbiljan društveni izazov.

3. Kreiranje baze podataka srpskog znakovnog jezika

Prepoznavanje znakovnih jezika koji koriste slušno oštećene osobe nije jednostavan zadatak, čak ni za moderne računarske sisteme koji se oslanjaju na veštačku inteligenciju i računarski vid (Fox, 2023). Ovaj problem je u oblasti računarskog vida jedan od najkompleksnijih i zahteva stručan i detaljan pristup. Uzimajući u obzir velike razlike između znakovnih jezika sa različitih govornih područja, raznolikost dijalekata u okviru samo jedne regije i države, neophodno je da se svakom znakovnom jeziku posveti posebna pažnja i kreira specijalizovana baza podataka. Kreirane baze mogu biti zasnovane na različitim osnovama i principima, ali sve one za cilj imaju prepoznavanje i beleženje gestova znakovnog jezika. Baza podataka, čije je kreiranje do detalja predstavljeno u ovom istraživanju, predstavlja čvrstu osnovu za razvoj različitih modela mašinskog učenja. Ovako kreirani modeli mogu direktno pomoći slušno oštećenoj zajednici u oblastima edukacije i u unapređenju opšte komunikacije. Predstavljeni postupak nije ograničen samo na

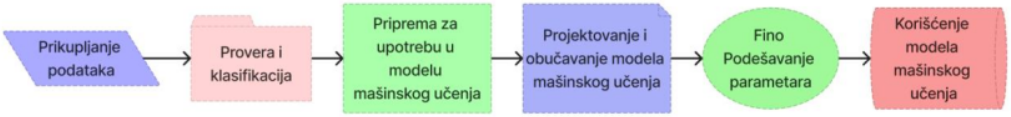
generisanje baze za slova srpske azbuke, već se može koristiti i za druge namene. Moguće je unaprediti ili izmeniti bazu podataka tako da sadrži veći broj reči ili gestove nekog drugog jezika.

Daktilologija je važan deo znakovnog jezika, a bavi se demonstracijom slova azbuke znakovnog jezika uz pomoć ruku (Isaković, L., 2023). U okviru ove teze, kreirana je baza podataka gestova 30 slova srpskog znakovnog jezika, koji se u praksi ne koriste direktno za svakodnevni razgovor između slušno oštećenih osoba. Pojedinačna slova znakovnog jezika se koriste obavezno u situacijama kada je potrebno pokazati lična imena, prezimena, geografske pojmove i pojmove za koje ne postoji određeni gest. Imajući u vidu mnoge izazove sa kojima se suočavaju osobe sa oštećenjem sluha u socijalnim okruženjima, ova disertacija pruža konkretna uputstva za izgradnju čvrstog mosta komunikacije između osoba sa oštećenjem sluha i govora i šire društvene zajednice (Ristić I. et al., 2019). Jedna od ključnih prepreka u stvaranju univerzalnih skupova podataka za znakovni jezik jeste regionalizacija tih jezika. S obzirom na značajne razlike u prikazivanju različitih jezika, dijalekata unutar regija, država i gradova, važnost stvaranja većeg broja regionalno specifičnih skupova podataka za primene znakovnog jezika je izuzetno visoka. U ovom istraživanju detaljno je opisan postupak prikupljanja, kreiranja, pripreme, obrade i analize jedinstvenog skupa podataka za SSL azbuku, s ciljem unapređenja društvenog položaja osoba sa oštećenjem sluha. Dobijeni skup podataka primenjiv je na zajednicu koja se prostire na susedne zemlje zbog mnogobrojnih jezičkih sličnosti. Sa manjim korekcijama, ovaj skup podataka može postati upotrebljiv i u zemljama koje graniče sa Srbijom.

3.1. Metodologija kreiranja baze podataka

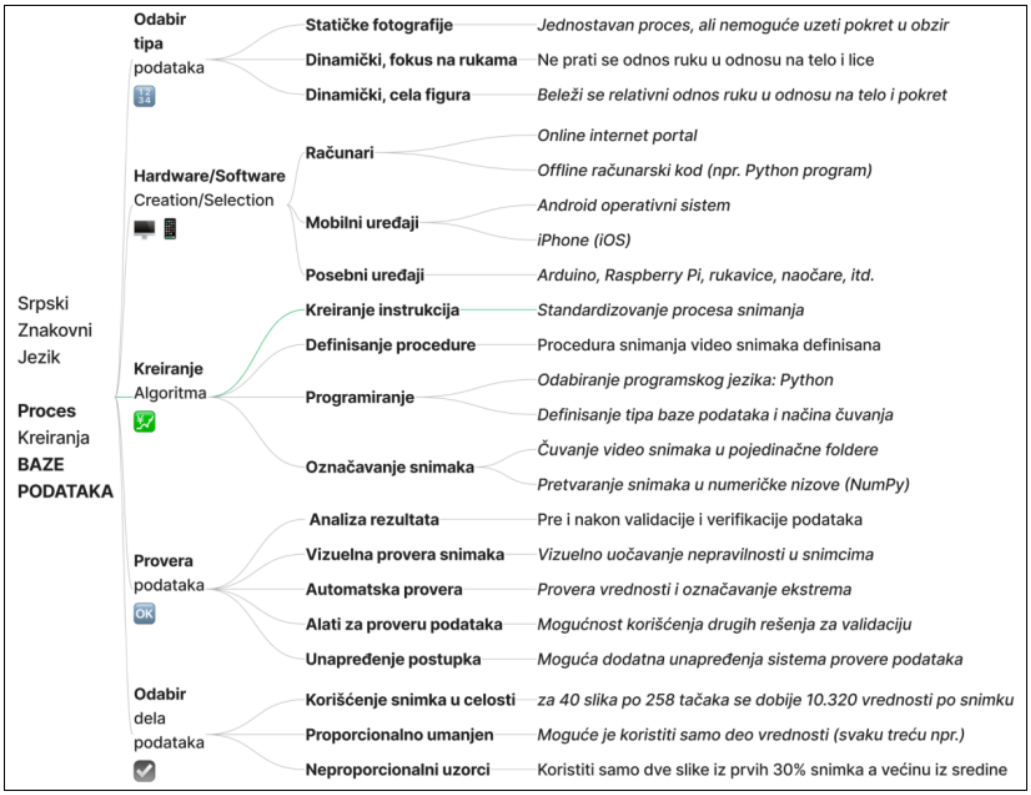
U ovom delu disertacije je objašnjena korišćena metodologija kreiranja baze podataka čije vrednosti opisuju pokrete koji predstavljaju slova srpskog znakovnog jezika. Baza je sastavljena od velikog broja fajlova, koji svaki pojedinačno, predstavlja jedan snimljeni pokret, jednog slova. Svaki fajl u sebi sadrži niz numeričkih vrednosti koji predstavljaju koordinate ključnih tačaka ruku i tela osobe koje su uz pomoć sistema računarskog vida prepoznate za vreme demonstracije slova učesnika u eksperimentu. Za prepoznavanje ključnih tačaka na telu i rukama je korišćena postojeća biblioteka mašinskog učenja *MediaPipe* (Hand Landmarks Guide, 2015), koja je zasnovana na Python programskom jeziku. Ovaj modul je

proizvod *Google*-a, a ukoliko se koristi kao osnova, pruža širok spektar mogućnosti nadgradnje u oblasti računarskog vida (Lugaresi, C., 2019). Obučen je na osnovu velikog broja podataka, rezultati koje generiše su visoke pouzdanosti i tačnosti, a zbog svoje odlične optimizacije, izuzetno je brz. Na slici Slika 1 je prikazan uopšteni proces kreiranja modela mašinskog učenja. Odmah nakon prikupljanja video uzoraka, identifikovane su ključne tačke na telu i rukama, potom pretvorene u numeričke vrednosti i konačno, sačuvane u bazu podataka. Nakon toga, validnost podataka je proverena, a zapisi sa greškama i ekstremnim vrednostima su eliminisani. Baza podataka za srpski znakovni jezik (SZJ) je izgrađena na osnovu kratkih video klipova, sačinjenih od 40 uzastopnih fotografija u rezoluciji od 720p (matrica sačinjena od 720 x 720 tačaka). Rezolucija i broj slika potrebnih za razvoj modela mašinskog učenja mogu biti smanjeni naknadno (po potrebi), ili na drugi način prilagođeni odgovarajućem podskupu podataka. Podaci se finalno dele na dva podskupa, za obuku i testiranje. Tako pripremljeni, mogu se koristiti za obuku odgovarajućeg modela mašinskog učenja. Ako su podaci koji su pruženi modelu mašinskog učenja relevantni, opisni, raznovrsni i sadrže veliki broj jedinstvenih zapisa, model mašinskog učenja će moći da bude obučen bolje i rezultati koje on proizvede će imati veću tačnost i pouzdanost.



Slika 1. Proces kreiranja modela mašinskog učenja (Radaković M., et al., 2024)

Korišćenjem Python programskog jezika (verzija 3.11), razvijen je program za prikupljanje snimaka, prepoznavanje ključnih tačaka i kreiranje baze podataka SZJ. Za te potrebe je korišćen PC računar sa Windows operativnim sistemom. Na slici Slika 2 je prikazan proces konceptualne pripreme koja je stajala iza napisanog programskog koda.



Slika 2. Koncept kreiranja baze podataka srpskog znakovnog jezikaSlika 1. Proces kreiranja modela mašinskog učenja

3.1.1. Izbor strukture i tipa podataka

Na izbor tipa podataka koji će biti korišćeni za kreiranje baze SZJ uticao je pregled postojećih rešenja i uvid u njihove prednosti i mane, kao i specifičnosti pokreta koje je potrebno prepoznati. Najveći broj tehničkih rešenja vezanih za prepoznavanje znakovnih jezika se baziraju na prepoznavanju statičkih slika. Takođe, postoji i veliki broj rešenja koji se fokusiraju na samo jednu ruku. U slučaju dvoručnog SZJ, postoji 5 slova koja se ne mogu prikazati statičnom fotografijom, već ih opisuje ceo pokret rukama. Slova koja se prikazuju pokretom: J (Ј), R (Р), DŽ (Џ), C (Ц) i Ć (Ћ). Da bi se uhvatio ceo gest i omogućilo što preciznije prepoznavanje i razlikovanje ovih slova, odabrano je korišćenje video snimaka za osnovu kreiranja baze podataka. Pored toga, odlučeno je da se u prepoznavanje ključnih tačaka uključi lice i deo tela, tako da se u sistem prepoznavanja unese i komponenta međusobnog položaja ruku u odnosu na lice i telo. Iz tog razloga nije primenjena relativizacija položaja ključnih tačaka ruku, već su zadržane apsolutne vrednosti. Za

razliku od podataka koji bi bili prikupljeni korišćenjem samo fotografija gestova, gde bi bila uzeta u obzir samo slika poslednje pozicije ruku u pokretu, podaci zasnovani na video snimcima čuvaju ceo tok pokreta slova. Korišćenje video snimaka umesto fotografija unosi veoma bitnu kvalitativnu razliku, koja značajno utiče na preciznost i pouzdanost budućih modela mašinskog učenja koji se kreiraju na osnovu tog seta podataka.

Svi video zapisi se čuvaju pomoću XVID video kodera-dekodera unutar univerzalnog AVI kontejnera (Esakki, G., 2021). Python izvorni kod koji sadrži postavke vezane za snimanje datoteka, prikazan je na sliciSlika 3. Kadrovi snimka kamere punog formata od 1280×720 tačaka su smanjeni na 720×720 tačaka, time smanjujući veličinu slike za 43,75%. Pojedinačne slike se tokom snimanja videa prosleđuju modulu MediaPipe za detekciju ključnih tačaka ruku i tela (Raju A., 2022). Prepoznate vrednosti se automatski čuvaju u numeričkoj NumPy datoteci. Za zvaki pojedinačni snimak se kreira poseban fajl. Video zapisi su sačuvani pored numeričkih skupova podataka za buduću kontrolu podataka, uklanjanje grešaka, klasifikaciju i moguće promene. Video datoteke su sačuvane u *.avi formatu, a numeričke vrednosti NumPy su sačuvane u *.npy datotekama sa istim imenima. Imena datoteka su generisana automatski i zasnovana su na računarskom satu.

```
# Kako bi snimljeni klipovi bili manji, izbacuje deo slike sa leve i desne strane
    frame_left_margin = 280
    frame_right_margin = 1000

    # CODEC and Recording details: Dell Latitude 7420 Camera: 1280x720@30fps
    targeted_frame_count = 40  # UKUPAN BROJ FRAME-ova koji se snima
    codec = "XVID"
    video_codec = cv2.VideoWriter_fourcc(*codec)
    # Kreiranje video snimača (za čuvanje video zapisa)

    cap = cv2.VideoCapture(0, cv2.CAP_DSHOW)
    # Postavljanje DirectShow API-ja kao osnovu za snimanje sa kamere
    cap.set(cv2.CAP_PROP_FRAME_WIDTH, video_width)
    cap.set(cv2.CAP_PROP_FRAME_HEIGHT, video_height)
    cap.set(cv2.CAP_PROP_FORMAT, -1)
    # Za kvalitetnije snimke, bez kompresije korišćenje RAW formata
```

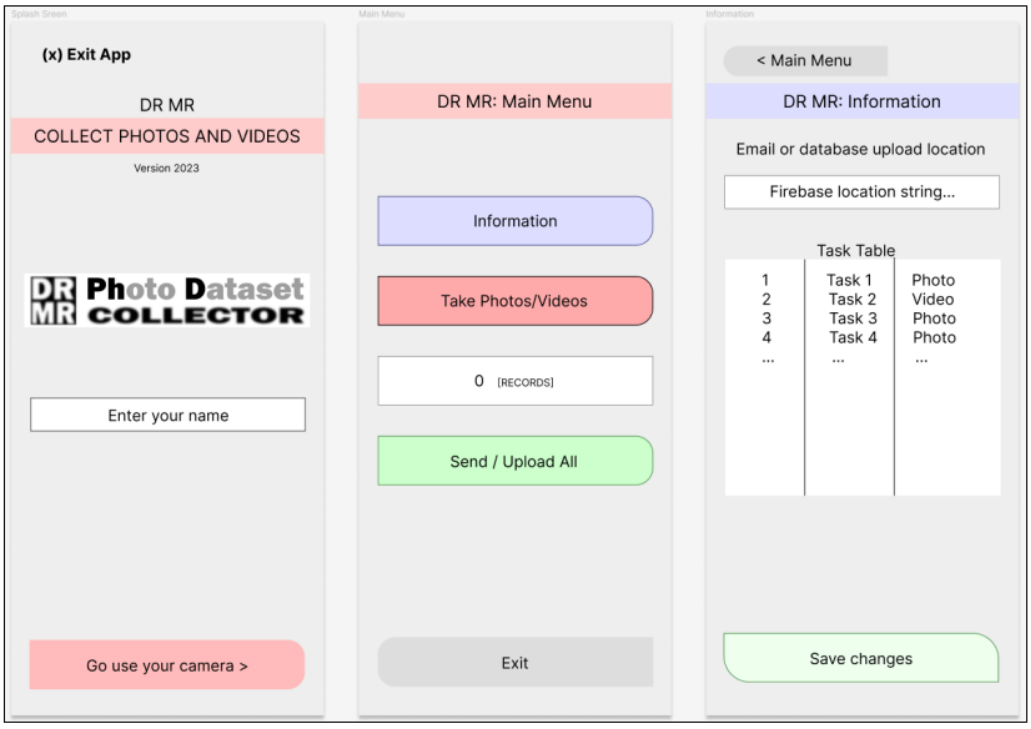
Slika 3. Programski parametri video snimanja i korišćenja kamere

3.1.2. Odabir uređaja za razvoj platforme

Prilikom odlučivanja o hardverskoj platformi koja će se koristiti za kreiranje baze podataka, razmatrani su:

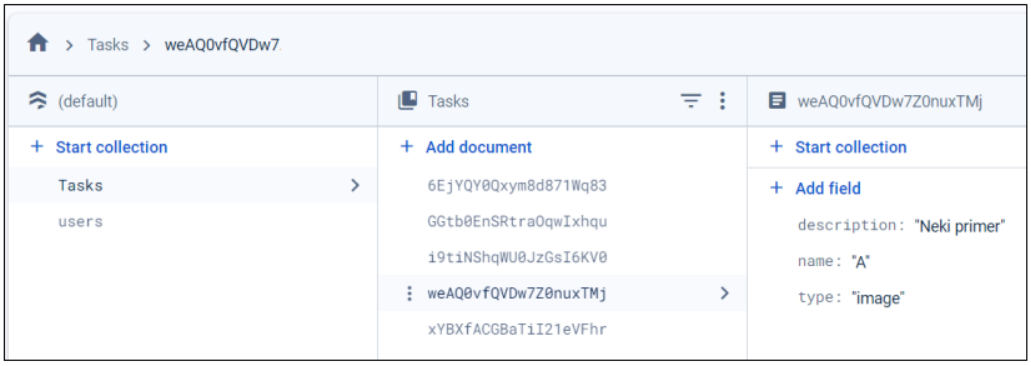
- mobilni uređaji (telefoni i tableti),
- personalni desktop računari i laptopovi,
- Arduino programabilna platforma, Raspberry PI, elektronske rukavice, nosivi senzori, itd. (Jeon et al, 2022).

Da bi se utvrdilo da li ovakvo rešenje na mobilnim uređajima može da zadovolji potrebe istraživanja, razvijena je odgovarajuća Android aplikacija „DR MR“ (Jayatilleke G. et al., 2018). Razvoju aplikacije, prethodilo je njeno projektovanje korišćenjem Figma platforme (Figma, 2024). Figma je besplatan internet portal, čija je glavna namena za kreiranje, deljenje i testiranje dizajna za internet prezentacije, mobilnih aplikacija ili drugih digitalnih proizvoda i rešenja. Nekoliko dizajniranih ekrana za prikaz na mobilnom telefonu je prikazano na slici Slika 4. Korisnik aplikacije unosi svoje ime, a potom u glavnom meniju (u sredini slike Slika 4.), može izabrati prikaz zadataka („Information“), snimiti tražene slike i video klipove („Take Photos/Videos“) ili trajno sačuvati sve na internet lokaciju u oblaku („Send / Upload All“).



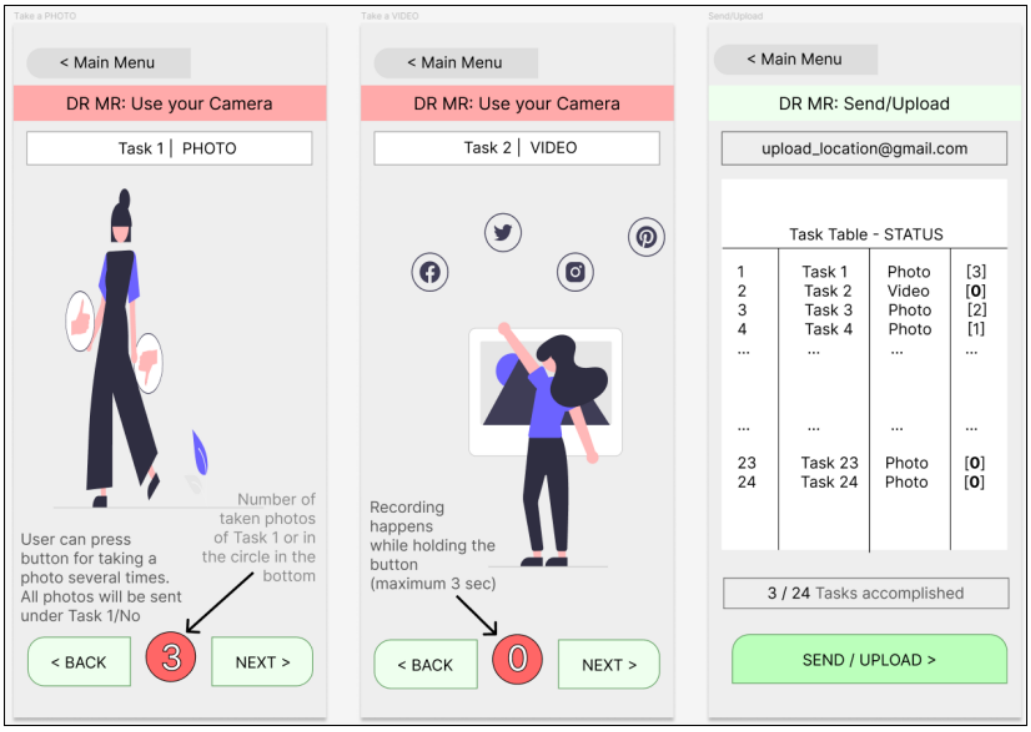
Slika 4. Dizajn Android mobilne aplikacije DR MR

Izborom pregleda zadataka (na desnoj strani slike Slika 4.), korisniku se prikazuje lista koja u slučaju ovog istraživanja sadrži sva slova srpske azbuke. Na ekranu sa informacijama se takođe mogu naći i detalji za povezivanje na internet bazu podataka u oblaku, a može se izmeniti i lista zadataka. Ova lista se može korigovati i u administrativnom delu portala u oblaku, što je prikazano na slici Slika 5. Firebase pregled postavljenih zadataka



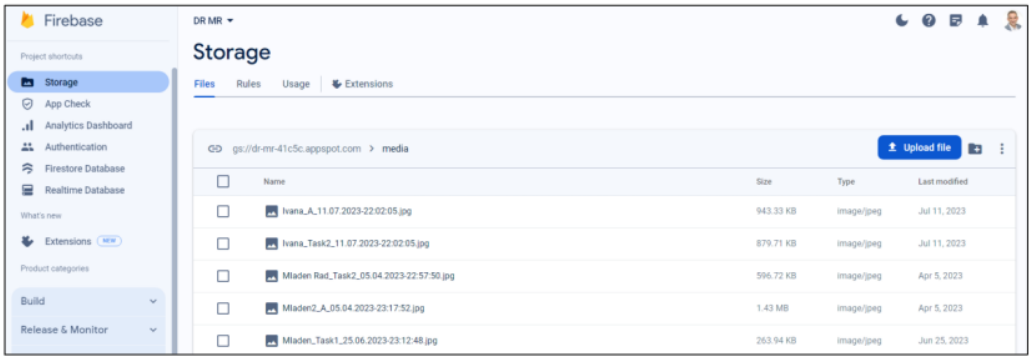
Slika 5. Firebase pregled postavljenih zadataka

Pokretanjem dela programa za snimanje video zapisa ili slikanjem, korisnik ima mogućnost da ispisani zadatak izvrši jednom ili više puta. Broj kreiranih zapisa je označen na centru donjeg dela ekrana, što se može videti na slici Slika 6. Dizajn funkcionalnog dela Android aplikacije DR MR Tip očekivanog zapisa (statična slika ili video zapis) je naznačen pored broja zadatka (Task).



Slika 6. Dizajn funkcionalnog dela Android aplikacije DR MR

Ukupan broj kreiranih zapisa je prikazan u završnoj tabeli (desni deo slike Slika 6. Dizajn funkcionalnog dela Android aplikacije DR MR), nakon čega korisnik može sve podatke poslati u bazu podataka u oblaku. Slanjem ovih zapisa, svi snimci sa telefona se brišu i oslobađa se mesto u memoriji za nove. Primer podataka koji su poslati u bazu u oblaku je prikazan na slici Slika 7. Baza slika i snimaka napravljenih mobilnim telefonom



Slika 7. Baza slika i snimaka napravljenih mobilnim telefonom

Uočene prednosti ovakvog prikupljanja snimaka i fotografija su rasprostranjenost i dostupnost Android mobilnih telefona i uređaja, centralizovano čuvanje svih zapisa u oblaku, olakšano održavanje programa i primena izmena, itd. Mana ovog rešenja je raznolikost mobilnih Android uređaja, što čini težim standardizaciju programa i ponavljanje uslova snimanja. Pored toga, čuvanje velike količine podataka u oblaku donosi dodatne troškove (srazmerne zauzetom prostoru), a potrebno je uzeti u obzir i ograničenja u brzini prenosa podataka.

Kreirana je Android aplikacija „DR MR“ pomoću programa „*Android Studio Electric Eel | 2022.1.1*“. Njenom upotrebom je prikupljeno oko 200 snimaka, za koje je utvrđeno da nisu odgovarajućeg kvaliteta. Snimci su napravljeni na mobilnom telefonu, a zatim su otpremljeni u oblak. Neki od glavnih razloga za napuštanje ovog načina razvoja bili su nekonzistentni zapisi, tehnička ograničenja baze u oblaku, troškovi skladištenja i sporost procesa.

Pošto su specifikacije kamera ugrađenih u laptopove i web kamera za desktop računare relativno konstantne tokom godina (rezolucija 720p), a lični računari su lako dostupni, kreiranje programa za njih je postalo logičan izbor za razvoj programa za kreiranje baze SZJ. Kao rezultat, odabirom PC platforme, olakšana je distribucija i korišćenje programa. Python, objektno-orijentisani programski jezik, izabran je zbog svoje univerzalnosti, prilagodljivosti, resursa i široke podrške programerske zajednice (Python Software Foundation, 2024). On je kompatibilan i može se pokretati na različitim operativnim sistemima, uključujući Windows, Linux i macOS.

Za potrebe istraživanja, proučavana je upotreba različitih uređaja kao što su Arduino ili Raspberry PI (Sapkota B. et al., 2018). Oni imaju određena fizička i još važnije, programska ograničenja. Moguće je kreirati ovakav sistem za prepoznavanje znakovnog jezika i na njima, ali bi to bio mnogo složeniji proces (Radaković M., 2021). To je bio važan faktor zbog kojeg ovi uređaji nisu bili glavni izbor za proces razvoja SZJ programa.

Uzimajući sve dobre i loše strane navedenih mogućih tehničkih rešenja i platformi, odlučeno je da se za razvoj koristi standardni PC računar (laptop) sa Windows operativnim sistemom. Python programski jezik (verzija 3.11) je odabran za programiranje, zbog univerzalnosti i velikog broja dostupnih kvalitetnih i za ovu namenu korisnih modula (Virtanen P., 2018).

3.2. Prikupljanje i snimanje podataka

Proces prikupljanja skupa podataka SZJ je od suštinskog značaja za razvoj modela mašinskog učenja. Veoma je važno da objekti u skupu prikupljenih podataka budu odgovarajućeg i primenjivog kvaliteta i broja. Ako je skup podataka previše mali ili ne odražava tačno podatke koje prikupljamo, rezultujući model mašinskog učenja može biti neefikasan i neprecizan, često proizvodeći pogrešne rezultate. Kada je skup podataka visokog kvaliteta, sa relevantnim podacima, moguće je, na osnovu njih, generisati model mašinskog učenja izuzetne tačnosti. Za zadovoljavajuće i efikasne rezultate, neophodno je kreirati modele mašinskog učenja na velikim i deskriptivnim skupovima podataka.

3.2.1. Struktura učesnika

U postupak snimanja gestova slova srpskog znakovnog jezika je do objavljivanja učestvovala 41 osoba. Svaki akter je snimio po 30 gestova koji predstavljaju slova u srpskom znakovnom jeziku. Učestvovali su studenti, profesori fakulteta i nastavnici specijalnog obrazovanja u uzrastu od 19 do 50 godina. Svi su prethodno bili informisani o detaljima i ciljevima istraživanja, a svi su potpisali saglasnost pre početka snimanja. Ovo istraživanje je sprovedeno u skladu sa *Deklaracijom iz Helsinkija* (World Medical Association, 2013), a protokol je odobrila Etička komisija Fakulteta savremenih umetnosti u Beogradu (Projektni kod: SSL23). Takođe, svim učesnicima je prikazano i prethodno pripremljeno video uputstvo, u kome je detaljno objašnjen proces i pravilna procedura snimanja. Učesnicima je naglašeno da koriste standardne gestove SZJ prilikom snimanja, kako bi skup podataka u bazi bio konzistentan i što bolje opisivao srpsku azbuku.

Kad god se koristi veštačka inteligencija, zbog svoje prirode, razmene i prikupljanja informacija, kao i interakcije sa ljudima, postavljaju se pitanja o bezbednosti podataka i etici. Pitanja koja se tiču privatnosti podataka i saglasnosti korisnika su validna i treba ih postaviti svaki put kada aktivnosti uključuju snimanje i analizu osetljivih ličnih informacija koje se prezentuju gestovima. Rezultat generisanog, prikazanog i objavljenog skupa podataka SZJ su numeričke vrednosti ključnih tačaka, bez mogućnosti identifikacije pojedinaca ili povratka na originalnu sliku. Skup podataka SZJ, kao i sam proces njegovog kreiranja pratili su i zadovoljili etičke i standarde bezbednosti podataka. Stvarni video zapisi nisu predmet deljenja,

objavljivanja ili javnog korišćenja. Umesto toga, koriste se samo privremeno, za generisanje numeričkih vrednosti koje predstavljaju ključne tačke pokreta ruku i tela. Te numeričke vrednosti se koriste kao osnova skupa podataka i ne mogu biti povezane sa video zapisima ili za prepoznavanje učesnika. One su najveća i stvarna vrednost objavljenog skupa podataka, jer se mogu koristiti kao osnova za kreiranje brojnih različitih modela mašinskog učenja. Video zapisi su privremeno čuvani lokalno na računaru i korišćeni samo za prepoznavanje ključnih tačaka na rukama i telu. Takođe, video zapisi su korišćeni za inicijalnu vizuelnu kontrolu snimljenih gestova u procesu čišćenja, kategorizacije i provere podataka.

3.2.2. Priprema okruženja za snimanje

Za potrebe doktorske disertacije, snimanje gestova je trebalo da obuhvati subjekte različitih veličina, pozicija i proporcija kako bi se proizveo upotrebljiv i značajan skup podataka. Kamera se pozicionira na različite udaljenosti od učesnika, pa se siluete i pozicije ruku i tela razlikuju (kako je prikazano na Slici Slika 8. Pozicija, veličina i udaljenost demonstratora od kamere). U ovom istraživanju, od subjekata je očekivano da prikažu slova znakovnog jezika različitim brzinama i stilovima, postavljajući svoje ruke u kadar na različite načine. Veći broj promenljivih vrednosti prilikom snimanja gestova doprinosi generisanju kvalitetnije baze podataka koja se može koristiti za dalju obuku efikasnijih modela mašinskog učenja.

Šarena i raznobojna pozadina i/ili odeća učesnika, mogu imati izuzetno negativan uticaj na rezultate prepoznavanja ključnih tačaka ruku i tela. Utvrđeno je da jarko obojene pozadine i šarena odeća na snimljenim subjektima štete korišćenom modelu prepoznavanja MediaPipe. Ako je pozadina i/ili odeća bila veoma šarena, model je pružao manje precizne rezultate. Nasuprot tome, ako je pozadina i/ili odeća bila samo jedne boje, okvir MediaPipe je mnogo preciznije detektovao ključne tačke.



Slika 8. Pozicija, veličina i udaljenost demonstratora od kamere

3.2.3. Procedura snimanja

Pre nego snimanje počne, učesnik treba da bude pozicioniran u optimalnoj poziciji ispred kamere (kao na slici Slika 8. Pozicija, veličina i udaljenost demonstratora od kamere), bez prikazivanja ruku. Čim ruke uđu u scenu (vidno polje snimanja kamere), one budu prepoznate od strane programa i snimanje automatski počinje. Snimanje podrazumeva čuvanje 40 uzastopnih pojedinačnih fotografija i odgovarajućih numeričkih vrednosti koje predstavljaju prepoznate ključne tačke ruku i tela. Za vreme snimanja, subjektu se prikazuje crvena traka u levom delu slike, koja prikazuje protok vremena i broj snimljenih kadrova. Prateći traku, demonstrator može uskladiti brzinu svog pokreta sa programom i rasporediti ga ravnomerno od početka do kraja. Ceo pokret uključuje ulazak ruku u scenu, prikazivanje samog znaka gesta i izlazak ruku iz snimljenog kadra. Ukoliko ruke ostanu u sceni duže od 40 kadrova, aplikacija ne započinje snimanje sledećeg filmskog zapisa sve dok se ruke ne uklone iz scene. Aplikacija snima tačno 40 uzastopnih kadrova. S obzirom na to da postoji mogućnost da neke ključne tačke na rukama ne budu prepoznate u svim uzastopnim kadrovima, postavljen je sistem zaštite koji bi prepoznao da su ruke sklonjene tek ukoliko su van scene tokom 5 uzastopnih kadrova. Sledeće snimanje može početi tek nakon što obe ruke napuste scenu.

Slika Slika 9. Parametri MediaPipe modela i prikazuje deo izvornog koda koji ilustruje određene opcije i osnovne postavke MediaPipe biblioteke. Parametar "holistic" aktivira deo modula koji na osnovu prosleđene fotografije, prepoznaje sve pojedinačne ključne tačke za celokupan položaj tela, lica i obe ruke.

```
# Postavljanje parametara za MediaPipe modul
    with mp_holistic_model.Holistic(min_detection_confidence=0.5,
                                    min_tracking_confidence=0.5) as holistic:
        while cap.isOpened():
            # Ulazni podaci sa kamere se očitavaju u promenljivu frame
            ret, frame = cap.read()
            # Smanjivanje slike na 3/5 ukupne širine: 1280x720px > 720:720px
            frame = frame[0:video_height, frame_left_margin:frame_right_margin]
            # Prepoznavanje ključnih tačaka iz slike koja je u promenljivoj frame
            results = mediapipe_keypoint_locations(frame, holistic)
```

Slika 9. Parametri MediaPipe modela i okvir snimanja

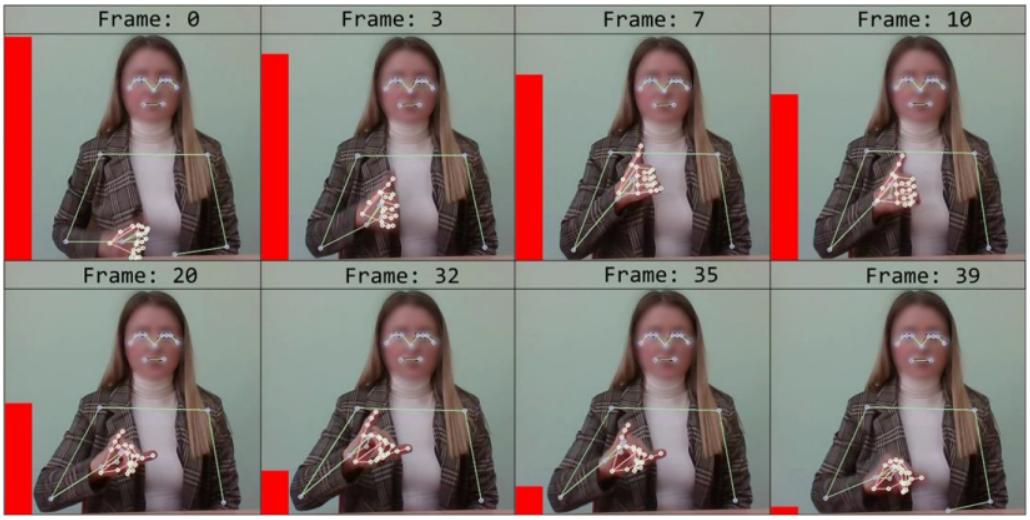
Kao rezultat procesa snimanja, aplikacija čuva dva fajla: kratak video zapis (oko 4 sekunde trajanja) i odgovarajući NumPy niz sa numeričkim vrednostima prepoznatih ključnih tačaka ruku i tela (tačno 10.320 vrednosti po jednom fajlu).

NumPy nizovi se koriste kao osnova za razvoj modela mašinskog učenja, dok se sačuvani video zapisi koriste za vizuelnu analizu i proveru snimaka. Kako snimci mogu biti generisani greškom ili klasifikovani u pogrešnu klasu, datoteke koji sadrže vizuelne informacije su važne za kontrolu i obradu podataka. Slike Slika 10. Kadrovi koji prikazuju gest SZJ slova „J“ i Slika 11. Prepoznate ključne tačke na rukama i telu gestom prikazanog slova „J“ prikazuju osnovnu razliku između dva sačuvana fajla (standardne slike i prepoznate ključne tačke).



Slika 10. Kadrovi koji prikazuju gest SZJ slova „J“ (Radaković M., et al., 2024)

Na slici Slika 10. Kadrovi koji prikazuju gest SZJ slova „J“ su prikazane samo neke statičke slike (od ukupno 40) iz snimljenog video zapisa slova "J". Zbog dinamičke prirode gesta koji se koristi za pokazivanje slova „J“, vidi se da samo jedna statička fotografija ne bi bila dovoljna za prepoznavanje značenja ovog gesta. Korišćenje celokupnih video zapisa za obradu podataka omogućava detekciju kompleksnijih pokreta i gestova koji u svojoj osnovi imaju i pokret. Kao što je prethodno navedeno, slova koja u svojoj osnovi imaju pokret su: J (J), R (Р), DŽ (Џ), C (Ц) i Ć (Ћ). Slika Slika 11. Prepoznate ključne tačke na rukama i telu gestom prikazanog slova „J“ prikazuje prepoznate ključne tačke na rukama i telu demonstratora, koje odgovaraju fotografijama na slici Slika 10. Kadrovi koji prikazuju gest SZJ slova „J“. Vrednosti su prepoznate korišćenjem modula MediaPipe, a čuvaju se u fajlu NumPy tipa, sa drugim tipom, ali istim imenom kao i čiste fotografije.

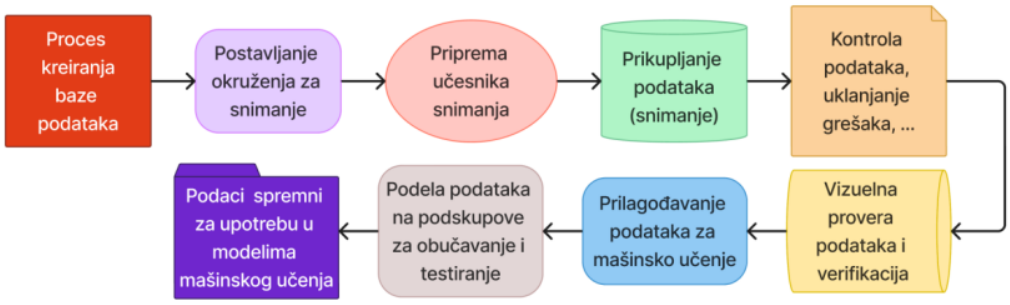


Slika 11. Prepoznate ključne tačke na rukama i telu gestom prikazanog slova „J“

3.2.4. Snimanje gestova 30 slova Azbuke

Cilj kreiranja skupa podataka je da se obezbedi dovoljan broj različitih snimaka svakog od 30 srpskih znakova Azbuke. Skup podataka i izvorni Python kod su javno dostupni, što omogućava nadogradnju postojećeg skupa podataka, razvoj novih modula i aplikacija, kao i nastavak teorijskih i praktičnih istraživanja.

Ceo proces kreiranja skupa podataka prikazan je na slici Slika 12. Pre početka procesa snimanja, scena snimanja se priprema prema unapred određenim preporukama. Osoba koja snima gestove treba da sedi naspram računara, držeći gornji deo tela u kadru, sa rukama van scene. Za učesnike u snimanju, pripremljeno je video uputstvo, sa detaljima procedure snimanja i precizno određenim ciljevima. Dobrom pripremom se osigurava kreiranje kvalitetnih video snimaka, sa malim brojem grešaka. Nakon prikupljanja podataka, prelazi se u fazu evaluacije i čišćenja podataka. Kontrola zabeleženih snimaka se vrši prvo vizuelnom proverom, a potom i pomoću automatizovane numeričke analize. Veliki broj snimljenih fotografija po slovu omogućava smanjivanje dobijenih uzoraka ili njihovu naknadnu obradu radi poboljšanja performansi budućih sistema i modela mašinskog učenja. Nakon potvrde validnosti podataka, oni se razdvajaju u podskupove za testiranje i obuku. Na kraju, odabrani skup podataka se prosleđuje modelu mašinskog učenja.



Slika 12. Algoritam kreiranja baze podataka (Radaković M., et al., 2024)

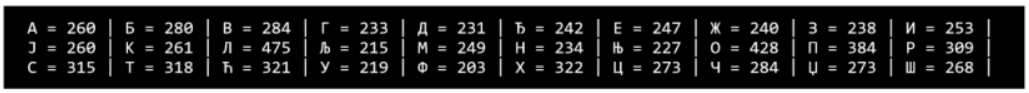
Tabela Tabela 3 sadrži 30 slova azbuke za koja se video zapisi kreiraju. Zbog lakše obrade podataka u Python-u, umesto ćiriličnih slova, programu se prosleđuju odgovarajuće latinične oznake (kao u tabeli Tabela 3). Originalna slova azbuke SZJ su prikazana u donjem redu tabele. Srednji red tabele prikazuje čitljivu i računarski prihvatljivu alternativnu formu originalnih srpskih ćiriličnih slova. Karakteri u srednjem redu korišćeni su u Python programu za SZJ u svrhu organizacije strukture datoteka i foldera skupa podataka.

Tabela 3. Snimanje 30 slova srpske Azbuke

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
A	B	V	G	D	DJ	E	ZH	Z	I	J	K	L	LJ	M	M	NJ	O	P	R	S	T	CC	U	F	H	C	CH	DZ	SH
А	Б	В	Г	Д	Ђ	Е	Ж	З	И	Ј	К	Л	Љ	М	Н	Њ	О	П	Р	С	Т	Ћ	У	Ф	Х	Ц	Ч	Џ	Ш

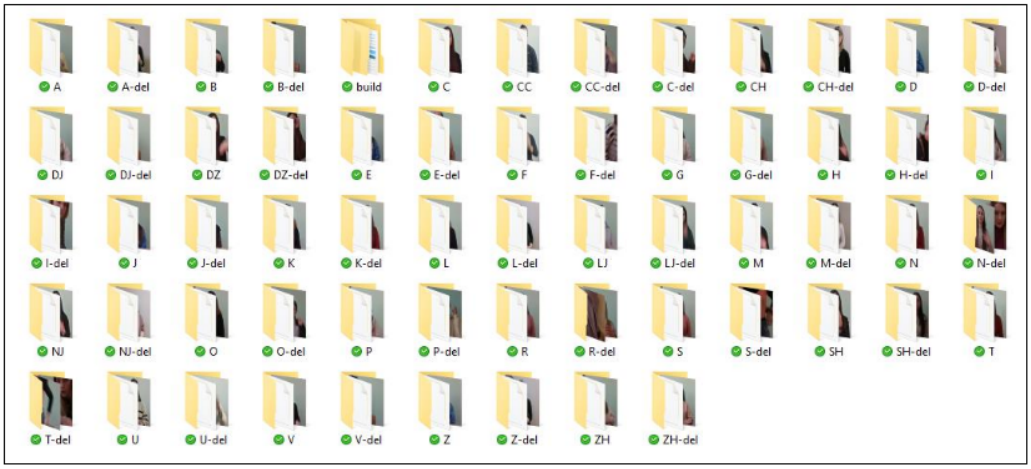
3.3. Rezultati snimanja

Konačni skup podataka srpskog znakovnog jezika uključuje 8.346 video zapisa i prateće datoteke koje sadrže numeričke vrednosti prepoznatih ključnih tačaka tela i ruku. Kao što se vidi na slici Slika 13, svako slovo je prosečno snimljeno 278 puta. Kreirana baza podataka ima približnu veličinu od 4,15 GB i sadrži ukupno 16.740 datoteka. Sa prosečnim trajanjem od 4 sekunde za svaki video zapis, ukupno vreme provedeno samo tokom kreiranja skupa podataka, bez priprema ili pauza, iznosi približno 33.384 sekundi ili 9,3 sata.



Slika 13. Broj snimljenih video zapisa SZJ

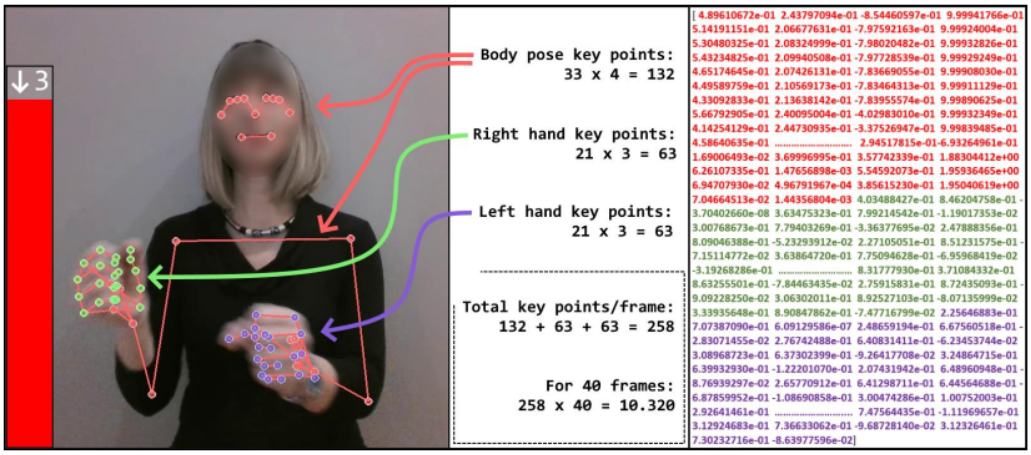
Slika 14 prikazuje strukturu foldera za svih 30 slova Azbuke koje uključuju i ispravne i neispravne zapise. Svi kreirani neprihvatljivi snimci su premešteni u odgovarajuće foldere za obrisane datoteke radi naknadne evaluacije i statističke obrade. Za dalja unapređenja i proširenja baze podataka, sve datoteke i folderi baze SZJ, kao i odgovarajući Python izvorni kod, koji može biti korišćen za kreiranje novih snimaka, su postavljeni na internet i javno su dostupni.



Slika 14. Struktura fajlova i foldera

3.3.1. Snimljeni video zapisi

Crvena traka u levom delu slike Slika 15, prikazuje tačan broj snimljenih slika od početka snimanja video snimka, što odgovara proteklom vremenu. U trenutku kada snimanje počne, crvena traka je potpuno popunjena, što ukazuje da je preostalo svih 40 slika za snimanje. Crvena traka se smanjuje kako snimanje napreduje, a u prikazanom primeru prikazuje status nakon snimljene 3 slike. Aktivni prikaz protoka vremena pomaže učesniku da proceni trajanje snimanja gesta, pa mu time pomaže da ga ravnomerno raspodeli u snimku i održi standardan ritam snimanja.



Slika 15. Prikaz gesta slova „Д“ (D) sa vrednostima ključnih tačaka na slici broj 3 od 40

Na desnoj strani prikazane slike, prikazani su neke od stvarnih numeričkih vrednosti koje čine osnovu skupa podataka. Kao što je prikazano na slici Slika 15, za jednu snimljenu sliku beleži se 258 numeričkih vrednosti. Crvene, zelene i ljubičaste linije sa centra slike pokazuju na ključne tačke tela i ruku gde su ove numeričke vrednosti kreirane. Kao rezultat procesa prepoznavanja ključnih tačaka, program generiše matrice koje se naknadno pretvaraju u jednodimenzionalni numerički niz od ukupno 10.320 vrednosti za svaki pojedinačni video zapis.

Ukoliko MediaPipe okvir ne prepozna neku ključnu tačku na ruci, licu ili telu, za nju se upisuje vrednost 0 u rezultujući niz. U slučaju video zapisa sa slike Slika 15, na kojoj je prikazana samo 3. slika, sve ključne tačke su prepoznate, pa se tako za nju ne pojavljuje nijedna vrednost nula. Neprepoznate ključne tačke su najčešće prisutne u prvoj i poslednjoj slici, kada ruke ulaze i izlaze iz scene snimanja.

Primer video zapisa sa nultim vrednostima ključnih tačaka u nekim od 40 slika, prikazan je na slici Slika 16. Dijagram prikazuje da je od početka snimanja subjekt imao jednu ruku u sceni (63 ključne tačke označavaju ključne tačke za jednu ruku), a druga ruka je ušla u scenu u trećoj slici. Počevši od slike 3, i leva i desna ruka, kao i sve ključne tačke položaja tela i lica, bile su prepoznate i zabeležene (nije bilo nijedne nulte vrednosti). Sve ključne tačke su prepoznate i zabeležene od slike 3 do slike 33. Jedna ruka izlazi iz scene snimanja u slici 34 (63 nultih vrednosti), dok je druga napušta u slici 38 (126 nultih vrednosti).



$$\text{Handedness} = \begin{bmatrix} \text{index} \\ \text{score} \\ \text{hand} \end{bmatrix} \quad (2)$$

gde je:

Handedness – matrica sa vrednostima koje pokazuju koja je ruka prepoznata

index – parametar koji označava redni broj jedne od više prepoznatih ruku

score – je verovatnoća tačnosti predviđanja da li je prepoznata leva ili desna ruka

hand – sadrži detaljne informacije o prepoznatoj ruci

$$\text{Landmarks} = \begin{bmatrix} x1 & x2 & x3 & \dots & x21 \\ y1 & y2 & y3 & \dots & y21 \\ z1 & z2 & z3 & \dots & z21 \end{bmatrix} \quad (3)$$

gde je:

Landmarks – matrica sa koordinatnim vrednostima ključnih tačaka

x1, *y1*, *z1* – koordinatne vrednosti jedne ključne tačke

$$\text{WorldLandmarks} = \begin{bmatrix} ax1 & ax2 & ax3 & \dots & ax21 \\ ay1 & ay2 & ay3 & \dots & ay21 \\ az1 & az2 & az3 & \dots & az21 \end{bmatrix} \quad (4)$$

gde je:

WorldLandmarks – matrica sa apsolutnim koordinatnim vrednostima tačaka

ax1, *ay1*, *az1* – apsolutne koordinatne vrednosti jedne ključne tačke

$$\text{HandKeyPoint1} = \begin{bmatrix} x1 \\ y1 \\ z1 \end{bmatrix} \qquad \text{ApsHandKeyPoint1} = \begin{bmatrix} ax1 \\ ay1 \\ az1 \end{bmatrix} \quad (5)$$

gde je:

HandKeyPoint1 - matrica sa apsolutnim koordinatnim vrednostima 1 tačke

x1, *y1*, *z1* – koordinatne vrednosti jedne ključne tačke

ApsHandKeyPoint1 - matrica sa apsolutnim koordinatnim vrednostima 1 tačke

ax1, *ay1*, *az1* – apsolutne koordinatne vrednosti jedne ključne tačke

Matrice koje prikazuju prepoznate relativne koordinatne vrednosti ključnih tačaka leve i desne ruke se mogu prikazati kao u jednačini (6).

$$rh = \begin{bmatrix} rhx1 & rhx2 & rhx3 & \dots & rhx21 \\ rhy1 & rhy2 & rhy3 & \dots & rhy21 \\ rhz1 & rhz2 & rhz3 & \dots & rhz21 \end{bmatrix} \quad lh = \begin{bmatrix} lhx1 & lhx2 & lhx3 & \dots & lhx21 \\ lhy1 & lhy2 & lhy3 & \dots & lhy21 \\ lhz1 & lhz2 & lhz3 & \dots & lhz21 \end{bmatrix} \tag{6}$$

gde je:

rh – matrica sa koordinatnim vrednostima tačaka za desnu ruku

lh – matrica sa koordinatnim vrednostima tačaka za levu ruku

Za razliku od vrednosti koje opisuju položaj ključnih tačaka leve i desne ruke, matrica koja sadrži informacije o ključnim tačkama lica i tela pored tri koordinate, sadrži i četvrtu veličinu. Ta dodatna promenljiva nosi informaciju o vidljivosti svake pojedinačne tačke. Matrica koja sadrži sve rezultate prepoznavanja ključnih tačaka na telu i licu je prikazana u jednačini (7).

$$\text{PoseLandmarkerResult} = [[Landmarks] \quad [WorldLandmarks]] \tag{7}$$

gde je:

PoseLandmarkerResult – matrica sa rezultatima prepoznavanja tela i lica sa slike

Landmarks – matrica sa koordinatnim vrednostima ključnih tačaka

WorldLandmarks – matrica sa apsolutnim koordinatnim vrednostima tačaka

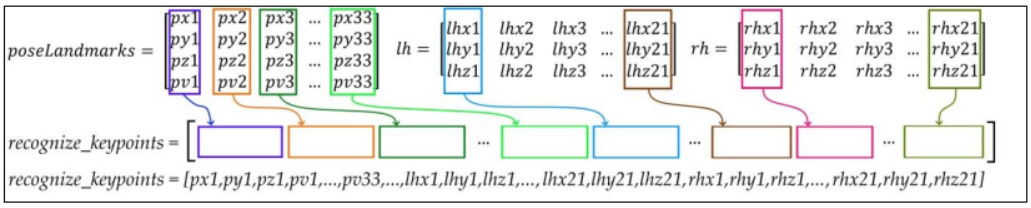
$$Landmarks = \begin{bmatrix} px1 & px2 & px3 & px4 & \dots & px32 & px33 \\ py1 & py2 & py3 & py4 & \dots & py32 & py33 \\ pz1 & pz2 & pz3 & pz4 & \dots & pz32 & pz33 \\ pv1 & pv2 & pv3 & pv4 & \dots & pv32 & pv33 \end{bmatrix} \tag{8}$$

gde je:

px_i, py_i, pz_i – koordinatne vrednosti jedne ključne tačke sa lica ili tela

pv_i – veličina koja pokazuje u kojoj meri je prepoznata tačka vidljiva

Za potrebe ovog istraživanja, trodimenzionalne vrednosti prepoznatih tačaka se transformišu u jednodimenzionalni niz. Na slici Slika 17 je prikazana metodologija transformacije podataka od značaja iz dobijenih matrica u jednodimenzionalni niz (Radaković M., et al., 2024).



Slika 17. Prikaz transformacije podataka iz matrica u jednodimenzionalni niz

Rezultujući jednodimenzionalni niz sadrži 258 veličina koje opisuju poziciju i vidljivost ključnih tačaka na rukama, licu i telu, a odnosi se na samo jednu sliku iz snimljenog video zapisa. Proračun broja numeričkih vrednosti po slici je prikazan u jednačini (9).

$$broj\_vrednosti\_po\_slici = pose + lh + rh = 33 \times 4 + 21 \times 3 + 21 \times 3 = 258 \quad (9)$$

gde je:

broj_vrednosti_po_slici – ukupan broj rezultujućih numeričkih vrednosti

pose – broj vrednosti ključnih tačaka na telu i licu

lh – broj vrednosti ključnih tačaka na levoj ruci

rh – broj vrednosti ključnih tačaka na desnoj ruci

Izvorni Python kod, koji transformiše matrice od interesa za istraživanje u jednodimenzionalni niz je predstavljen na slici Slika 18. Korišćena je ugrađena funkcija *flatten()*, koja na osnovu ulaznih višedimenzionalnih podataka, vraća jednodimenzionalni niz (kao na slici Slika 17).

```
def recognize_keypoints(results):
    right_hand_keypoints = np.array([[result.x, result.y, result.z] for result in
results.right_hand_landmarks.landmark]).flatten() \
        if results.right_hand_landmarks else np.zeros(21*3)
        # 21 - broj kljucnih tacaka
    left_hand_keypoints = np.array([[result.x, result.y, result.z] for result in
results.left_hand_landmarks.landmark]).flatten() \
        if results.left_hand_landmarks else np.zeros(21*3)
        # 21 - broj kljucnih tacaka
```

```
    body_keypoints = np.array([[result.x, result.y, result.z, result.visibility]
for result in results.pose_landmarks.landmark]).flatten() \
        if results.pose_landmarks else np.zeros(33*4)
        # 33 - broj kljucnih tacaka
    return np.concatenate([body_keypoints, left_hand_keypoints,
right_hand_keypoints])
        # Ukupan broj tacaka po FRAME-u:
        # 33x4 + 21x3 + 21x3 = 258 / za 40 frame-ova = 40 x 258 = 10.320 tacaka
```

Slika 18. Funkcija za konverziju vrednosti ključnih tačaka u jednodimenzionalni niz

Kao što je u metodologiji i proceduri snimanja baze pomenuto, svaki snimak pojedinačnog slova Azbuke sadrži 40 slika. To znači da je za čuvanje informacija o kretanju ključnih tačaka na samo jednom snimku, jednog slova, neophodno pomnožiti 40 slika sa 258 numeričkih vrednosti. Jedan zapis, na osnovu toga, sadrži tačno 10.320 brojeva. Ovako velika količina informacija o kretanju ključnih tačaka na snimku(109) svakog slova, pri svakom pokretu, predstavlja dobru osnovu za dalju obradu i potencijalnu optimizaciju ili izmene.

3.3.3. Zapisivanje ključnih tačaka sa video snimaka u bazu podataka

Video zapisi dobijeni snimanjem demonstratora pomoću kamere ugrađene u računar se koriste samo kao osnova za dobijanje fajlova koji sadrže koordinatne vrednosti ključnih tačaka na rukama, licu i telu. Rezultat prepoznavanja jedne od slika u video zapisu je prikazan na slici Slika 15, dok je računarski kod koji može naknadno da učita i koristi fajl sa vrednostima prikazan na slici Slika 19.

```
import numpy as np
import os
import time

def konvertuj_npy_fajl_u_csv(file_name):
    numpy_niz = np.load(file_name)
    ELEMENATA_U_JEDNOM_FREJMU = 33*4 + 21*3 + 21*3
    broj_frejma = 0
    svi_frejmovi = []
    for j in range(int(len(numpy_niz)/(ELEMENATA_U_JEDNOM_FREJMU))):
        broj_frejma += 1
        jedan_red_podataka = []
        for i in range(ELEMENATA_U_JEDNOM_FREJMU):
            jedan_red_podataka.append(numpy_niz[i+j*ELEMENATA_U_JEDNOM_FREJMU])
        svi_frejmovi.append(jedan_red_podataka)

    print(svi_frejmovi)
    # SAČUVAJ SVE PODATKE U .csv FAJL
    try:
        np.savetxt(f"{file_name}.csv", svi_frejmovi, delimiter=",")
        print(f"Successfully saved {file_name}.csv file.")
    except:
```

```
        print("Snimanje nije uspelo!")
    return


if __name__ == "__main__":
    naziv_fajla_za_konverziju = "..\\SSL Program\\D\\17039342851960452-XVID.npy"
    konvertuj_npy_fajl_u_csv(naziv_fajla_za_konverziju)
```

Slika 19. Python izvorni kod za čitanje snimljenog NumPy zapisa iz baze podataka

U zavisnosti od namene, konačni skup podataka se može prilagoditi potrebama. Broj od 40 slika u svakom od video zapisa može biti umanjen, rezolucija od 720 × 720 tačaka po slici takođe može biti smanjena, a sve sa ciljem brže obrade podataka i zbog manje trošenja vremena u modelima mašinskog učenja koju će ovu bazu koristiti za obučavanje. Pomenute vrednosti su za kreiranje inicijalnih video zapisa korišćene sa namenom da mogu zadovoljiti i potencijalno zahtevnije modele (Siebert J. Et al., 2021).

Nakon kreiranja finalnog skupa podataka sa velikim brojem datoteka sa podacima, podela skupa na podskupove za obučavanje i testiranje je važan korak u procesu kreiranja kvalitetnog modela mašinskog učenja. Podskupovi se kreiraju za obuku, testiranje i proveru. Strategija podela podataka koja će se koristiti određuje se na osnovu karakteristika skupa podataka, problema koji se rešava i ciljeva planiranog projekta mašinskog učenja. Bitno je pažljivo odabrati strategiju podele kako bi se osigurala buduća pouzdanost i preciznost modela, kao i performanse nad nepoznatim ulaznim podacima.

3.3.4. Kontrola i provera podataka

Kontrola i provera skupa generisanih vrednosti predstavlja ključni korak u procesu pripreme podataka za bilo koji projekat mašinskog učenja. Ovo osigurava tačnost, pouzdanost i prikladnost podataka korišćenih za obuku i testiranje modela za željeni zadatak. Tehnika generisanja baze podataka SZJ nije dozvoljavala stvaranje duplih zapisa zbog svog načina beleženja jedinstvenih video isečaka. U proizvedenom skupu podataka takođe nije bilo praznih podataka ili informacija koje je bilo potrebno dodatno popuniti.

Razlozi za odbacivanje i uklanjanje video zapisa iz finalnog skupa podataka mogu biti:

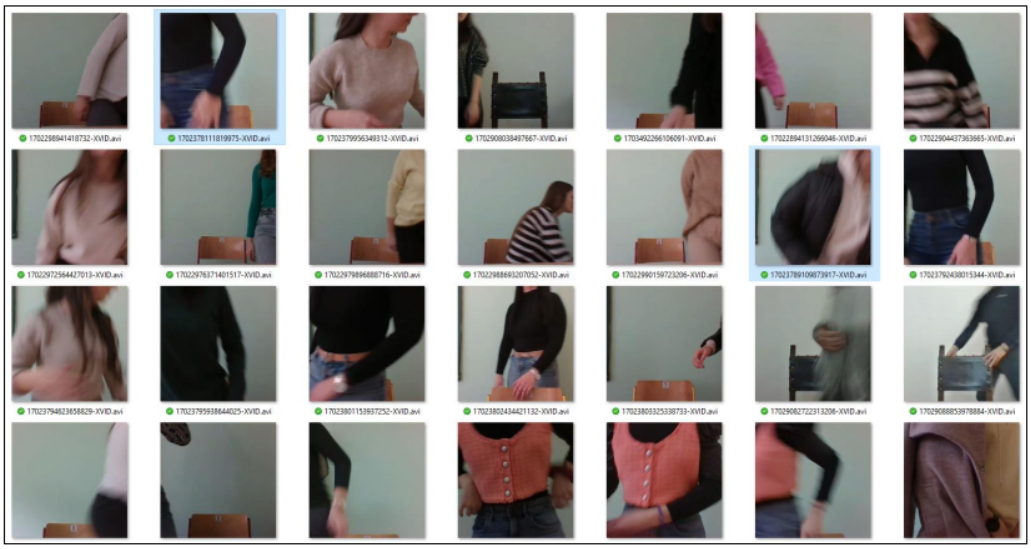
- Zapisi su napravljeni nenamerno, ukoliko su ruke ušle u kadar;

- Snimci su zabeleženi dok se osoba pripremala za snimanje;
- Zapisi su unapred klasifikovani kao neodgovarajuće slovo (snimanje pogrešnog slova);
- Predstavljeni znak gesta koji odgovara SZJ standardu (pogrešna ruka je korišćena ili gest);
- Brzina kretanja gesta nije odgovarala video isečku od 40 slika (prebrzo ili presporo izveden pokret).

Metodi provere, čišćenja i klasifikacije skupa podataka korišćeni u procesu kreiranja SSL baze podataka bili su sledeći:

- Vizuelna provera i klasifikacija podataka na osnovu video isečaka;
- Numerička provera i klasifikacija na osnovu podataka generisanih namenski generisanim Python kodom.

Vizuelna provera video zapisa za skup podataka je korišćena kao prvi korak u procesu provere ispravnosti podataka. Ovo uključuje pregledanje filmova kako bi se osiguralo da odgovaraju željenim standardima i kriterijumima. Svi neadekvatno snimljeni video isečci su uklonjeni i smešteni u posebne fascikle za obrisana slova. Na slici Slika 20 je prikazana kolekcija nepravilno snimljenih video isečaka.



Slika 20. Neispravno snimljeni video zapisi

Postoje različiti gotovi računarski programi i moduli koji su specijalizovani za automatizovanje kontrole i čišćenja podataka. Alati koji se u te svrhe koriste zavise

od tipa podataka i specifičnosti zadatka. Pošto je programski jezik Python odabran kao glavna programska platforma za ovaj projekat, u okviru njega je kreiran modul koji pomaže u prepoznavanju neadekvatnih snimaka. Na slici Slika 21 je prikazan deo računarskog koda koji prebrojava broj neprepoznatih ključnih tačaka na svakom snimku i predstavlja rezultate u preglednoj tabeli. Na taj način, program ukazuje na snimke koji sadrže ekstremne vrednosti i predlaže njihovo uklanjanje iz baze.

```
def prebroj_nulte_vrednosti(naziv_fajla):
    numpy_niz = np.load(naziv_fajla)
    ukupno_vrednosti = len(numpy_niz)
    ELEMENATA_U_JEDNOM_FREJMU = 33*4 + 21*3 + 21*3
    brojac = 0
    broj_frejma = 0
    svi_frejmovi = []
    ukupan_broj_nula = 0
    for j in range(int(len(numpy_niz)/(ELEMENATA_U_JEDNOM_FREJMU))):
        frejm_sa_nulom = 0
        broj_frejma += 1
        data_row = []
        for i in range(ELEMENATA_U_JEDNOM_FREJMU):
            data_row.append(numpy_niz)
            if numpy_niz[brojac] == 0:
                frejm_sa_nulom += 1
                ukupan_broj_nula += 1
            brojac += 1
        svi_frejmovi.append(data_row)
        #time.sleep(0.5)
        #print("Frame number: ", frame_number, "Frame Zeros: ", frame_zeros)

    np.savetxt("jedan_klip.csv", numpy_niz, delimiter=",")
    return(ukupno_vrednosti, ukupan_broj_nula)

if __name__ == "__main__":
    prebroj_snimke_po_slovima()
    prebroj_obrisane_snimke_po_slovima()
    fajl_za_snimanje = input("Enter name for the destination .csv file: ")
    fajl_za_snimanje += ".csv"
    file = open(fajl_za_snimanje, "w")
    write = csv.writer(file, delimiter=",")

    for key, value in sign_dict.items():
        zadato_slovo = key
        lista = list_of_files_for_a_letter(zadato_slovo)
        lista_nula = [zadato_slovo]
        for fajlovi_u_listi in lista:
            ukupan_broj_vrednosti, broj_nula =
                    prebroj_nulte_vrednosti(project_folder + zadato_slovo +
                    "\\" + fajlovi_u_listi)
            lista_nula.append(broj_nula/ukupan_broj_vrednosti*100)

        lista.insert(0, zadato_slovo)
        write.writerows([lista])
        write.writerows([lista_nula])
        lista = []
    file.close()
```



Slika 21. Funkcija za pronalaženje neispravnih video zapisa

Rezultat rada funkcije *„prebroj_nulte_vrednosti(naziv_fajla)“* prikazanoj u slici Slika 21, je generisana datoteka „*sva_slova.csv*“. Sadržaj rezultujuće datoteke je prikazan na slici Slika 22. Za svako slovo Azbuke i svaki pojedinačni snimak je izračunato procentualno učešće neprepoznatih ključnih tačaka koji su prikazani u kolonama koje počinju brojem 1. „*Ruke*“ kolona označava broj ruku koji je potreban da bi se gest prikazao (jedna ili dve ruke). „*AVG*“ polje prikazuje prosečnu vrednost procenata neprepoznatih ključnih tačaka. Polja „*MIN*“ i „*MAX*“ sadrže ekstremne vrednosti za zadato slovo. Polja koja sadrže vrednosti su automatski obojena, zbog lakše identifikacije ekstrema, pa tamnija polja sadrže visoke vrednosti, dok svetla polja prikazuju niske vrednosti.

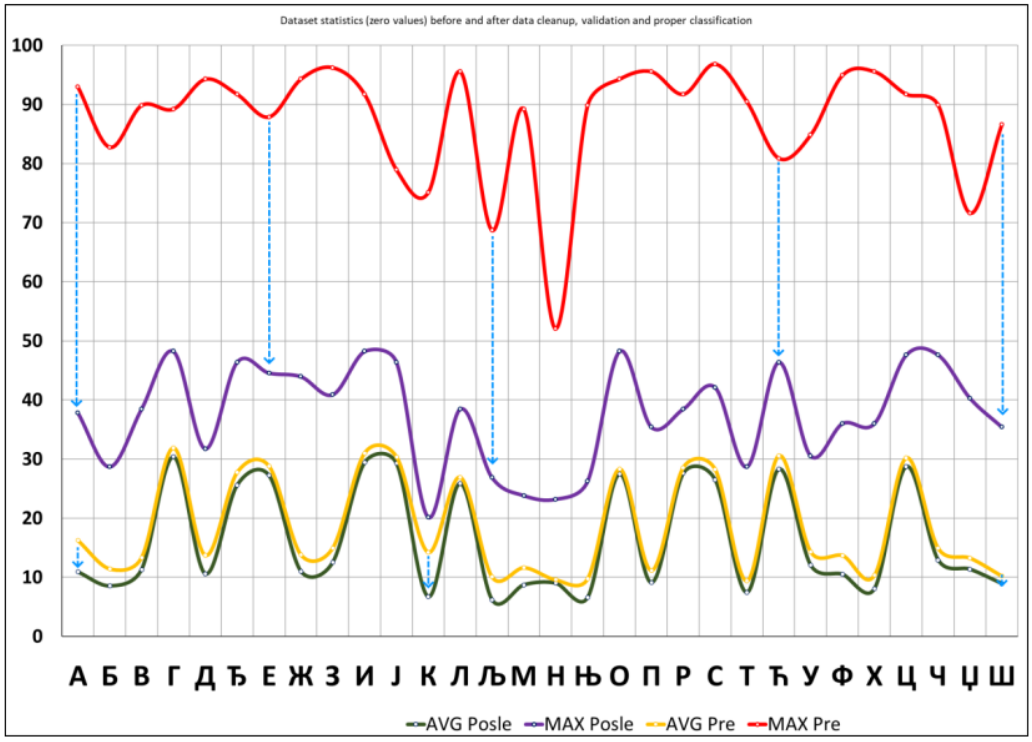
Na primer, za slovo A i snimak u koloni broj 13, procenat neprepoznatih ključnih tačaka je 93%, što podrazumeva da je snimak kreiran greškom i da je prepoznata jedna ruka u kadru samo u kratkom vremenskom periodu. Taj rezultat podrazumeva uklanjanje tog snimka iz baze. Na primeru slova Ж (ZH), vidno odstupaju vrednosti snimaka pod rednim brojevima 10 i 19. Uklanjanje ekstremnih vrednosti iz konačne baze podataka je neophodno sprovesti, pošto značajno pomaže unapređenju kvaliteta i homogenosti podataka koji će se kasnije koristiti u kreiranju modela mašinskog učenja. Zadržavanjem nepravilno snimljenih podataka u okviru baze, budući modeli bi bili pogrešno obučeni, što bi rezultiralo većoj nepreciznosti i nepouzdanosti.

Slovo	Ruke	AVG	MIN	MAX	Broj	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
A=A						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
A=A	2	16.26	0.00	92.99	335	35	48	26	41	21	1	13	48	19	25	14	17	93	52	47	59	48	47	34	36
B=Б						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
B=Б	2	11.45	0.00	82.76	243	10	4	4	3	7	8	8	3	4	7	37	4	23	6	7	6	2	2	15	5
V=В						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
V=В	2	13.34	0.00	89.83	245	21	5	4	20	20	22	24	16	10	23	18	4	5	2	6	7	8	4	6	5
G=Г						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
G=Г	1	31.88	23.81	89.22	233	73	40	29	26	25	26	29	41	24	27	26	28	48	48	48	42	28	31	29	24
D=Д						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
D=Д	2	13.74	0.00	94.27	231	64	26	4	7	2	5	10	4	5	8	7	29	13	21	14	10	18	17	24	26
DJ=Ђ						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
DJ=Ђ	1	27.74	1.83	91.72	242	44	26	26	28	26	28	28	29	30	24	27	25	37	36	29	35	37	36	34	37
E=E						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
E=E	1	28.78	15.26	87.88	247	73	42	26	27	26	24	25	26	25	26	24	24	24	30	28	31	32	36	32	29
ZH=Ж						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
ZH=Ж	2	13.84	0.61	94.27	240	10	12	11	9	6	9	10	6	7	34	12	12	11	16	13	15	14	14	23	3
Z=З						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
Z=З	2	14.97	0.00	96.22	238	96	10	16	9	18	9	8	7	11	13	9	39	38	20	26	20	25	16	11	10
I=И						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
I=И	1	31.00	23.20	91.72	253	50	34	26	25	26	26	24	32	25	24	25	44	35	32	35	38	38	34	24	27
J=J						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
J=J	1	30.36	20.76	78.92	260	43	24	23	23	24	24	26	27	27	28	23	31	27	25	30	28	29	29	26	27
K=K						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
K=K	2	14.25	0.00	75.09	261	48	40	1	42	1	1	0	41	21	47	48	25	47	16	44	39	45	34	47	46
L=Л						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
L=Л	1	26.91	0.61	95.55	475	26	26	26	26	23	24	24	24	27	24	26	25	26	23	25	25	51	52	24	29
LJ=Љ						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
LJ=Љ	2	10.07	0.00	68.69	215	14	37	48	48	6	1	0	1	1	1	0	1	1	4	37	45	46	19	31	27
M=M						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
M=M	2	11.64	0.00	89.16	248	15	0	12	1	2	43	12	2	10	2	4	5	36	2	89	48	13	12	7	6
N=H						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
N=H	2	9.58	0.00	52.06	233	40	5	1	4	5	4	2	3	3	1	4	3	2	2	2	1	0	2	2	14
NJ=Њ						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
NJ=Њ	2	9.77	0.00	89.83	227	47	0	1	2	2	1	1	0	1	1	2	6	0	0	42	34	0	4	5	4
O=O						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
O=O	1	28.29	6.10	94.27	428	47	34	8	27	56	29	24	44	24	24	25	27	30	29	26	25	27	24	27	42
P=П						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
P=П	2	11.17	0.00	95.55	384	45	4	4	4	2	2	4	3	4	2	2	3	3	79	23	13	13	12	13	38
R=P						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
R=P	1	28.56	9.16	91.72	308	35	21	30	30	27	26	18	23	9	31	14	29	31	32	37	27	26	25	26	27
S=C						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
S=C	1	28.26	10.38	96.83	315	48	24	25	24	24	25	25	24	24	24	24	26	24	24	42	29	24	24	27	24
T=T						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
T=T	2	9.42	0.00	90.44	317	3	2	2	4	4	2	1	1	1	2	0	1	19	5	27	10	7	4	7	6
CC=Ћ						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
CC=Ћ	1	30.55	10.99	80.87	319	45	45	44	33	33	29	31	32	31	35	32	31	28	38	46	47	47	48	29	31
U=У						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
U=У	2	14.31	0.00	84.83	221	48	6	10	4	4	2	10	12	5	5	7	9	16	20	11	56	29	26	25	27
F=Ф						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
F=Ф	2	13.70	0.61	94.94	203	24	13	12	27	24	29	26	26	23	24	21	16	31	8	5	10	7	13	11	16
H=X						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
H=X	2	10.35	0.00	95.55	322	48	46	0	1	0	1	1	48	1	2	3	9	4	2	6	1	4	2	0	2
C=Ц						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
C=Ц	1	30.19	23.81	91.72	299	47	26	24	24	27	24	27	31	25	27	24	27	84	35	31	28	32	26	27	32
CH=Ч						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
CH=Ч	2	14.91	0.00	89.88	283	3	0	7	1	1	1	4	2	0	3	62	45	8	3	7	2	4	4	4	8
DZ=Џ						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
DZ=Џ	2	13.27	0.00	71.60	273	45	1	4	5	1	5	2	5	5	2	8	56	7	4	4	3	3	3	3	2
SH=Ш						170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170	170
SH=Ш	2	10.22	0.00	86.60	268	13	5	6	9	1	8	6	7	7	1	0	0	87	23	21	20	5	6	8	17

Slika 22. Procenat neprepoznatih ključnih tačaka po slovu, po snimku

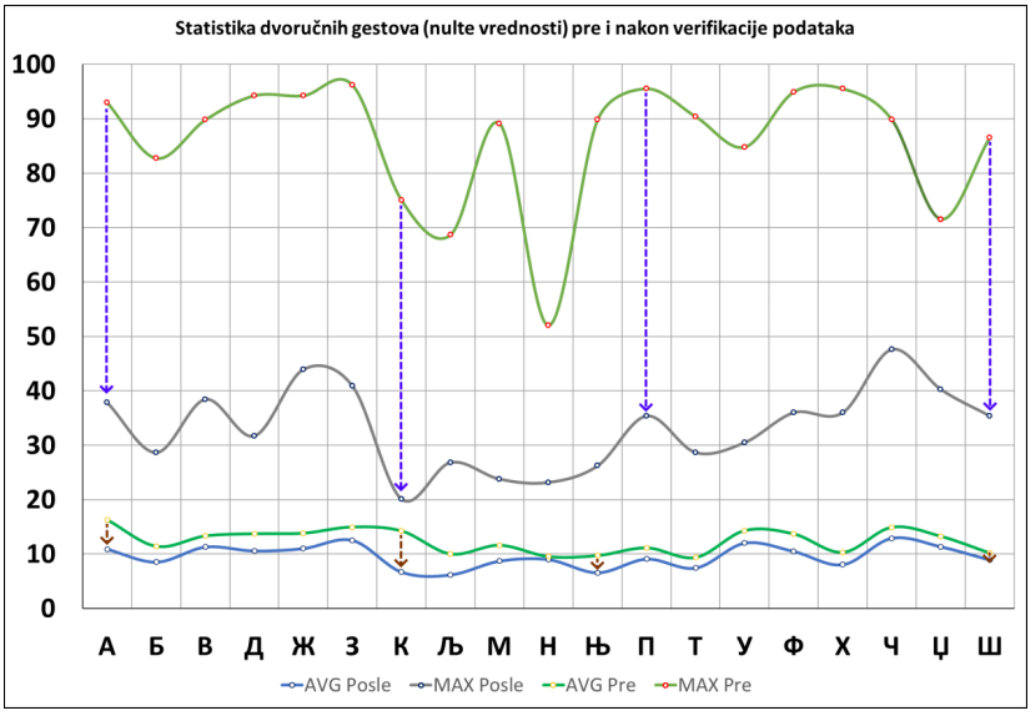
Primetna je razlika u broju neprepoznatih ključnih tačaka između gestova koji se prikazuju jednom ili sa dve ruke. Veći broj neprepoznatih tačaka sadrže gestovi koji se izvode samo sa jednom rukom, pošto je jedna ruka uvek van kadra za snimanje. Prosek od oko 9.6% nultih vrednosti za gestove koji se izvode pomoću dve ruke je značajno manji od jednoručnih gestova koji imaju prosek od oko 27.8% nula po video snimku.

Slika 23 prikazuje prosečan procenat nultih vrednosti po slovu, pre uklanjanja nepravilno snimljenih video snimaka i nakon uklanjanja. Crvena linija prikazuje maksimalne procente po slovu, pre čišćenja baze, dok plava predstavlja maksimume nakon čišćenja. Značajno umanjenje ekstremnih vrednosti obezbeđuje kvalitetniju finalnu bazu podataka. Žuta linija prikazuje prosečne procentualne nulte vrednosti pre čišćenja, dok zelena uzima vrednosti nakon čišćenja.



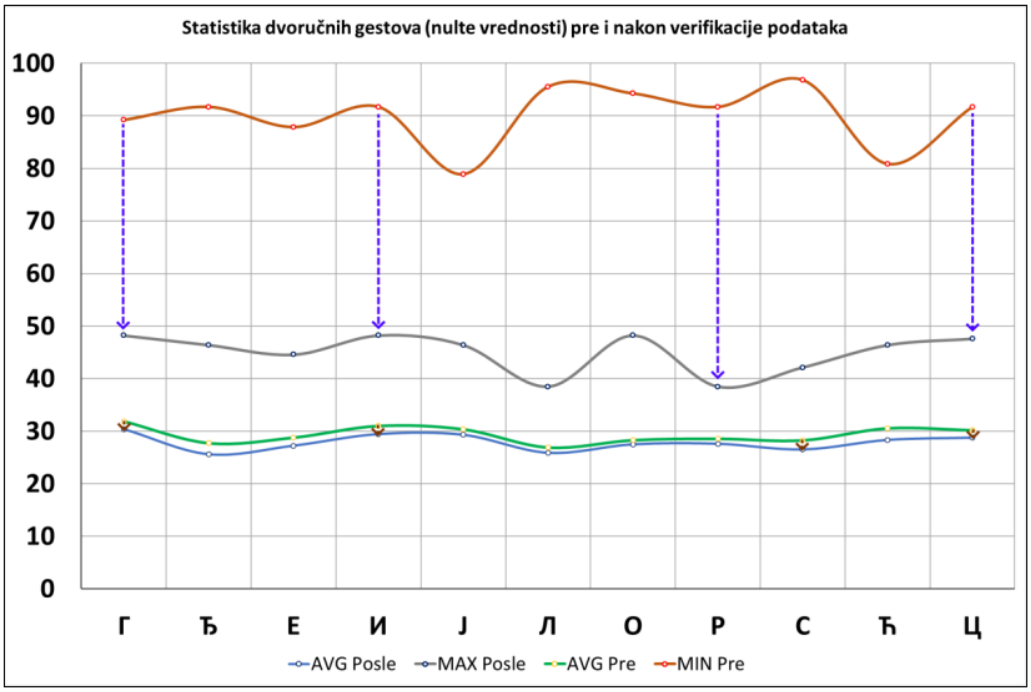
Slika 23. Prikaz broja video snimaka pre provere i nakon brisanja neprihvatljivih snimaka

Na slici Slika 24 je prikazana promena u kvalitetu podataka preuzetih za bazu podataka dvoručnih znakovnih gestova nakon uklanjanja neodgovarajućih. Primetan je značajan pad u svakom od slova Azbuke u procentu koji pokazuje neprepoznate ključne tačke u video zapisima.



Slika 24. Pre i nakon čišćenja za dvoručne gestove

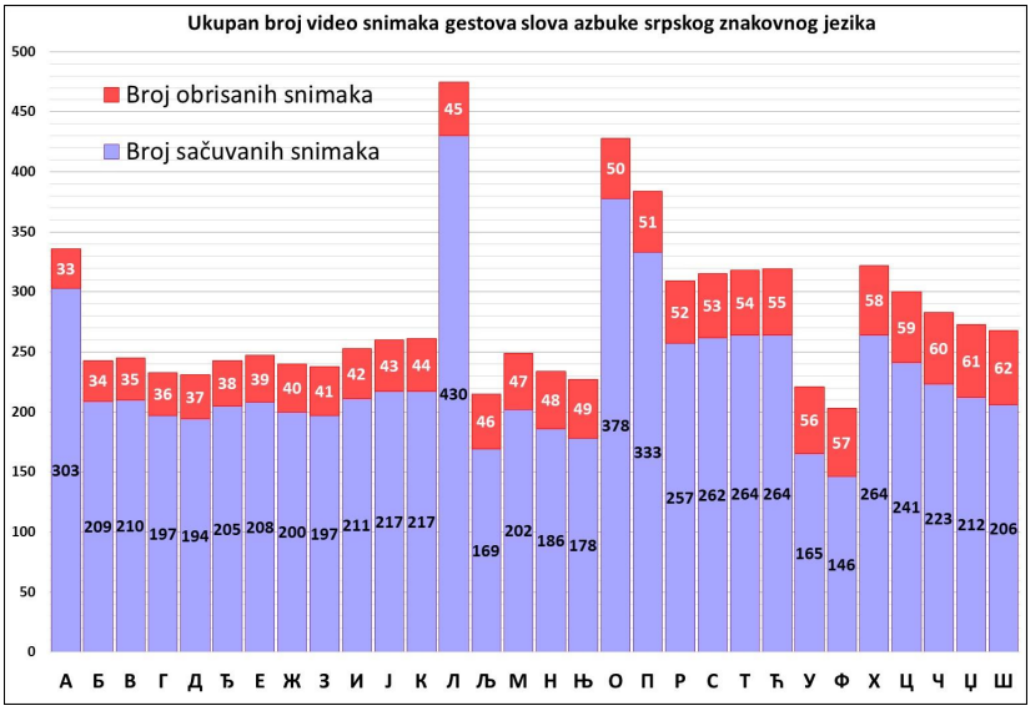
Slika 25 pokazuje pad u procentu snimaka sa nultim vrednostima u jednoručnim gestovima srpskog znakovnog jezika. Završni procenti su veći od gestova za dvoručne, pošto je jedna ruka uvek van kadra, pa sa nje ključne tačke nikada nisu prepoznate, pa se beleže nule.



Slika 25. Pre i nakon čišćenja za jednoručne gestove

3.3.5. Konačni oblik i veličina baze podataka

Broj snimaka koji ne zadovoljavaju uslove su uklonjeni iz konačne baze podataka, a na slici Slika 26 su prikazani crvenom bojom. Ispravni fajlovi, koji zadovoljavaju kriterijume, a zadržani su za konačnu bazu, su obojeni plavom bojom (Radaković M. et al, 2024).



Slika 26. Broj odbačenih snimaka i konačni broj usvojenih snimaka za bazu podataka

Ukupan broj snimljenih video klipova je 8.346, od kojih je uklonjeno i obrisano neodgovarajućih i greškom snimljenih 621, što je podrazumeva da je u finalnoj bazi podataka sačuvano 7.725 fajlova. U proseku, svako slovo je prikazano i računarski obrađeno oko 257 puta.

3.4. Diskusija rezultata snimanja gestova koji predstavljaju slova azbuke srpskog znakovnog jezika

Skup podataka razvijen i prikazan u okviru ove disertacije se bazira na gestovima SZJ i može se koristiti za izgradnju različitih modela mašinskog učenja. Ključne tačke koje opisuju prikazane gestove su izdvojene iz video klipova i sačuvane u numeričkim NumPy fajlovima (Sapre A., Vartak S., 2020). Snimljeni video zapisi se mogu koristiti i naknadno, kao osnova za drugačiju obradu podataka korišćenjem drugih tehnika računarske vizije. Za potrebe ovog istraživanja, korišćena je MediaPipe platforma za prepoznavanje koordinata ključnih tačaka koje određuju pokrete demonstratora.

Snimci su prikupljeni od 41 pojedinca i to u učionicama, kancelarijama ili kod kuće - koristeći običan laptop računar. Na ovaj metod prikupljanja podataka ne utiču

negativno spoljni uslovi i može se replicirati u različitim okruženjima. Učesnik u snimanje sedi na stolici, u prirodnom položaju, dok se računar nalazi na stolu, ispred njega. Sa ovakvom postavkom, moguće je naknadno proširiti skup podataka, pošto se lako mogu ustanoviti slični radni uslovi. Dodatno, prilikom korišćenja modela mašinskog učenja baziranog na ovakvom skupu podataka za prepoznavanje gestova SZJ, korisnici će biti u sličnom, skoro istom, položaju kao prilikom procesa prikupljanja podataka. Zbog toga će model biti pouzdaniji i proizvoditi preciznije rezultate.

Da bi se postigli kvalitetniji rezultati i smanjila količina beskorisnih i greškom snimljenih video zapisa, pripremljen je instruktivni video, koji je prikazan svim učesnicima pre učestvovanja u istraživanju. Standardizacija gestova, dinamika pokreta i standardizacija postupka snimanja u celosti - rezultirali su generisanjem efikasnijeg skupa podataka SZJ sa malim procentom grešaka.

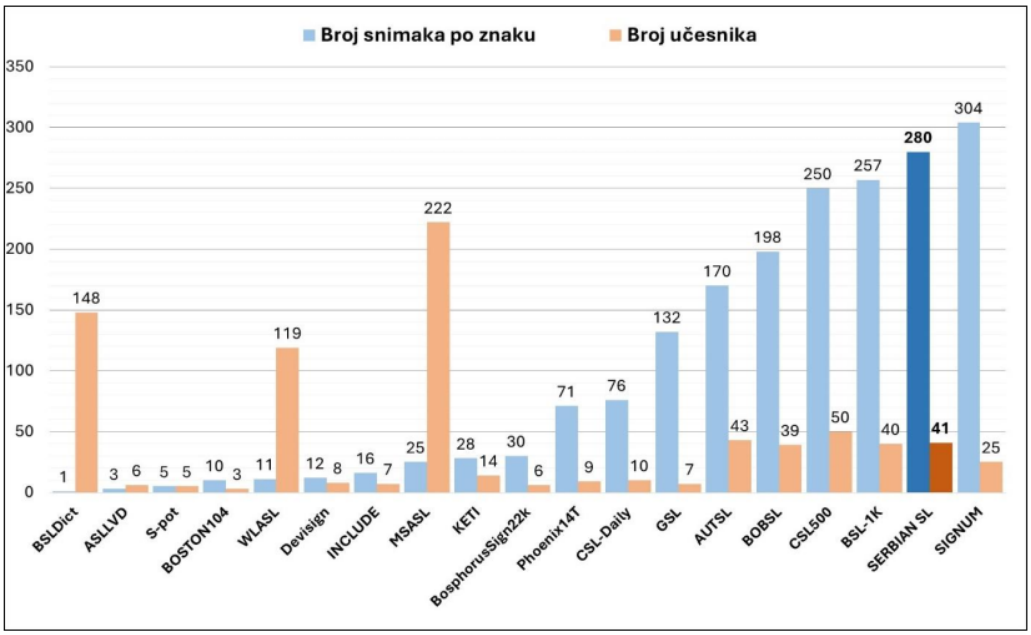
Prosečno trajanje snimanja sastojalo se od oko 20% vremena u kojem ruke ulaze u kadar, 60% vremena snimka je sadržalo sam pokret znakovnog jezika rukama i 20% vremena su učesnici koristili da uklone ruke iz kadra. Pošto se svaki snimak sastoji od 40 pojedinačnih slika, budući kreatori modela ML mogu da prilagode parametre i kvalitet baze pre korišćenja u svojim projektima, prema svojim potrebama. Korišćenjem tehnika računarske vizije, trenutak kada se jedna od ruku pojavi u kadru, snimanje od 40 slika počinje automatski, čime se proces generisanja baze ubrzava i čini efikasnijim.

Kako su različiti učesnici snimanja ulazili i izlazili iz kadra, primećene su i greške prilikom snimanja. Konačan broj od 621 video klipova uklonjen je od ukupno 8.436. Zajedno sa snimcima koji nisu bili u skladu sa kriterijumima SZJ (pogrešna ruka, nepravilan pokret, itd.), samo 7.44% snimaka napravljeno je greškom, što je vrlo dobar rezultat.

Pouzdana mera kvaliteta zapisanih podataka je broj nula vrednosti u identifikovanim slikama ključnih tačaka ruku i tela. Jedan koristan način za identifikaciju snimaka koji nisu u skladu sa očekivanim kvalitetom zapisa jeste obrada svih rezultujućih datoteka i identifikacija onih koji imaju veći procenat nula vrednosti ili značajno odstupaju od proseka. Vizuelna kontrola video datoteka korišćena je kao dodatna mera, kako bi se obezbedilo uklanjanje snimaka iz konačnog skupa

podataka, nakon numeričke evaluacije i označavanja datoteka sa ekstremnim vrednostima.

Poređenje baze SZJ sa drugim sličnim bazama podataka je prikazano na slici Slika 27. Broj učesnika i broj snimaka po obrađenom gestu je u vrhu svih pronađenih baza, koje su prikazane i obrađene u poglavlju koje se bavi pregledom literature.

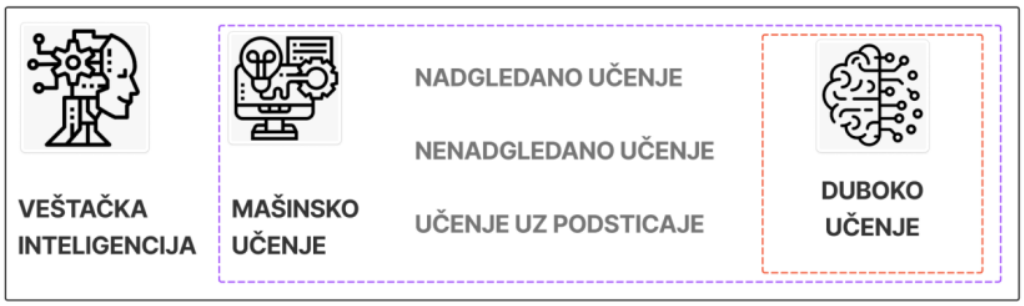


Slika 27. Poređenje kreirane baze podataka SZJ sa drugim bazama

Rezultat angažmana demonstratora gestova SZJ je sveobuhvatna, kvalitetna i obimna baza podataka slova srpske Azbuke, koja je prekontrolisana, očišćena, obrađena i pripremljena za korišćenje u budućim različitim modelima mašinskog učenja. Ona se može koristiti u izvornom obliku, izmeniti, prilagoditi, a takođe se može raditi i na njenom proširivanju i unapređenju. Zbog toga što je omogućeno da bude javno dostupna, očekuje se njena kontinuirana dalja dorada, unapređenje i uvećanje. Takođe se očekuje i veliki broj različitih modela mašinskog učenja kreiranih na osnovu nje, za različite namene.

4. Kreiranje modela mašinskog učenja

Termini „veštačka inteligencija“, „mašinsko učenje“ i „duboko učenje“ se koriste često, naizmenično i nasumično, ali među njima postoje jasne razlike (slika Slika 28). Termin „veštačka inteligencija“ je prikladan za situacije kada se govori o mašinama koje imitiraju ljudske mogućnosti za rešavanje problema ili donošenja odluka. Veštačka inteligencija je uopšteni termin, koji u svom okviru sadrži mašinsko i duboko učenje. Mašinsko učenje predstavlja poseban deo veštačke inteligencije koji je specijalizovan za korišćenje različitih algoritama sposobnih za samostalno učenje, za proizvodnju znanja generisanog iz podataka, radi predviđanja rezultata. Duboko učenje predstavlja poseban podskup mašinskog učenja, najčešće značajno uvećanih parametara i neuronskih slojeva, koji funkcioniše bez potrebe za ljudskom intervencijom, a u cilju grupisanja podataka i uočavanja ljudskom oku teško uočljivih zakonitosti u rezultatima (Kaiming H. et al., 2016). Da bi se model mašinskog učenja tretirao kao model dubokog učenja, podrazumeva se da mu je broj neuronskih slojeva veći od tri, kao i da je u pitanju nenadgledano učenje.



Slika 28. Odnos između veštačke inteligencije, mašinskog učenja i dubokog učenja

4.1. Teorijska osnova mašinskog učenja

Mašinsko učenje (MU), kao podskup veštačke inteligencije, može se podeliti na tri osnovne grupe (Batta M., 2018). Kao što je prikazano na slici Slika 29, MU se deli na nadgledano učenje (*supervised learning*), nenadgledano učenje (unsupervised learning) i učenje uz podsticaje (*reinforced learning*).



Slika 29. Klasifikacija mašinskog učenja

Nadgledano učenje podrazumeva da su podaci pre korišćenja za kreiranje modela mašinskog učenja označeni, imenovani ili klasifikovani na način na koji se precizno opisuju karakteristike podataka. Označavanje i klasifikacija podataka može biti jednostavno ili može da sadrži i kompleksne atribute, ali je važno da na potreban način podatke opiše (Nhangumbe M., 2023).

Nenadgledano učenje se koristi za otkrivanje skrivenih zakonitosti ili grupa podataka u velikom broju informacija, bez potrebe za ljudskom intervencijom ili pomoći. Podaci koji se koriste prethodno ni na koji način nisu označeni, prepoznati ili unapred grupisani. Kao i nadgledano učenje, tehnike za nenadgledano učenje se takođe mogu podeliti na dve: grupisanje podataka (*clustering*) i smanjenje dimenzionalnosti (*dimensionality reduction*). Grupisanje podataka se odnosi na proces organizovanja objekata u grupe, na osnovu sličnosti između njih. Cilj ove tehnike je da se postigne da objekti unutar iste grupe (klastera) budu što sličniji jedni drugima, a da objekti iz različitih grupa budu što različitiji je moguće. Smanjenje dimenzionalnosti se odnosi na tehniku koja se koristi za smanjenje broja atributa ili karakteristika u skupu podataka. Cilj je da se smanji broj redundantnih i nevažnih informacija, što ubrzava i olakšava analizu podataka, a za posledicu ima poboljšanje performansi modela.

Učenje uz podsticaje predstavlja vid delimično nadgledanog učenja, u kome automatizovani agent (računarski program), koji funkcioniše u tačno definisanom okruženju, na osnovu generisanih rezultata, dobija nagrade ili kazne od tog okruženja (Ding Z., 2020). Da li će agent da dobije nagradu ili kaznu, zavisi od preciznosti generisanih rezultata. Na osnovu velikog broja ovih povratnih informacija,

agent će vremenom korigovati svoj rad i „naučiti“ da preciznije i pouzdanije ispunjava postavljeni zadatak.

4.1.1. Arhitektura modela

Prvi korak u eksperimentisanju sa modelom je uključivao prilagođavanje opšte arhitekture modela obliku izvornih podataka. Varijacije u broju i konfiguraciji LSTM slojeva, uključujući broj jedinica unutar svakog sloja, značajno utiču na sposobnost modela da „nauči“ složene zavisnosti unutar podataka. Eksperimentisano je i sa uvođenjem dvosmernih *Bidirectional LTSM* slojeva, kako bi se omogućilo modelu da bolje „razume“ kontekst, tako što prolazi kroz sekvencijalne podatke u oba smera – unapred i unazad.

4.1.2. Sprečavanje prevelikog prilagođavanja modela podacima

Za potrebe kreiranja modela iz ove disertacije, uzete su u obzir i tehnike regularizacije. Regularizacija je jedna od važnih komponenti u oblasti mašinskog učenja, koja se koristi za sprečavanje prevelikog prilagođavanja modela podacima koji su obezbeđeni za obučavanje. Do prevelikog prilagođavanja dolazi kada kreirani model postane previše kompleksan i nauči detalje i šumove u setu podataka za obučavanje do te mere da mu se smanjuje sposobnost rada na novim, prethodno nikad viđenim podacima (Caruana R., 2000). Tehnike regularizacije efikasno ograničavaju složenost generisanog modela i osiguravaju da se univerzalnije obuči i pruži bolje performanse u radu. Tehnike koje spadaju u oblast regularizacije su L1 (*Lasso*), L2 (*Ridge*), elastična mreža (*Elastic Net*), proređivanje podataka (*Dropout*), rano zaustavljanje obučavanja (*Early Stopping*). Primena regularizacije zahteva pažljiv izbor hiper-parametara, kao što su intenzitet regularizacije za L1 i L2, verovatnoća isključivanja za proređivanje podataka i kriterijumi za rano zaustavljanje. Optimalni hiper-parametri u modelu često zavise od specifičnog skupa podataka i problema, te se obično naknadno određuju i prilagođavaju kroz proces provere sa posebno odvojenim podskupom podataka za tu namenu.

4.1.3. Optimizovanje hiper-parametara modela

Optimizacija hiper-parametara, uključujući stopu učenja, veličinu serije (*batch size*) i broj epoha (jedan prolaz modela kroz ceo skup podataka), ključna je za finu kalibraciju procesa obuke. Tehnike kao što su „nasumična pretraga“ (*Random*

Search) i Bajesova optimizacija (*Bayesian Optimization*) mogu se koristiti za sistematsku analizu hiper-parametara i identifikovanje optimalnih konfiguracija (Feurer M., Hutter F., 2019

Nasumična pretraga je tehnika optimizacije hiper-parametara koja uključuje nasumično odabiranje vrednosti za hiper-parametre iz unapred definisanih opsega ili raspodela. Ovaj pristup se često koristi zbog svoje jednostavnosti i sposobnosti da otkrije dobra rešenja u velikim i složenim modelima. Iako nasumična pretraga može biti manje efikasna od drugih metoda optimizacije, jer ne koristi informacije iz prethodnih evaluacija, njegova glavna prednost leži u sposobnosti da brzo pokrije širok spektar mogućih vrednosti.

Bajesova optimizacija je napredna tehnika za optimizaciju hiper-parametara koja koristi Bajesovu statistiku da bi iterativno izgradila model prostora hiper-parametara i efikasnije pretražila njihove najbolje vrednosti. Ova metoda se oslanja na minimalizovanje funkcije gubitaka (*loss function*), kako bi procenila koje vrednosti hiper-parametara bi mogle dovesti do najbolje performanse modela. Ova tehnika je efikasna u pronalaženju optimalnih vrednosti hiper-parametara sa malim brojem evaluacija, pošto koristi prethodno dobijene rezultate, što je čini pogodnom za velike skupove podataka i kompleksne modele. Ona zbog toga zahteva složenije matematičke i statističke metode u odnosu na optimizaciju nasumične pretrage.

4.1.4. Metode naprednih tehnika

Napredne tehnike u mašinskom učenju obuhvataju širok spektar metoda, dizajniranih da poboljšaju performanse, efikasnost i funkcionalnost modela. Zasnivaju se na primeni postojećeg znanja stečenog na jednom zadatku za poboljšanje performansi na drugom, generisanju novih uzoraka iz postojećih, fokusiranju na ograničene delove podataka, kombinovanju tehnika predviđanja iz različitih modela i slično. Veliki broj trenutno dostupnih i razvijenih metoda ovog tipa pomažu rešavanju složenih problema u mašinskom učenju, neprestano se razvijaju i adaptiraju za nove izazove i zadatke.

4.1.5. Procena rada modela

Evaluacija modela je ključan korak u procesu razvoja mašinskog učenja, koji omogućava ocenjivanje performansi modela i njegovo poređenje sa drugim modelima. Za potrebe ovog istraživanja i disertacije, kreiran je veliki broj različitih

modela koji su bili više ili manje uspešni. Ovaj proces u zavisnosti od vrste modela može da uključi korišćenje specifičnih merila, prikazanih u tabeli Tabela 4.

Tabela 4. Procena rada modela

Klasifikacioni zadaci	Regresioni zadaci
- tačnost (*accuracy*) - preciznost (*precision*) - odziv (*recall*) - F1 rezultat	- srednja kvadratna greška (*mean squared error*) - srednja apsolutna greška (*mean absolute error*)

Važno je primeniti evaluaciju na odvojenom skupu podataka koji nije korišćen tokom obuke, kako bi se osigurala objektivnost ocene. Unakrsna provera, posebno tehnika „*k-fold cross validation*“, često se koristi za dobijanje pouzdanije ocene performansi modela. Ova tehnika evaluacije modela mašinskog učenja osigurava bolje iskorišćenje podataka podelom celokupnog skupa podataka na *k* manjih podskupova. Proces se odvija tako što se model obučava na *k-1* podskupova, dok se preostali podskup koristi kao skup za procenu performansi modela. Ovaj postupak se ponavlja *k* puta, pri čemu se svaki podskup jednom koristi kao podskup za testiranje. Na kraju se uzima prosečna vrednost rezultata evaluacije iz svih *k* iteracija, kako bi se dobila sveobuhvatna ocena performansi modela. Čest izbor za k je 10, iako se ta vrednost može prilagođavati u zavisnosti od veličine i prirode skupa podataka.

Evaluacija modela ne samo da pomaže u određivanju koliko je model dobar u predviđanjima, već pruža i uvid u njegove moguće slabosti, vodeći ka daljem finom podešavanju i poboljšanju. Procena rada modela kroz rigorozno testiranje, uključujući upotrebu odvojenih skupova podataka za obučavanje, proveru i testiranje, omogućava i detaljnu analizu performansi. Upotreba pomenutih merila, pruža uvid u sposobnosti modela i područja koja zahtevaju dalje poboljšanje.

4.1.6. Uporedna analiza različitih modela

Uporedna analiza modela mašinskog učenja predstavlja proces evaluacije i poređenja performansi različitih modela da bi se identifikovao onaj koji najbolje rešava određeni problem. Ovaj proces obuhvata upotrebu jedinstvenih skupova podataka za obuku i testiranje, kao i istih mera evaluacije, poput tačnosti, preciznosti,

odziva, F1 rezultata ili srednje kvadratne greške, kako bi se osigurala objektivnost i pravednost u poređenju. Uporedna analiza može uključiti različite tipove modela, hiper-parametre, tehnike obrade podataka i strategije obučavanja modela, što omogućava dublje razumevanje prednosti i slabosti svakog modela u kontekstu specifičnog problema. Osim toga, uporedna analiza može pružiti uvide u složenost modela, vreme obuke i izvršavanja, kao i njihovu sposobnost generalizacije na nikada pre viđenim podacima. Ovaj pristup je ključan za donošenje informisanih odluka prilikom odabira optimalnog modela za implementaciju u realnim aplikacijama.

Kroz sistematsko eksperimentisanje i primenu različitih strategija izrade modela mašinskog učenja, ovaj rad teži ka razvoju funkcionalnog i preciznog modela za klasifikaciju znakovnog jezika, sa naglaskom na optimizaciju performansi i prilagodljivosti. Kontinuirani napredak i iterativno poboljšanje, uz upotrebu najsavremenijih tehnika i pristupa, su ključni za postizanje visokog nivoa upotrebljivosti generisanog modela. U poglavlju 4.2. je predstavljeno poređenje modela sa kojima je eksperimentisano, a kao glavni pokazatelj je korišćena preciznost predviđanja (*Accuracy*).

4.1.7. Principi poređenja performansi kreiranih modela

Kao što je prikazano na slici Slika 17 i u jednačini 9, baza podataka koja se koristi za kreiranje modela mašinskog učenja sadrži oko 300 snimaka po znakovnom gestu, dok se svaki od snimaka sastoji od 10.320 numeričkih vrednosti. Ove numeričke vrednosti su sačuvane u pojedinačnim datotekama u obliku jednodimenzionalnih *NumPy* nizova, zbog lakše manipulacije i korišćenja. Svaki pojedinačni fajl sadrži 258 numeričkih koordinatnih vrednosti ključnih tačaka za po 40 pojedinačnih slika, što ukupno čini 10.320 numerika. Uzimajući u obzir strukturu, veličinu i tip podataka, izvedeno je istraživanje različitih pristupa i tehnika koje se mogu koristiti za kreiranje odgovarajućeg modela mašinskog učenja za ovakav tip podataka. Model za prepoznavanje gestova znakovnog jezika je potrebno bazirati na osnovu sekvencijalnih, vremenski povezanih podataka dobijenih iz serija slika (pokreti ruku i tela). Pored vremenske komponente, podaci sadrže i numeričke vrednosti, koje predstavljaju prostorne veličine (koordinate ključnih tačaka). Zbog prisutne vremenske komponente u podacima, u ovom radu su korišćene tehnike zasnovane na rekurentnim neuronskim mrežama (RNN), sa posebnim fokusom na LSTM (*Long Short-Term Memory)* mreže, a u cilju poboljšanja rezultata klasifikacije.

Eksperimentisano je sa velikim brojem različitih tehnika i kombinacija modela, a u cilju pronalaženja sistema koji proizvodi najbolje rezultate (Desai R. J., 2020).

U kontekstu obrade i analize jednodimenzionalnih nizova koji sadrže ključne tačke prstiju i tela, specifičan izazov predstavlja ekstrakcija relevantnih karakteristika i njihova dalja interpretacija. Ovaj skup podataka, iako bogat informacijama, zahteva korišćenje pažljivo odabranih metoda mašinskog učenja koje mogu efikasno da obrade i analiziraju takav skup podataka.

Pri odabiru arhitekture modela, posebno broja slojeva, potrebno je uzeti u obzir složenost i prirodu podataka. Modeli sa većim brojem slojeva, imaju sposobnost da uče složenije obrasce, ali sa povećanim rizikom od prevelikog prilagođavanja podacima. To je posebno uočljivo kod baza sa ograničenim skupovima podataka. Zbog toga, modeli kreirani u početku projektovanja, obuhvataju upotrebu manjeg broja slojeva, sa postepenim povećanjem tokom procesa eksperimentisanja, sve dok se ne postigne optimalna ravnoteža između sposobnosti funkcionisanja i tačnosti modela.

Za obradu jednodimenzionalnih nizova numeričkih vrednosti koji predstavljaju ključne tačke, slojevi koji mogu efikasno modelovati odnose između susednih i udaljenih tačaka unutar niza su od suštinskog značaja. Slojevi poput 1D konvolucionih slojeva (*Conv1D*) mogu biti posebno korisni u ovom kontekstu, jer omogućavaju modelu da nauči hijerarhijske karakteristike iz sekvencijalnih podataka. Osim toga, slojevi za smanjenje dimenzionalnosti, kao što su slojevi za grupisanje (*pooling layers*), mogu se koristiti za smanjenje broja parametara i poboljšanje efikasnosti modela, dok istovremeno čuvaju važne karakteristike.

Dubina modela, broj i vrsta neuronskih slojeva treba da budu usklađeni sa složenošću i tipom podataka, kao i sa ciljevima analize, uz neprestano praćenje rizika od prevelikog prilagođavanja. Eksperimentisanje i iterativni pristup, uz osvrt na postojeća istraživanja i naučnu praksu, ključni su za postizanje optimalnih rezultata.

Korišćena je ista programska struktura za sve kreirane modele mašinskog učenja, kao na slici Slika 30. Akcenat u procesu programiranja je stavljen na funkcionalnost i krajnji cilj aplikacije, dok se manje pažnje posvećivalo samoj estetici i strukturi kôda. Prikazani računarski kôd može biti unapređen dodavanjem funkcija za

automatsko prepoznavanje grešaka, omogućiti unos proizvoljnih promenljivih parametara, unaprediti i obogatiti grafički interfejs, itd.

```
import os
import numpy as np
import tensorflow as tf
from tensorflow import keras
from keras import layers
from datetime import datetime
from sklearn.model_selection import train_test_split

# AZBUKA - lista znakova za koje ce se kreirati baza podataka
sign_dict = {"A":"А", "B":"Б", "V":"В", "G":"Г", "D":"Д", "DJ":"Ђ", "E":"Е",
             "ZH":"Ж", "Z":"З", "I":"И", "J":"Ј", "K":"К", "L":"Л", "LJ":"Љ",
             "M":"М", "N":"Н", "NJ":"Њ", "O":"О", "P":"П", "R":"Р", "S":"С",
             "T":"Т", "CC":"Ћ", "U":"У", "F":"Ф", "H":"Х", "C":"Ц", "CH":"Ч",
             "DZ":"Џ", "SH":"Ш"}

# Pretvaranje kljuca u numericke vrednosti, prikladne za ML model
sign_dict_to_int = {"A":0, "B":1, "V":2, "G":3, "D":4, "DJ":5, "E":6, "ZH":7,
                    "Z":8, "I":9, "J":10, "K":11, "L":12, "LJ":13, "M":14,
                    "N":15, "NJ":16, "O":17, "P":18, "R":19, "S":20, "T":21,
                    "CC":22, "U":23, "F":24, "H":25, "C":26, "CH":27,
                    "DZ":28, "SH":29}

# Učitavanje svih pojedinacnih *.npy fajlova iz svih 30 foldera
# u jedan veliki fajl za kreiranje ML modela

def sacuvaj_sve_podatke(fajl_za_snimanje):
    print("# Učitavanje svih podataka u jedan veliki fajl - data.csv,
          a nazivi su u fajlu - labels.csv")
    print("Naziv fajla: ", fajl_za_snimanje)
    data = []
    labels = []
    vreme_pocetno = datetime.now()
    counter = 0
    for sign_label, value in sign_dict.items():
        folder_path = f"./{sign_label}/"
        for file_name in os.listdir(folder_path):
            if file_name.endswith(".npy"):
                file_path = os.path.join(folder_path, file_name)
                array = np.load(file_path)
                data.append(array)
                # U labels se upisuje redni broj slova,
                # a ne naziv slova (zbog modela)
                labels.append(sign_dict_to_int[sign_label])
                counter += 1
                print(f"{counter}. Pridružujem bazi sve fajlove iz\
                      foldera slova [{sign_label}]")

    vreme_nakon_ubacivanja_podataka = datetime.now()
    print(f"Pocetno vreme: {vreme_pocetno}")
    print(f"Vreme nakon ubacivanja podataka: {vreme_nakon_ubacivanja_podataka}")
    print(f"Proteklo vreme: {vreme_nakon_ubacivanja_podataka-vreme_pocetno}")

    print("Pretvori liste u NumPy nizove")
    data = np.array(data)
    labels = np.array(labels)
```

```
    try:
        print("Podaci se snimaju u fajlove...")
        np.savetxt(fajl_za_snimanje + "_label.csv", labels, delimiter=',')
        print(f"Snimljene su oznake slova u: {fajl_za_snimanje}_label.csv")
        np.savetxt(fajl_za_snimanje + "_data.csv", data, delimiter=',')
        print(f"Snimljeni su podaci u: {fajl_za_snimanje}_data.csv")
    except:
        print("Došlo je do greške tokom snimanja fajla. Fajl nije kreiran.")

    return data, labels

# Ucitavanje podataka iz prethodno pripremljenog jedinstvenog fajla za ML
def ucitaj_sve_podatke(fajl_za_ucitavanje):
    data = []
    labels = []

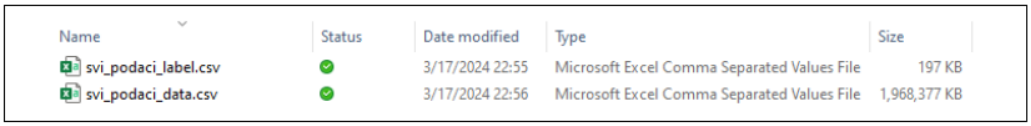
    try:
        print("Učitavanje podataka je u toku...")
        labels = np.loadtxt(fajl_za_ucitavanje + "_label.csv", delimiter=',')
        print(f"Učitane su oznake slova iz: {fajl_za_ucitavanje}_(labels).csv")
        data = np.loadtxt(fajl_za_ucitavanje + "_data.csv", delimiter=',')
        print(f"Učitani su podaci iz: {fajl_za_ucitavanje}_data.csv")
    except Exception as e:
        print(f"Došlo je do greške pri učitavanju:{e}. Podaci nisu učitani!")

    return data, labels
```

Slika 30. Programska osnova za kreiranje eksperimentalnih modela

Početni deo programa prikazanog na slici Slika 30 prikazuje module koju su korišćeni u programu. Predstavljeni su i rečnici kojima su definisana slova Azbuke i način na koji će ih program koristiti i tumačiti. Takođe, prikazan je i rečnik koji slova predstavlja u numerike koji se mogu koristiti u modelima mašinskog učenja za potrebe klasifikacije.

Drugi deo računarskog programa prikazuje funkciju „*snimi_sve_podatke*“, koja za cilj ima da pročita podatke iz unapred pripremljenih NumPy fajlova iz svih foldera i sačuva ih zajedno u jedan veliki fajl. Ovako kreirani fajl sa svim podacima će se kasnije koristiti za obučavanje modela. Sam proces čitanja svih pojedinačnih NumPy datoteka može da potraje za 7752 fajla oko 4 minuta, što ukoliko bi se ponavljalo za svaki eksperiment i probu modela bi značajno usporilo proces istraživanja. Čuvanjem svih podataka u jedan fajl i naknadnim čitanjem iz njega, proces se desetostruko ubrzava i smanjuje se potrebno vreme za isprobavanje različitih vrsta modela na ukupno oko 34 sekunde. Na slici Slika 31 su prikazane veličine i nazivi dva kreirana fajla pomoću ove funkcije.



Name	Status	Date modified	Type	Size
svi_podaci_label.csv	✓	3/17/2024 22:55	Microsoft Excel Comma Separated Values File	197 KB
svi_podaci_data.csv	✓	3/17/2024 22:56	Microsoft Excel Comma Separated Values File	1,968,377 KB

Slika 31. Kreirane datoteke sa svim grupisanim podacima i nazivima

Treći deo računarskog programa sadrži Python funkciju „*ucitaj_sve_podatke*“ koja za cilj ima da učita unapred pripremljen i snimljen fajl tipa „*csv*“ (*comma separated values*) sa podacima i pripadajućim oznakama za te podatke (prikazani na slici Slika 31). Funkcionisanje programa pri učitavanju sveobuhvatne baze je prikazano na slici Slika 32.

```
Da li želite da koristite ranije snimljene podatke ili da ponovo kreirate bazu iz
svih foldera:
[S]nimi sve u novi fajl sa svim podacima / [U]čitaj sve iz fajla sa svim podacima:
u
Učitavanje podataka je u toku...
Učitane su oznake slova iz: svi_podaci_(labels).csv
Učitani su podaci iz: svi_podaci_data.csv
data.shape: (7752, 10320)
labels.shape:  (7752,)
Podeli podatke u podskupove za obučavanje i testiranje
X_train.shape:  (6201, 10320)
X_test.shape:  (1551, 10320)
y_train.shape:  (6201,)
y_test.shape:  (1551,)
X_train_reshaped.shape:  (6201, 40, 258)
X_test_reshaped.shape:  (1551, 40, 258)
```

Slika 32. Učitavanje prethodno generisanog fajla sa svim podacima

U centralnom delu programa za kreiranje modela mašinskog učenja, prikazano je korišćenje prethodno kreiranih funkcija za grupisanje, snimanje i učitavanje podataka, preoblikovanje podataka iz jednodimenzionalnog niza u dvodimenzionalnu matricu, kao i sam postupak projektovanja modela, njegovog kreiranja i testiranja. Na slici Slika 33 je početni deo koda, koji prikazuje mogućnost korisnika da učita unapred pripremljeni jedinstveni fajl sa podacima ili kreira novi. Novi fajl je neophodno kreirati samo ukoliko je došlo do izmena u samoj bazi podataka. Ove izmene se mogu odnositi na povećanje broja snimljenih gestova ili njihovo smanjivanje, u slučaju izbacivanja fajlova koji se svojim kvalitetom ne uklapaju među ostale. Nakon učitavanja iz postojećeg jedinstvenog fajla ili učitavanja iz postojećih, podaci se dele u dva podskupa. Jedan od njih se koristi za obučavanje modela (80%), dok se preostali podaci (20%) ne koriste i čuvaju se za naknadno testiranje performansi modela. Uzorak koji se čuva za testiranje se ne ubacuje u

model sa namerom da nakon što je model kreiran, oni služe za ispitivanje preciznosti prognoziranja modela (Tan J., 2021). Podela podataka na podskupove za obučavanje i naknadno testiranje modela se u prikazanom kôdu vrši pomoću funkcije „*train_test_split*“, koja je deo modula „*Scikit-learn*“ (Pedregosa, F. et.al, 2011). Promenljiva „*test_size*“ definiše procenat podataka koji će biti sačuvan za testiranje, dok se promenljiva „*random_state*“ koristi u generisanju pseudo-slučajnih brojeva. Ovaj parametar se često koristi u mašinskom učenju i analizi podataka kako bi se osiguralo da se eksperimenti mogu ponoviti na isti način, tj. kako bi se osiguralo da se isti rezultati dobiju svaki put kada se isti kôd pokrene. Kada se postavi „*random_state*" promenljiva na određenu vrednost (kao npr. "42" u prikazanom programu), to znači da će se svaki put kada se funkcija pozove s tim istim parametrom, generisati isti pseudo-slučajni raspored podataka. Ovo je korisno za ponovljivost eksperimenata i rezultata, što je ključno u naučnim istraživanjima kako bi se potvrdila pouzdanost rezultata i omogućila provera rada modela.

```
if __name__ == "__main__":

    print("Da li želite da koristite ranije snimljene podatke
          ili da ponovo kreirate bazu iz svih foldera:")
    if input("[S]nimi sve u novi fajl sa svim podacima /
             [U]čitaj sve iz fajla sa svim podacima: ").capitalize() == "S":
        data, labels = sacuvaj_sve_podatke("svi_podaci")
    else:
        data, labels = ucitaj_sve_podatke("svi_podaci")

    # Prikaži formu podataka za data i labels
    print("data.shape:", data.shape)
    print("labels.shape: ", labels.shape)

    print("# Split the data into training and testing sets")
    X_train, X_test, y_train, y_test = train_test_split(data, labels,
                                                        test_size=0.2,
                                                        random_state=42)

    # Prikaži oblik podskupova za obučavanje (train) i testiranje (test)
    print("X_train.shape: ", X_train.shape)
    print("X_test.shape: ", X_test.shape)
    print("y_train.shape: ", y_train.shape)
    print("y_test.shape: ", y_test.shape)

    # Izmena oblika podataka - iz jednodimenzionalnog niza u 2-D matricu
    X_train_reshaped = X_train.reshape((y_train.size, 40, 258))
    # Novi oblik: (Broj uzoraka, frejmovi, ključne tačke)
    X_test_reshaped = X_test.reshape((y_test.size, 40, 258))
    # Nije potrebno menjati formu naziva (labels), ostaju 1-D: y_test, y_train
```

Slika 33. Centralni deo programa za kreiranje modela mašinskog učenja

Eksperimenti su vršeni sa jednodimenzionalnim i dvodimenzionalnim vrednostima, pa je deo programskog koda za izmenu oblika podataka korišćen samo u situaciji kada su se radila ispitivanja na dvodimenzionalnim matricama.

U delu programa prikazanom na slici Slika 34, definisan je sekvencijalni modela mašinskog učenja, korišćenjem Python biblioteke *Keras* (Chollet, F. et. al, 2015), koja je deo *TensorFlow-a* (Abadi, M. et. al, 2015), jedne od najpopularnijih biblioteka za duboko učenje.

```
print("Definisanje sekvencijalnog modela mašinskog učenja")
model = keras.models.Sequential([
    layers.LSTM(128, activation='relu', input_shape=(40, 258),
                return_sequences=True),
    # 128 LSTM jedinica, prosleđuje ceo niz izlaza sledećem sloju
    layers.LSTM(64, activation='relu'),
    # Drugi LSTM sloj vraca samo poslednji izlaz
    layers.Dense(30, activation='softmax')  # 30 izlaznih kategorija
])
```

Slika 34. Primer jednog od projektovanih modela mašinskog učenja

Linija koda u kojoj se poziva modul *Keras*, inicijalizuje sekvencijalni model. Sekvencijalni model u *Keras*-u omogućava kreiranje modela sloj po sloj, što je odgovarajući pristup za većinu problema. To znači da se svaki sloj dodaje jedan za drugim, pri čemu svaki sloj ima tačno jedan ulazni „*tenzor*“ i jedan izlazni „*tenzor*“.

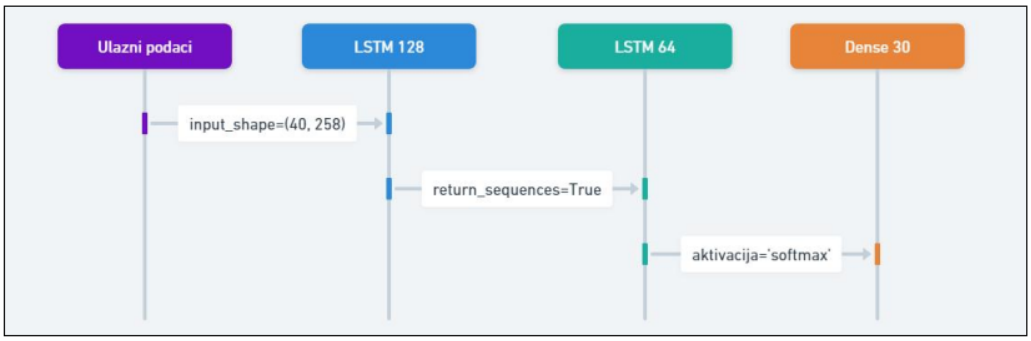
Tenzor je matematički objekat koji se može posmatrati kao generalizacija skalara, vektora i matrica (Panagakis Y., 2021). U kontekstu mašinskog učenja, tenzori su osnovne strukture podataka koje se koriste za predstavljanje i manipulisanje podacima. Tenzori mogu imati različite dimenzije:

- Tenzor nultog reda je skalar (pojedinačna vrednost – broj)
- Tenzor prvog reda je vektor (niz brojeva)
- Tenzor drugog reda je matrica (dvodimenzionalni niz brojeva)
- Tenzori viših redova

U okviru *TensorFlow*-a, Google-ove biblioteke otvorenog koda za mašinsko učenje, tenzori su osnovni elementi. Svi podaci koji prolaze kroz TensorFlow su predstavljeni kao tenzori. Ova biblioteka koristi tenzore kao standardne jedinice podataka, omogućavajući visoku efikasnost i fleksibilnost prilikom izrade i treniranja modela mašinskog učenja. Tenzori u *TensorFlow*-u u sebi ne sadrže samo podatke (vrednosti), već i informacije o tipu i obliku podataka (dimenzijama). Ovo omogućava

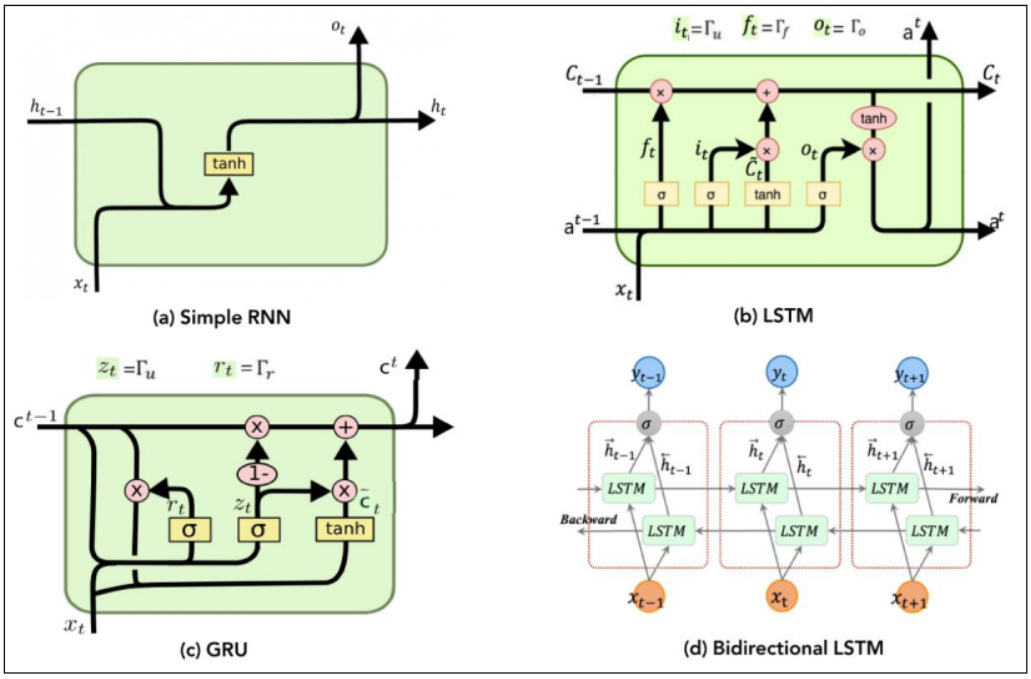
precizno definisanje operacija nad podacima i efikasno izvršavanje na različitim fizičkim platformama.

Grafički, model iz primera na slici Slika 34, sa nazivima i tipovima slojeva, kao i sa šematski predstavljenim tenzorima (kao vezama između slojeva), ilustrovan je na slici Slika 35.



Slika 35. Grafički prikaz modela neuronske mreže

Primer korišćenja sekvencijalnog modela može uključivati različite vrste slojeva, u zavisnosti od specifičnih potreba projekta i vrste podataka. Kroz ovakav pristup, sekvencijalni model omogućava efikasnu i sistematičnu konstrukciju modela, čineći proces eksperimentisanja bržim i lakšim (Natarajan B., 2022). Neki od slojeva koji najviše odgovaraju tipu i obimu podataka u bazi slova srpskog znakovnog jezika su prikazani na slici Slika 36.



Slika 36. Struktura modela: Simple RNN, LSTM, GRU, Bi-LSTM (Apaydin et al., 2020)

Na slici Slika 36 su prikazane arhitekture četiri različita tipa rekurentnih neuronskih mreža koji su korišćeni u eksperimentu koji je deo ovog naučnog rada. Sva četiri tipa mreža spadaju u rekurentne neuronske mreže (*Recurrent Neural Networks - RNN*) (Shiri F. M. et al., 2023).

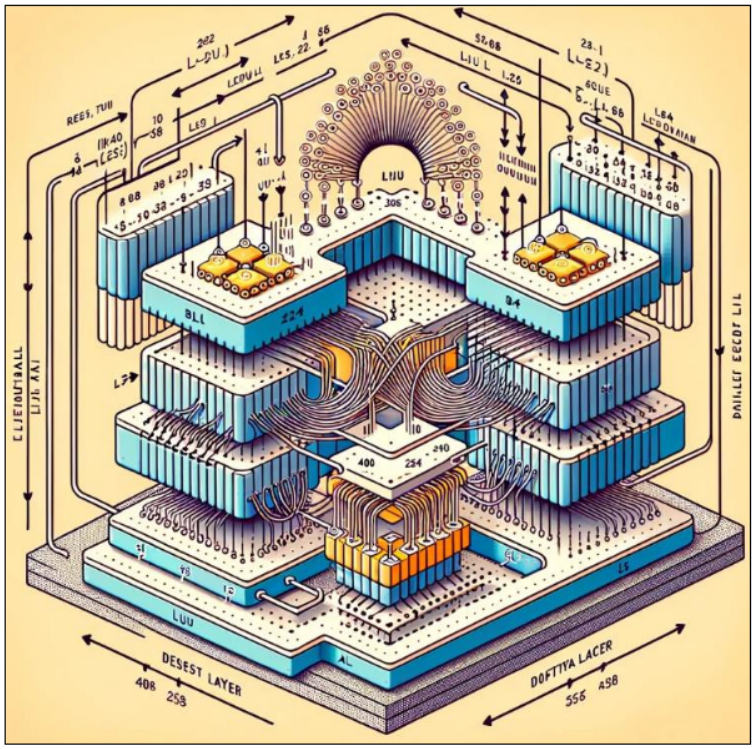
„*Simple RNN*“ je osnovni oblik RNN-a koji koristi `*tanh*` aktivacionu funkciju za obradu ulaznih podataka x_t i prethodnog skrivenog stanja h_{t-1} da bi dobio izlaz o_t i novo skriveno stanje h_t. Primarni problem ove arhitekture je što se teško nosi s dugoročnim zavisnostima zbog pojave problema nestajućeg gradijenta. Gradijenti predstavljaju vektore koji pokazuju pravac u kom treba ažurirati težinske koeficijente modela, kako bi se minimizovala funkcija gubitka, što doprinosi poboljšanju opšte funkcije modela.

„*LSTM*“ (dugoročna kratkoročna memorija) je složeniji oblik RNN-a, koji ima mehanizme vrata (*gates*) - zaboravljanja f_t, ulaza i_t, i izlaza o_t, kao i ćelijsko stanje C_t, koje omogućava mreži da bolje zadrži dugoročne informacije i efikasnije rešava problem nestajućeg gradijenta (Yang S., 2020).

„*GRU*“ (*Gated Recurrent Unit* – jedinica sa kontrolisanim protokom) je slična kao LSTM, ali sa pojednostavljenom strukturom. GRU koristi izmenljiva z_t i resetujuća vrata r_t za kontrolu protoka informacija, što ga čini efikasnim u mnogim zadacima, pri čemu je računski manje zahtevan od LSTM-a.

„*Bidirectional LSTM*“ (dvosmerni LSTM) predstavlja pod-varijantu LSTM-a, koja prolazi kroz podatke u oba smera - i "napred" i "nazad". To modelu omogućava da ima u vidu informacije iz prošlih i budućih tačaka sekvence. Ovo često rezultira poboljšanim performansama na zadacima gde kontekst sa obe strane trenutne tačke podataka može pružiti korisne informacije za predviđanje rezultata.

„*Dense*“ sloj je poznat i kao „potpuno povezani sloj“ i ne pripada grupi RNN (rekurentnih slojeva). Oni imaju karakteristiku da je svaki neuron u nekom sloju povezan sa svim neuronima iz prethodnog sloja. Za razliku od rekurentnih neuronskih mreža (RNN, LSTM, GRU), koje imaju povratne veze i mogućnost obrađivanja sekvencijalnih podataka zadržavajući informacije kroz vreme, gusti slojevi ne poseduju ovu temporalnu komponentu. U kontekstu sekvencijalnih modela, kao što su ovi prikazani na slici Slika 36, Dense sloj se obično koristi na kraju modela nakon rekurentnih slojeva. Rekurentni slojevi (Simple RNN, LSTM, GRU, Bidirekcionalni LSTM) obrađuju sekvencijalne podatke i izlaz svakog koraka (ili samo poslednjeg koraka, zavisno od konfiguracije) i šalju ih dalje niz neuronsku mrežu. Nakon što rekurentni slojevi obave svoju obradu, izlaz se često šalje u jedan ili više gustih slojeva, koji služe za dalju obradu i za donošenje konačne odluke, kao što je klasifikacija ili regresija. Fotografija generisana pomoću alata veštačke inteligencije, sa zadatkom da predstavi mašinski model iz prethodnog primera, prikazana je na slici Slika 37.



Slika 37. Imaginarna ilustracija proizvoljnog modela mašinskog učenja generisana pomoću veštačke inteligencije

Važno je napomenuti da iako sekvencijalni modeli pružaju veliku udobnost i jednostavnost za mnoge tipične zadatke, oni imaju svoja ograničenja, posebno kad je reč o modelima koji zahtevaju složenije arhitekture, kao što su modeli sa višestrukim ulazima i izlazima ili modeli koji zahtevaju slojeve koji dele informacije na nestandardne načine. Za kompleksnije slučajeve, koristi se „funkcionalni model“, koji je takođe deo *Keras* modula.

Nakon projektovanja modela mašinskog učenja, neophodno je kompajlirati, obučiti, testirati i na kraju, sačuvati model. Python programski kod za kompajliranje modela je prikazan na slici Slika 38. **Kompajliranje** predstavlja proces pretvaranja računarskog koda iz ljudima čitljive forme u formu koja se može izvršiti na platformi na kojoj se program pokreće. U procesu kompajliranja, detaljno se konfiguriše model za trening. U okviru funkcije „*model.compile*“, može da se definiše veliki broj parametara, a u kodu prikazanom na slici Slika 38, definisani su optimizator, funkcija za proračun gubitka i merne veličine koju želimo da pratimo (*accuracy* – preciznost). U kontekstu mašinskog učenja i dubokog učenja, "kompajliranje" modela je proces u

kojem konfigurišemo model za trening. U okviru već pomenute biblioteke Keras, funkcija „*model.compile*“ se koristi za definisanje više parametara, kao na primer: takozvanog „*optimizatora*“, funkcije gubitka (*loss function*), merila koje želimo da pratimo, itd.

```
print(“Kompajliranje modela")
model.compile(optimizer='adam',
              loss='sparse_categorical_crossentropy',
              metrics=['accuracy'])
```

Slika 38. Kompajliranje generisanog modela mašinskog učenja

Optimizator korišćen u prikazanom primeru „*optimizer='adam'*“ je jedan od najčešće korišćenih i najpopularnijih. Optimizator 'Adam' je algoritam za optimizaciju stohastičkog gradijenta koji se koristi za ažuriranje težinskih faktora u neuronskoj mreži. Težinski faktori direktno utiču na rad modela i služe za njegovo prilagođavanje podacima i budućoj funkcionalnosti. Adam je popularan zato što kombinuje prednosti dva druga algoritma optimizacije i to „*AdaGrad*“ i „*RMSProp*“. To je razlog zbog čega se smatra efikasnim u različitim uslovima i za različite vrste podataka. Poređenje nekih od najčešće korišćenih optimizatora je prikazano u tabeli Tabela 5.

Tabela 5. Najčešće korišćeni optimizatori za sekvencijalne modele (Chollet, F. et.al, 2015)

Adam (Adaptive Moment Estimation) kombinuje dobre osobine *AdaGrad* i *RMSProp* optimizatora. Koristi procene prvog optimizatora (srednje vrednosti) i drugog, kako bi dinamički prilagodio brzinu učenja za svaki parametar.
AdaGrad (Adaptive Gradient Algorithm) prilagođava stopu učenja za svaki pojedinačni parametar, tako što prilagođava stopu učenja obrnuto proporcionalno sa kvadratnim korenom svih prethodnih kvadrata gradijenata. Pogodan je za sekvencijalne modele gde određene karakteristike mogu imati različite frekvencije pojavljivanja.
RMSProp (Root Mean Square Propagation) je optimizator koji je dizajniran da ubrza SGD u kontekstu optimizacije njegovih funkcija. Ovo omogućava da se brzina učenja (obučavanja) prilagodi za svaki parametar, čineći ga manjim za parametre sa velikim gradijentima i većim za parametre sa malim gradijentima.

SDG (Stochastic Gradient Descent) je vrsta optimizatora koji za svoje funkcionisanje uzima male, nasumično odabrane, podskupove podataka. Obično zahteva manuelno podešavanje stope učenja, što doprinosi ubrzanju konvergencije i smanjenju oscilacija procesa učenja.
Adamax (Adam Max) je varijanta Adam optimizatora, a naziv "*Adamax*" proizilazi iz kombinacije reči "*Adam*" i "*max*". Umesto standardnih parametara, ovaj optimizator koristi maksimum normu, a zbog posebnog pristupa za prilagođavanje stope učenja za svaki parametar, povećava se stabilnost i efikasnost optimizacije u nekim scenarijima.
Nadam (Nesterov-accelerated Adaptive Moment Estimation) kombinuje Adam optimizaciju sa *Nesterovim* momentom, čime se poboljšava opšta performansa modela. *Nesterov* moment ima prednosti u praćenju kretanja gradijenata, što dolazi do bržeg pronalaženja rešenja. Ovo je posebno izraženo u sekvencijalnim modelima gde je dinamika podataka kompleksnija.
Adadelta (Adagrad extension) je algoritam za optimizaciju koji unapređuje neke karakteristike *Adagrad* algoritma. Glavna inovacija ovog algoritma leži u tome što se za prilagođavanje stope učenja koristi samo deo ograničene veličine prethodno korišćenih gradijenata, umesto svih prethodnih gradijenata, kao u *Adagradu*.
AdamW optimizator predstavlja varijantu *Adam* optimizatora, gde "W" označava deo koji se odnosi na "*Weight Decay*", što bi u kontekstu optimizacije težinskih faktora moglo biti protumačeno kao "umanjivanje težinskih faktora". Glavna inovacija *AdamW* optimizatora u odnosu na klasični Adam optimizator je razdvajanje mehanizma za regulaciju težinskih faktora od mehanizma za adaptivne stope učenja. Ovaj pristup sprečava prekomerno prilagođavanje podacima (overfitting), što čini *AdamW* pogodnijim za neke primene u dubokom i mašinskom učenju.

Pored optimizatora koji se najčešće koriste za sekvencijalne modele, kao što su rekurentne neuronske mreže (RNN) i LSTM mreže, Keras modul nudi i druge optimizatore koji su pogodniji za druge namene. Neki od njih su prikazani u tabeli Tabela 6.

Tabela 6. Optimizatori koji su manje pogodni za primenu u sekvencijalnim modelima

FTRL (Follow The Regularized Leader) optimizator za osnovnu ideju ima kombinovanje prednosti Online učenja sa sposobnošću da upravlja velikim i retkim skupovima podataka.
LION optimizator je inovativan pristup u optimizaciji za treniranje dubokih neuronskih mreža. Otkriven je putem simboličke pretrage programa i odlikuje se većom efikasnošću u korišćenju memorije u poređenju sa većinom adaptivnih optimizatora
Loss Scale Optimizer nije klasični optimizator, već samo tehnika optimizacije koja se koristi prilikom treniranja dubokih neuronskih mreža. Za cilj ima da poboljša stabilnost i performanse procesa treniranja, posebno kada se koristi niska preciznost (na primer, 16-bitni brojevi sa pokretnim zarezom) za računanje gradijenata. Ova metoda radi tako što prilagođava "skalu gubitka" (*loss scale*) kako bi se izbegli problemi sa gradijentima, što je česta pojava pri treniranju kompleksnih modela dubokog učenja.

Funkcija gubitka, poznata i kao funkcija troška, ključni je koncept u oblasti mašinskog učenja i veštačke inteligencije. Ona meri razliku između stvarnih izlaznih vrednosti i onih prognoziranih od strane modela. Cilj optimizacije modela jeste smanjivanje funkcije gubitka na što je moguće manju vrednost, što dovodi do povećanja tačnosti modela. U zavisnosti od tipa problema i modela, mogu se koristiti različite funkcije gubitka.

U zavisnosti od tipa problema i modela, mogu se koristiti različite funkcije gubitka. Za regresione probleme se najčešće koristi standardna devijacija (*MSE - Mean Squared Error)*, koja pokazuje koliko u proseku elementi uzorka odstupaju od aritmetičke sredine uzorka. Za probleme klasifikacije se najčešće koristi unakrsna entropija, koja je korišćena i u prikazanom primeru. Funkcija gubitka *loss='sparse_categorical_crossentropy'* se koristi kada podaci sadrže više klasa i kada su nazivi za klase iskazane kao celi brojevi. Ova funkcija gubitka računa unakrsnu entropiju "*crossentropy*" između pravih naziva klasa i prognoziranih, što je standardni način za merenje performansi klasifikacionog modela. Izbor funkcije koja

odgovara tipu podataka je ključan za uspešnost obučavanja modela i njegovoj budućoj funkcionalnosti. Keras modul obezbeđuje 33 različitih funkcija za proračun gubitaka, grupisanih u tri celine.

Metrike su funkcije koje koristimo za ocenjivanje performansi modela mašinskog učenja. U primeru koji je analiziran, korišćena je funkcija „*metrics=['accuracy']*“, gde *'accuracy'* označava metriku za „tačnost“ (Zhou J. Et al., 2021). Ona računa udeo pravilno predviđenih naziva kategorija u odnosu na ukupan broj kategorija. Tačnost je najčešća metrika za klasifikacione probleme i daje intuitivan uvid u to koliko je često model ispravno klasifikovao ulazne podatke. Za potrebe ove procene, koristi se podskup podataka koji je unapred odvojen i uklonjen iz procesa obučavanja. U okviru Keras modela, postoji veliki broj mogućih metrika koji se mogu koristiti u svrhu procenjivanja performansi rada modela (Chollet, F. et.al, 2015):

- accuracy - preciznost
- binary_accuracy – binarna preciznost
- categorical_accuracy – kategorička preciznost
- sparse_categorical_accuracy – retka kategorička tačnost
- top_k_categorical_accuracy – top K kategorička preciznost
- sparse_top_k_categorical_accuracy – retka top K kategorička tačnost
- cosine_proximity – kosinusna sličnost
- clone_metric – kloniranje metrike

Na analiziranom primeru sa slike Slika 38, kada uključimo ove tri komponente u funkciju kompajliranja „*compile*“ modela, model je konfigurisan da koristi „Adam“ optimizator za ažuriranje težinskih faktora tokom obučavanja, da minimizira nivo greške korišćenjem '*sparse_categorical_crossentropy*', kao meru koliko su predviđanja modela tačna u odnosu na tačnu klasifikaciju, kao i da prati tačnost „*accuracy*“ kao glavnu metriku performansi tokom obučavanja.

Obučavanje modela mašinskog učenja, na određenom skupu podataka, podrazumeva korišćenje Python metode „*fit*“. Model je prethodno projektovan i kompajliran, kao u primeru na slici Slika 38. U datom primeru koda, na slici Slika 39,

„*model.fit*" se poziva sa nizom parametara koji definišu kako i koliko dugo će model biti obučavan.

```
    print("Obučavanje modela")
    model.fit(X_train_reshaped, y_train, epochs=30, batch_size=32,
              validation_data=(X_test_reshaped, y_test))
```

Slika 39. Programski kod za obučavanje modela mašinskog učenja

„*X_train_reshaped*" promenljiva predstavlja ulazni skup podataka za obučavanje, koji je prethodno prilagođen kako bi odgovarao očekivanoj ulaznoj formi modela.

„y_train" promenljiva u sebi nosi stvarne (tačne) izlazne vrednosti za prosleđeni skup podataka za obučavanje, koje model pokušava podešavanjem svojih unutrašnjih parametara da predvidi što preciznije.

„*epochs=30*" parametar određuje broj prolaza obučavanja (tzv. epoha). On definiše koliko će se puta proći kroz ceo skup podataka za obučavanje. U prikazanom primeru, model će biti obučavan kroz 30 epoha.

„batch_size=32" parametar određuje veličinu grupe uzoraka koje će se obrađivati zajedno. U ovom slučaju, model će obraditi 32 uzorka odjednom pre nego što ažurira unutrašnje parametre. Veličina grupe ima značajan uticaj na proces obučavanja modela. Veće grupe zahtevaju više računarske radne memorije, stabilnije su pri ažuriranju gradijenata i efikasnije koriste hardver. Korišćenjem manje grupe podataka mogu se poboljšati mogućnosti modela da prepozna i predvidi tačne rezultate za širi spektar podataka. Korišćenje manjih grupa podataka može produžiti vreme obučavanja, ali i ostvariti veću efikasnost i preciznost modela (Xu H., 2020).

„*validation_data=(X_test_reshaped, y_test)*" omogućava evaluaciju modela na posebnom, unapred pripremljenom, podskupu podataka. Ova procena performanse i preciznosti se izvršava nakon svakog ciklusa obučavanja modela (epoha). Ovaj parametar omogućava praćenje rezultata predviđanja modela korišćenjem prethodno neviđenih podataka. Rezultati procene performansi i preciznosti su prikazani na slici Slika 40.

```
194/194 [==============================] - 15s 58ms/step - loss: 2.8957 -
accuracy: 0.1263 - val_loss: 2.5921 - val_accuracy: 0.2005
Epoch 2/30
```

```
194/194 [==============================] - 10s 53ms/step - loss: 2.2684 -
accuracy: 0.2419 - val_loss: 2.0052 - val_accuracy: 0.3295
Epoch 3/30
194/194 [==============================] - 10s 53ms/step - loss: 1.8342 -
accuracy: 0.3735 - val_loss: 1.4956 - val_accuracy: 0.4662

...

Epoch 28/30
194/194 [==============================] - 14s 72ms/step - loss: 0.2740 -
accuracy: 0.9044 - val_loss: 0.3452 - val_accuracy: 0.8730
Epoch 29/30
194/194 [==============================] - 14s 71ms/step - loss: 0.2834 -
accuracy: 0.8971 - val_loss: 0.3737 - val_accuracy: 0.8756
Epoch 30/30
194/194 [==============================] - 14s 71ms/step - loss: 0.2914 -
accuracy: 0.8978 - val_loss: 0.3333 - val_accuracy: 0.8794
```

Slika 40. Procena performansi modela u 30 zadatih prolaza (epoha)

Evaluacija performansi kreiranog i konačno obučenog modela se ostvaruje korišćenjem *Keras* metode „*evaluate*“. Ova procena performansi se izvršava na samom kraju obučavanja modela, korišćenjem podskupa podataka za testiranje. Programski kod je prikazan na slici Slika 41.

```
    print("Evaluacija modela")
    test_loss, test_accuracy = model.evaluate(X_test_reshaped, y_test)
    print(f"Preciznost testa: {test_accuracy}")
```

Slika 41. Programski kod za procenu performansi potpuno obučenog modela

Rezultat procene modela, korišćenjem podskupa za testiranje je prikazan na slici Slika 42. Funkcija prikazuje gubitak i preciznost predviđanja. Što je veća preciznost i manji gubitak, utoliko je obučeni model kvalitetniji, funkcionalniji i precizniji u budućim predviđanjima.

```
Evaluacija modela
49/49 [==============================] - 1s 20ms/step - loss: 0.3333 - accuracy:
0.8794
Preciznost testa: 0.8794326186180115
```

Slika 42. Rezultat evaluacije kreiranog modela iz primera

Čuvanje (snimanje) modela je preporučeno ukoliko je kreirani model zadovoljavajućeg kvaliteta i performansi. Ovako sačuvani model se iz prethodno snimljene datoteke može brzo učitati i koristiti za različite primene. Postupak čuvanja gotovog, obučenog modela je prikazan na slici Slika 43.

```
model_name = "model_3.keras"
print(f"Snimanje modela u toku: {model_name}")
try:
    model.save(model_name)
    print(f"Model: {model_name} je sačuvan.")
except:
    print("Došlo je do greške pri čuvanju modela. Model nije sačuvan!")
```

Slika 43. Snimanje modela za naknadno korišćenje

4.2. Sprovođenje eksperimenta korišćenjem različitih modela mašinskog učenja

Za potrebe disertacije i pronalaženja najboljeg tehničkog rešenja za prepoznavanje gestova srpskog znakovnog jezika, izvršeni su eksperimenti sa različitim eksperimentalnim modelima. Prva grupa podataka koja je korišćena za obučavanje modela je jednodimenzionalna, pri čemu svaki video snimak gesta znakovnog jezika sadrži 10.320 numeričkih vrednosti. Svaka od ovih vrednosti označava jednu koordinatu iz skupa prepoznatih ključnih tačaka na rukama, licu i telu demonstratora. Na slici 44 prikazana je osnovna struktura računarskog koda za generisanje modela mašinskog učenja, dok su u tabelama 7 i 8 prikazane analize tačnosti kada se taj kod koristi.

```
model = keras.models.Sequential([
    {STRUKTURA MODELA}
])
```

Slika 44. Struktura računarskog koda za generisanje modela mašinskog učenja

Tabela 7 se odnosi na modele za koje se koristi jednodimenzionalni set podataka, dok za modele u tabeli Tabela 8 osnovu čini dvodimenzionalni skup podataka. Empirijskom metodom, došlo se do zaključka da se najbolji rezultati većine modela dobijaju sa oko 30 iteracija (*epochs*), pa je taj broj uzet za poređenje osnovnih rezultata preciznosti testa (Javid A.M., 2021).

Tabela 7. Poređenje performansi kreiranih modela za jednodimenzionalni skup podataka

{STRUKTURA MODELA} mašinskog učenja	**Preciznost testa**
`layers.Dense(128, activation='relu', input_shape=(10320,)),` `layers.Dense(64, activation='relu'),` `layers.Dense(30, activation='softmax')`	`0.78143131732940` **cca 78.1%**

layers.Dense(40, activation='relu', input_shape=(10320,)), layers.Dense(64, activation='relu'), layers.Dense(30, activation='softmax')	0.12701483070850 **cca 12.7%**
layers.Dense(258, activation='relu', input_shape=(10320,)), layers.Dense(172, activation='relu'), layers.Dense(114, activation='relu'), layers.Dense(30, activation='softmax')	0.83494520187377 **cca 83.5%**
layers.Dense(512, activation='relu', input_shape=(10320,)), layers.Dropout(0.5), layers.Dense(256, activation='relu'), layers.Dropout(0.5), layers.Dense(128, activation='relu'), layers.Dense(30, activation='softmax')	0.11992263048887 **cca 12%**
layers.Dense(258, activation='relu', input_shape=(10320,)), layers.Dense(172, activation='relu'), layers.Dense(114, activation='relu'), layers.Dense(76, activation='relu'), layers.Dense(30, activation='softmax')	0.84332686662673 **cca 84.3%**
layers.Dense(258, activation='relu', input_shape=(10320,)), layers.BatchNormalization(), layers.Dense(172, activation='relu'), layers.BatchNormalization(), layers.Dense(114, activation='relu'), layers.BatchNormalization(), layers.Dense(76, activation='relu'), layers.BatchNormalization(), layers.Dense(30, activation='softmax')	0.55125725269317 **cca 55.1%**
layers.Conv1D(64, 3, activation='relu', input_shape=(10320, 1)), layers.MaxPooling1D(2), layers.Conv1D(128, 3, activation='relu'), layers.GlobalAveragePooling1D(), layers.Dense(30, activation='softmax')	0.23404255509376 **cca 23.4%**
layers.Dense(128, activation='relu', input_shape=(10320,)), layers.Dense(64, activation='relu'), layers.Dense(30, activation='softmax')	0.80077368021011 **cca 80.1%**
layers.Dense(256, activation='relu', input_shape=(10320,)), layers.Dense(128, activation='relu'), layers.Dense(30, activation='softmax')	0.81108963489532 **cca 81.1%**
layers.Dense(256, activation='relu', input_shape=(10320,)),	0.65183752775192

layers.Dense(128, activation='relu'), layers.Dense(64, activation='relu'), layers.Dense(30, activation='softmax')	**cca 65.2%**
layers.Dense(512, activation='relu', input_shape=(10320,)), layers.Dense(256, activation='relu'), layers.Dense(128, activation='relu'), layers.Dense(64, activation='relu'), layers.Dense(30, activation='softmax')	0.80657643079757 **cca 80.7%**
layers.Dense(258, activation='relu', input_shape=(10320,)), layers.Dense(128, activation='relu'), layers.Dense(64, activation='relu'), layers.Dense(30, activation='softmax')	0.93036752939224 **cca 93%**

Druga grupa podataka je preoblikovana u dvodimenzionalnu matricu, gde su podaci podeljeni u grupe koje predstavljaju pojedinačne slike koje zajedno čine video snimak. Ukupan broj od 10.320 podataka je predstavljen u formi od 258 vrednosti u po 40 grupa. Rezultati modela kreiranih na dvodimenzionalnom skupu podataka su prikazani u tabeli Tabela 8.

Tabela 8. Poređenje performansi kreiranih modela za dvodimenzionalni skup podataka

{STRUKTURA MODELA} mašinskog učenja	**Preciznost testa**
layers.LSTM(64, activation='relu', input_shape=(40, 258)), layers.Dense(30, activation='softmax')	0.83752417564392 cca 83.7%
layers.LSTM(128, activation='relu', input_shape=(40, 258)), layers.LSTM(64, activation='relu'), layers.Dense(30, activation='softmax')	0.79342342343444 cca 79.3%
layers.LSTM(128, activation='relu', input_shape=(40, 258)), layers.Dense(64, activation='relu'), layers.Dense(30, activation='softmax')	0.05544809624552 cca 5.5%
layers.LSTM(256, activation='relu', input_shape=(40, 258)), layers.Dense(128, activation='relu'), layers.Dense(64, activation='relu'), layers.Dense(30, activation='softmax')	0.05544809624552 cca 5.5%
layers.LSTM(100, activation='relu', input_shape=(40, 258), return_sequences=True), layers.LSTM(64, activation='relu'), layers.Dense(30, activation='softmax')	0.77111542224884 cca 77.1%
layers.LSTM(128, activation='relu', input_shape=(40, 258),	0.90457767248153

return_sequences=True), layers.LSTM(64, activation='relu'), layers.Dense(30, activation='softmax')	cca 90.5%
layers.LSTM(258, activation='relu', input_shape=(40, 258), return_sequences=True), layers.LSTM(120, activation='relu'), layers.Dense(60, activation='relu'), layers.Dense(30, activation='softmax')	0.28820115327835 cca 28.8%
layers.LSTM(258, activation='relu', input_shape=(40, 258), return_sequences=True), layers.LSTM(120, activation='relu'), layers.Dense(80, activation='relu'), layers.Dense(40, activation='softmax'), layers.Dense(30, activation='softmax')	0.05093488097190 cca 50.9%
layers.SimpleRNN(64, activation='relu', input_shape=(40, 258)), layers.Dense(30, activation='softmax')	0.59123146533966 cca 59.1%
layers.SimpleRNN(128, activation='relu', input_shape=(40, 258)), layers.Dense(64, activation='relu'), layers.Dense(30, activation='softmax')	0.58929723501205 cca 58.9%
layers.SimpleRNN(256, activation='relu', input_shape=(40, 258)), layers.Dense(128, activation='relu'), layers.Dense(64, activation='relu'), layers.Dense(30, activation='softmax')	0.53578335046768 cca 53.6%
layers.GRU(64, activation='relu', input_shape=(40, 258)), layers.Dense(30, activation='softmax')	0.87105095386505 cca 87.1%
layers.GRU(128, activation='relu', input_shape=(40, 258)), layers.Dense(64, activation='relu'), layers.Dense(30, activation='softmax')	0.88201159238815 cca 88.2%

Poređenjem rezultata rada više različitih eksperimentalnih modela mašinskog učenja koji koriste jednodimenzionalne i dvodimenzionalne skupove podataka za dalju analizu su odabrana najbolja dva. Ti modeli imaju najveću verovatnoću preciznog prognoziranja rezultata.

4.3. Analiza dva reprezentativna modela zasnovana na jednodimenzionalnim i dvodimenzionalnim podacima

Sprovođenjem eksperimentalnih analiza nad različitim modelima, korišćenjem iste baze podataka, preoblikovane u jednodimenzionalni i dvodimenzionalni skup podataka, odabran je model koji daje najpreciznije rezultate predviđanja predstavljene u tabeli Tabela 8. Ova analiza se bavi detaljnim pregledom performansi modela mašinskog učenja tokom procesa obučavanja, koji je obuhvatao 100 iteracija (tzv. epoha). Svaka iteracija predstavlja jedan prolaz kroz celokupan skup podataka koji je pripremljen i odvojen za obučavanje modela. Analiziraju se sledeći parametri:

Parametri koji se proračunavaju na osnovu skupa podataka koji su odvojeni **za obučavanje modela**:

- Parametar „***loss***“ (gubitak) pokazuje koliko je predviđanje modela udaljeno od stvarnih vrednosti. Manji gubitak ukazuje na bolju preciznost modela. Što je manji gubitak, to je kvalitetniji model (Wang Q., 2020).
- Preciznost modela je označena veličinom „***accuracy***“ (preciznost), koja pokazuje koliko često su predviđanja modela tačna. Viša preciznost ukazuje na bolje performanse modela. Najviši cilj je kreiranje modela koji pokazuje najvišu preciznost u radu (0 je minimum, 1 je maksimum), što se najbolje može pratiti ovom veličinom. Ova preciznost se proračunava na bazi podataka koji su korišćeni za samo obučavanje modela i po pravilu je viša od reprezentativnije preciznosti (koja se zasniva na podacima odvojenim za testiranje modela) (Gupta S., 2022).

Parametri koji se proračunavaju na osnovu skupa podataka koji su odvojeni **za proveru rada** modela (skup za validaciju modela):

- Gubitak na skupu podataka za proveru rada modela se označava promenljivom „***val_loss***” (gubitak 2). Slično kao i gubitak, ova veličina se meri na setu podataka koji model nije video tokom obučavanja (tzv. skup za validaciju). Pored ostalih, ova promenljiva takođe bitno pomaže u proceni kako će model raditi na potpuno novim podacima. Na dijagramima koji prikazuju promenu vrednosti tokom kreiranja modela, ova promenljiva je označena sa „gubitak 2“.
- Preciznost izračunata na skupu podataka za validaciju (proveru) rada modela se izražava pomoću promenljive „***val_accuracy***“ (preciznost 2). Ona pokazuje koliko je model tačan kada se koristi na skupu podataka odvojenom za testiranje modela. Ova veličina je u tabelama Tabela 7 i Tabela 8

korišćena za inicijalnu procenu rada modela. Na dijagramima koji prikazuju promenu vrednosti tokom kreiranja modela, ova promenljiva je označena sa „preciznost 2“.

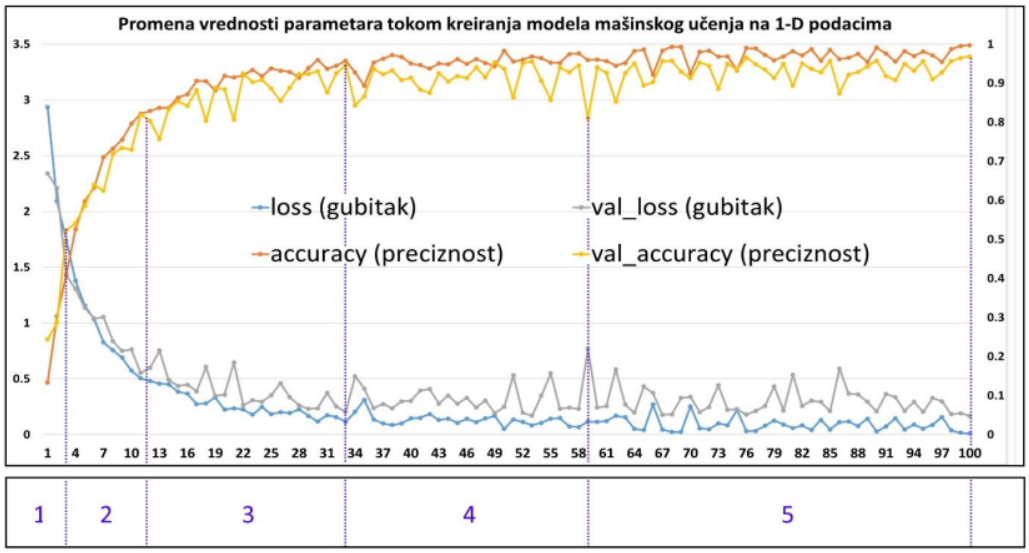
4.3.1. Analiza parametara pri obučavanju modela koji se bazira na jednodimenzionalnima podacima

Model koji proizvodi najbolje rezultate predviđanja, a generisan je na osnovu jednodimenzionalnog skupa podataka je prikazan na slici Slika 45:

```
layers.Dense(258, activation='relu', input_shape=(10320,)),
layers.Dense(128, activation='relu'),
layers.Dense(64, activation='relu'),
layers.Dense(30, activation='softmax')
```

Slika 45. Najpreciznji model zasnovan na jednodimenzionalnim podacima 1-D (10.320)

Na slici Slika 46 je grafikon koji prikazuje promenu parametara tokom obučavanja modela koji koristi jednodimenzionalne (1-D) podatke.



Slika 46. Promena vrednosti parametara tokom kreiranja 1-D modela mašinskog učenja

Za potrebe analize toka obučavanja eksperimentalnog modela mašinskog učenja, grafikon prikazan na slici Slika 46 je podeljen u pet segmenata. Iz dobijenih

rezultata, koji prikazuju tačnost i gubitke na skupovima podataka za obučavanje (*loss, accuracy*) i validaciju (*val_loss, val_accuracy*) imaju sličan trend, ali su razlike među njima izuzetno važan pokazatelj pravog stanja i sposobnosti modela da radi sa nepoznatim podacima.

I 1. Segment I - Predstavlja početnu fazu obučavanja modela. Model se brzo unapređuje, što se vidi iz oštrog pada u gubitku i značajnog povećanja preciznosti predviđanja rezultata. U ovoj fazi, model „uči“ osnovne obrasce podataka.

I 2. Segment I - Model nastavlja da se brzo poboljšava, ali su za nijansu manja od onih u prvom segmentu. Preciznost postaje veća, gubici su sve manji, što znači da model počinje da uopštava naučene obrasce na podatke koje model nije prethodno „video“ tokom treninga.

I 3. Segment I - U ovoj fazi je primetno malo, ali stabilno unapređenje rada modela. Preciznost blago raste, dok se gubici smanjuju. Ovakvo ponašanje može ukazivati na početak konvergencije, gde dodatne iteracije (epohe) u obučavanju donose sve manje poboljšanje.

I 4. Segment I - U ovoj zoni, primetan je nagli i značajan skok nivoa greške i nagli pad preciznosti modela. Na grafikonu se primećuje i mogućnost početka prevelikog prilagođavanja modela obezbeđenom skupu podataka. Ukoliko preciznost na validacionom skupu ne prati opštu preciznost modela, vrlo je izvesno da je došlo do „prevelikog prilagođavanja“.

I 5. Segment I - Model pokazuje zrelost u učenju, sa veoma malim promenama u gubitku i blagim povećanjem preciznosti. Velika je verovatnoća da je model blizu svoje najbolje moguće performanse na trenutno obezbeđenom skupu podataka. Neke iteracije pokazuju neočekivane skokove u gubitku ili padu preciznosti, što označava *overfitting* (preterano prilagođavanje). Na taj način, model gubi sposobnost „generalizacije“, tj. rada sa novim podacima. U toku ove faze, ili čak i u toku prethodne, potrebno je i moguće razmotriti uvođenje tehnika koje bi prekinule obučavanje modela ili uvela druga fina podešavanja. Cilj je da konačno izrađeni model zadrži funkciju uopštenog rada na neviđenim podacima.

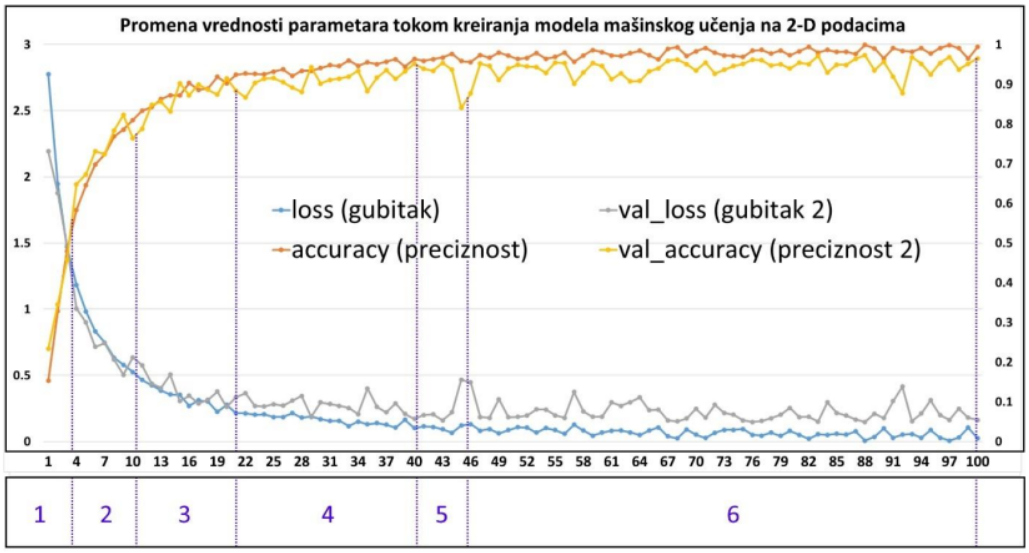
4.3.2. Analiza parametara pri obučavanju modela koji se bazira na jednodimenzionalnima podacima

Model koji generiše najbolje rezultate predviđanja, a baziran je na osnovu dvodimenzionalnog (2-D) skupa podataka je prikazan na slici Slika 47:

```
layers.GRU(128, activation='relu', input_shape=(40, 258)),
layers.Dense(64, activation='relu'),
layers.Dense(30, activation='softmax')
```

Slika 47. Najprecizniji model zasnovan na dvodimenzionalnim podacima 2-D (258×40)

Na slici Slika 48 je grafikon koji prikazuje promenu parametara tokom obučavanja modela koji koristi dvodimenzionalne podatke. Važan pokazatelj performansi modela predstavljaju razlike između rezultata generisanih na podskupovima za obučavanje i za validaciju (testiranje).



Slika 48. Promena vrednosti parametara tokom kreiranja 2-D modela mašinskog učenja

I 1. Segment I - Model započinje s relativno visokim gubitkom i niskom preciznošću, što je uobičajeno na početku obučavanja. Odmah se primećuje značajno poboljšanje na oba skupa podataka, što pokazuje da model dobro uči iz oba skupa obezbeđenih podataka (i za validaciju i za obučavanje).

| 2. Segment | - Kako obučavanje napreduje, uočljivo je kontinuirano poboljšanje u obe metrike - gubitku i preciznosti, kako na skupu za obučavanje tako i na validacionom skupu podataka. Ovo pokazuje da model uspešno uči obrasce iz podataka i da je sposoban da generalizuje ono što je naučio na novim podacima. „Generalizacija“ označava mogućnost da kvalitetno, precizno i sa malim gubicima radi sa podacima koje nikada pre nije „video“.

| 3. Segment | - Uočava se značajan skok u preciznosti na oba skupa podataka, što može ukazivati na to da je model „otkrio“ neke ključne karakteristike u podacima koje mu omogućavaju efikasnije predviđanje rezultata.

| 4. Segment | - Model nastavlja da postiže napredak, sporije i sa manjim stepenom poboljšanja. To može ukazivati na početak „konvergencije“, gde dodatne iteracije u procesu obučavanja donose manje dobitke u performansama.

| 5. Segment | - Primećuje se iznenadni porast gubitka i pad preciznosti na validacionom setu, što može ukazivati na tzv. „preveliko prilagođavanje“ (*overfitting*). Najverovatnije model previše uči specifičnosti trening seta, umesto da uopštava obrasce rada i parametre.

| 6. Segment | - Model završava sa visokom preciznošću i relativno niskim gubitkom, što ukazuje na uspešno treniranje.

Iz prikazanog dijagrama, tokom 100 iteracija obučavanja (*epoha*) i analize promena parametara u toku obučavanja, zaključuje se da je optimalan broj iteracija za obučavanje ovog modela oko 30, maksimalan oko 40, a da se ne ostvaruje nikakav benefit sa iteracijama koje prelaze ovaj broj. Prevelikim brojem iteracija (epoha), model se previše prilagođava dostupnim podacima, pa se opšta sposobnost modela da radi sa nepoznatim podacima značajno umanjuje. Rezultati ukazuju da ovaj model ima značajan potencijal predviđanja rezultata i da je adekvatno prilagođen problemu koji rešava.

5. Praktična primena kreiranog modela mašinskog učenja

Izbor modela za korišćenje u eksperimentu sveden je na samo dva najbolja – jedan je kreiran na osnovu jednodimenzionalnih podataka, a drugi na bazi dvodimenzionalnih podataka. Poređenjem odabranih performansi oba modela, prikazan je postupak odabira samo jednog modela na kome će se razvijati praktično i primenljivo softversko rešenje za prepoznavanje gestova slova srpskog znakovnog jezika.

5.1. Izbor modela za sprovođenje eksperimenta

Da bismo odredili koji će model biti korišćen u eksperimentu (jednodimenzionalni ili dvodimenzionalni), odlučeno je da se uporede performanse oba modela na osnovu dva ključna pokazatelja:

1. gubitka na validacionom setu (*val_loss*) i
2. tačnosti na validacionom setu (*val_accuracy*).

Manji gubitak i veća tačnost su najbolji pokazatelji bolje performanse modela. Na temelju ovih podataka, izračunava se:

1. prosečni gubitak (Prosečna vrednost *val_loss*) i
2. prosečna tačnost (Prosečna vrednost *val_accuracy*),

za oba modela. Razmatranjem srednje vrednosti 1-D i 2-D validacionog gubitka (*val_loss*) i srednje vrednosti tačnosti (*val_accuracy*) kroz sve iteracije (epohe) obučavanja modela. Nakon toga, razmatraju se i drugi faktori, kao što su varijabilnost performansi i tendencije poboljšanja tokom vremena. Takođe, važno je razmotriti kontekst u kojem su modeli trenirani, kao i složenost modela, jer složeniji modeli mogu imati veći kapacitet za učenje ali i veći rizik od prevelikog prilagođavanja (*overfitting*).

Izračunavanje prosečnih vrednosti za „validacionu“ preciznost i gubitak, prikazano je u formulama (10) i (11).

$$\text{Prosečna vrednost } val\_loss = \frac{\sum_{i=1}^{n} val\_loss_i}{n} \tag{10}$$

$$\text{Prosečna vrednost } val\_accuracy = \frac{\sum_{i=1}^{n} val\_accuracy_i}{n} \tag{11}$$

gde je:

val_loss$_i$ – vrednost gubitka na validacionom skupu i-te iteracije (epoha)

val_accuracy$_i$ – vrednost preciznosti na validacionom skupu i-te iteracije (epoha)

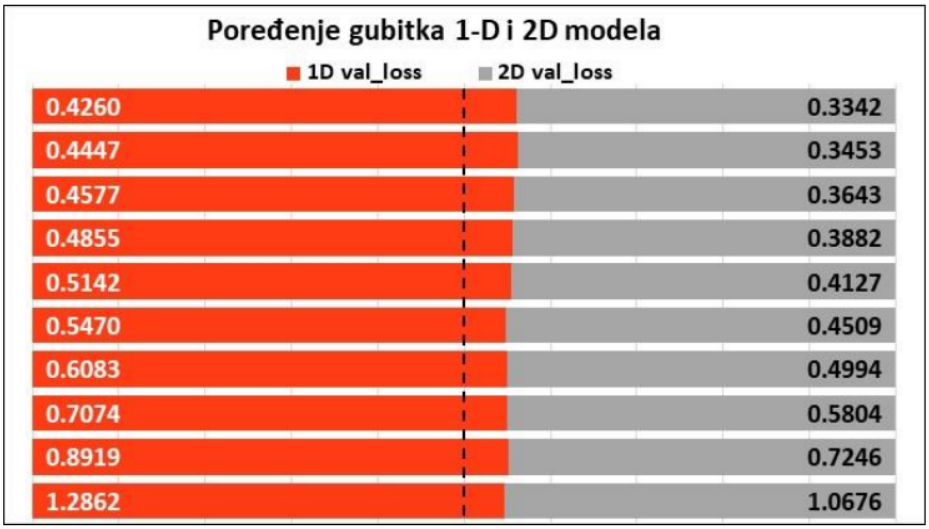
n – ukupan broj iteracija (epoha)

Poređenjem prosečnog gubitka izmerenog na validacionom skupu podataka (*val_loss*) za model izgrađenog na dvodimenzionalnom setu podataka pokazuje primetno bolji rezultat od modela izgrađenog na jednodimenzionalnim podacima (Tabela 9). Poređenje rezultata je prikazano za svakih 10 iteracija (epoha), a u proseku, na uzorku od 100 iteracija, su greške pri primeni dvodimenzionalnog modela manje za 9.17%.

Tabela 9. Numeričko poređenje gubitaka dva modela

Iteracija	1-D val_loss	2-D val_loss	Δ val_loss (1D-2D)	Δ [%]
10	1.2862	1.0676	0.2186	20.48%
20	0.8919	0.7246	0.1673	18.76%
30	0.7074	0.5804	0.1270	17.96%
40	0.6083	0.4994	0.1088	17.89%
50	0.5470	0.4509	0.0961	17.58%
60	0.5142	0.4127	0.1015	19.74%
70	0.4855	0.3882	0.0974	20.05%
80	0.4577	0.3643	0.0934	20.41%
90	0.4447	0.3453	0.0995	22.36%
100	0.4260	0.3342	0.0917	21.53%

Grafički prikaz poređenja gubitaka iz tabele Tabela 9, predstavljen je na slici Slika 49. Uočljivo je da funkcija gubitaka modela izgrađenom na jednodimenzionalnim podacima u svakoj iteraciji pokazuje veće (lošije) vrednosti.



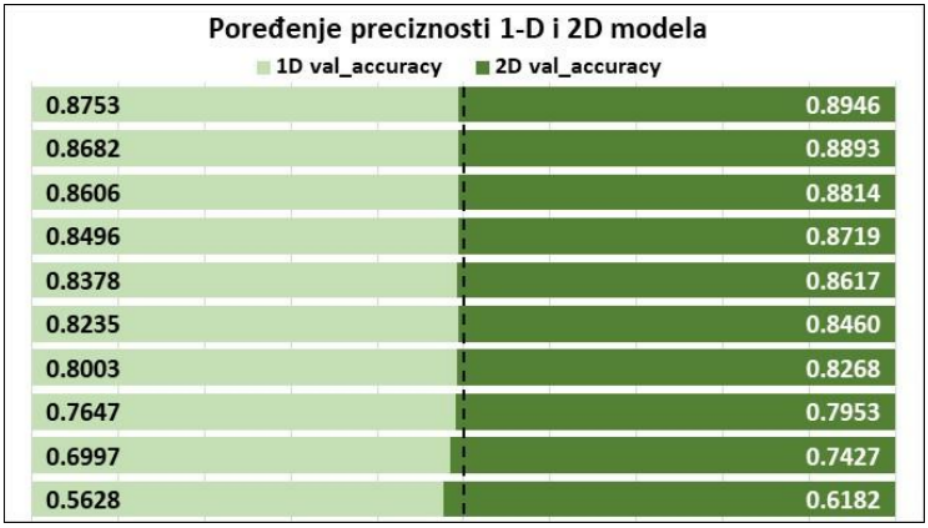
Slika 49. Poređenje funkcije gubitka za 1-D i 2-D model

Prosečna tačnost na validacionom setu (val_accuracy) je veća za 2-D model sa 89.46% u poređenju sa 87.53% za 1-D model. Tačnost je u proseku veća za 1.93% u korist modela kreiranog nad dvodimenzionalnim podacima (Tabela 10).

Tabela 10. Numeričko poređenje preciznosti dva modela

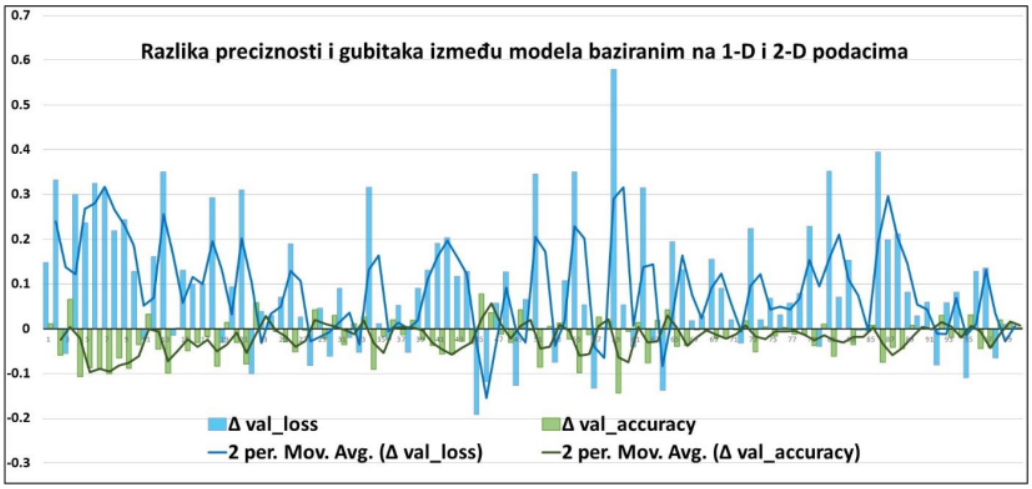
Iteracija	1-D val_accuracy	2-D val_accuracy	Δ val_accuracy	Δ [%]
10	0.5628	0.6182	-0.0554	5.54%
20	0.6997	0.7427	-0.0431	4.31%
30	0.7647	0.7953	-0.0306	3.06%
40	0.8003	0.8268	-0.0265	2.65%
50	0.8235	0.8460	-0.0225	2.25%
60	0.8378	0.8617	-0.0239	2.39%
70	0.8496	0.8719	-0.0223	2.23%
80	0.8606	0.8814	-0.0207	2.07%
90	0.8682	0.8893	-0.0211	2.11%
100	0.8753	0.8946	-0.0193	1.93%

Na osnovu prikazanih rezultata, 2-D model ima neznatno bolju prosečnu tačnost, dok je gubitak značajno manji. Krajnji izbor modela pored tačnosti i greške, zavisi i od specifičnih ciljeva i uslova primene, a po svim parametrima, 2-D model pokazuje bolju prilagođenost bazi podataka srpskog znakovnog jezika. Grafički prikaz vrednosti iz tabele Tabela 10 je predstavljen na slici Slika 50. Primetno je da su neznatno bolje vrednosti na strani modela zasnovanog na 1-D podskupu podataka.



Slika 50. Poređenje funkcije preciznosti za 1-D i 2-D model

Razlike u preciznosti i gubicima između 1-D i 2-D modela su grafički prikazane na slici Slika 51. Primetan je veći stepen grešaka 1-D modela (plave, pozitivne vrednosti), a takođe i manji stepen tačnosti u odnosu na 2-D model (zelene, negativne vrednosti).

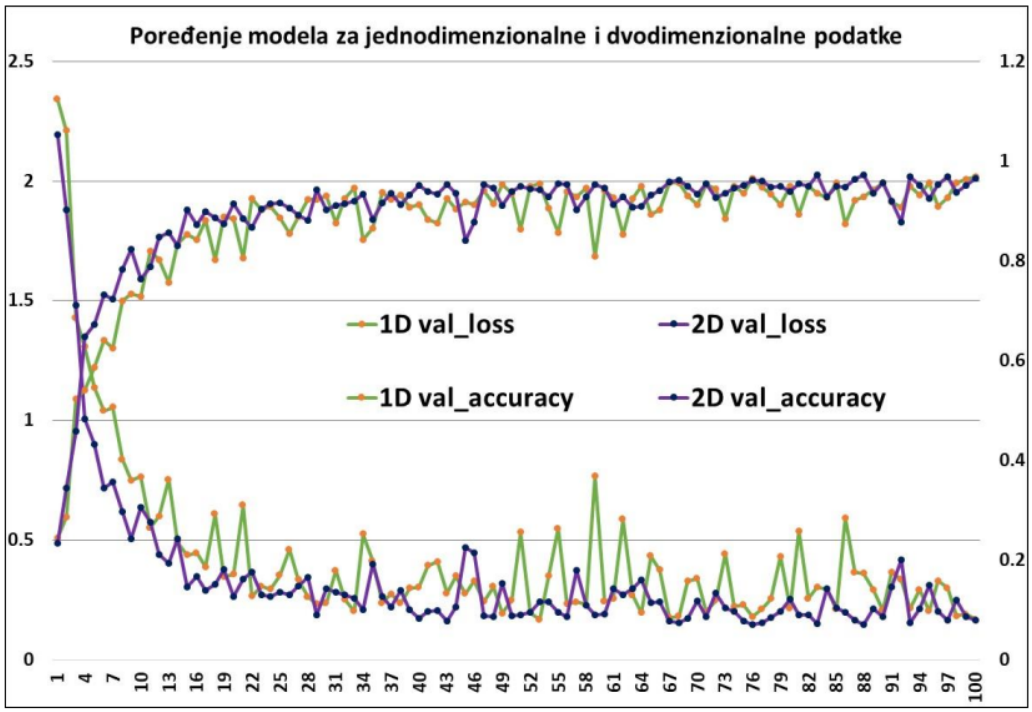


Slika 51. Poređenje preciznosti i grešaka između modela zasnovanih na jednodimenzionalnim i dvodimenzionalnim podacima

Pored analize dva najvažnija parametra (preciznosti modela i gubitaka), pri izboru najboljeg modela, važno je uzeti u obzir i promenu performansi modela tokom

vremena. Ako neki model pokazuje znakove prevelikog prilagođavanja (*overfitting*) u kasnijim iteracijama (epohama), ili ako se performanse modela stabilizuju, to su važni faktori koji bi takođe trebali biti razmatrani prilikom donošenja konačne odluke.

Na slici Slika 52, grafički je prikazano poređenje preciznosti i gubitaka za modele koji se porede. Model kreiran nad dvodimenzionalnim podacima (označen plavom bojom), u manjem broju iteracija dostiže višu preciznost. U isto vreme, gubici koje model generiše su manji od jednodimenzionalnog modela. Važnu razliku predstavlja i veća prikazana konzistentnost u rezultatima, što potvrđuje da je model zasnovan na dvodimenzionalnim podacima bolji izbor za dalje eksperimentisanje.



Slika 52. Poređenje tačnosti i gubitaka modela za 1-D i 2-D podatke

Uzimanjem u obzir svih analiziranih parametara i performansi dva modela, blagu prednost u izboru je ostvario 2-D model (neznatno viša preciznost, a značajno manji gubici). Iz tog razloga, za dalje sprovođenje eksperimenata i primenu u praksi je **odabran model mašinskog učenja zasnovan na dvodimenzionalnim podacima**.

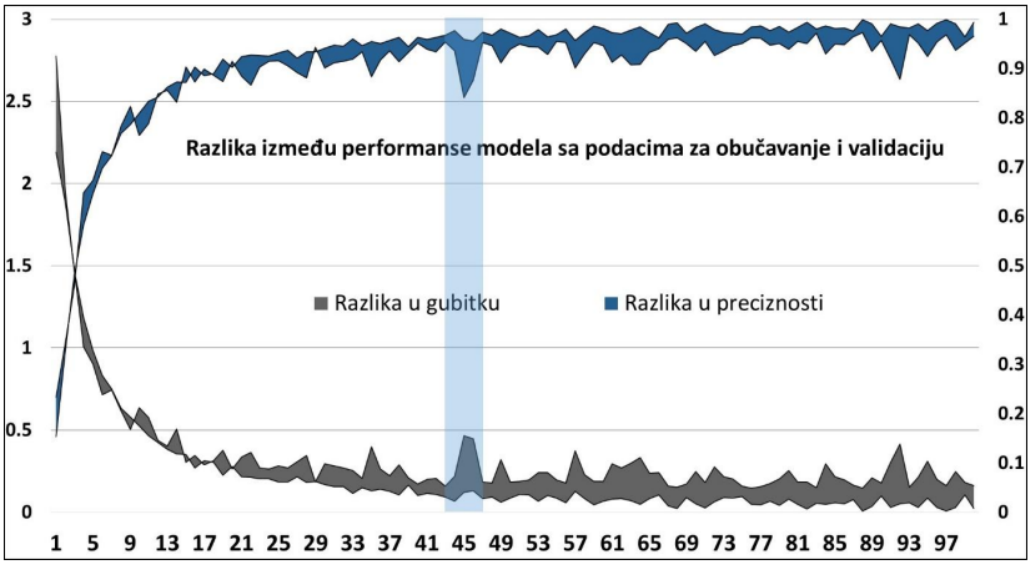
5.2. Izbor parametara za obučavanje odabranog dvodimenzionalnog modela mašinskog učenja

Za proces obučavanja modela, dva parametra koja definišu pokretanje procesa su epochs (broj iteracija) i batch_size (veličina paketa podataka koji se odjednom prosleđuje modelu). Navedeni parametri su direktno povezani sa brzinom i efikasnošću učenja, kao i sa sposobnošću generalizacije obučenog modela. Izmenom i analizom uticaja ovih parametara na rezultate obučavanja se može doći do boljeg razumevanja kako različiti uslovi obučavanja utiču na performanse modela.

5.2.1. Broj iteracija (epochs)

Parametar „epochs“ definiše ukupan broj kompletnih prolazaka kroz celokupan skup podataka za obučavanje koji će se izvršiti tokom samog procesa. Jedna „epoha“ tj. prolaz, podrazumeva da je svaki uzorak u skupu podataka prosleđenom modelu za obučavanje korišćen tačno jednom za ažuriranje težinskih faktora modela.

Obzirom na rezultate odabranog dvodimenzionalnog modela, prikazane na slici Slika 53, može se izvesti zaključak da je optimalan broj iteracija (epochs) za njegovo obučavanje oko 40. Primetan je skok gubitaka i pad preciznosti u koraku 43, što pokazuje da model najverovatnije ulazi u fazu prevelike prilagođenosti korišćenoj bazi podataka. Do koraka 40, model je ostvario preciznost od **96.36%** korišćenjem podataka za obuku, odnosno **95.16%** sa podskupom podataka odvojenim za validaciju (testiranje).



Slika 53. Razlike u funkcijama gubitka i preciznosti modela zasnovanog na dvodimenzionalnom skupu podataka

Za potrebe eksperimenta i nastavka kreiranja funkcionalnog modela za prepoznavanje gestova srpskog znakovnog jezika, odlučeno je da za konačno obučavanje modela bude korišćeno **40 prolaza** (epochs).

5.2.2. Veličina paketa podataka (batch_size)

Parametar „batch_size“ definiše broj uzoraka koji će se koristiti u jednoj iteraciji obučavanja modela. Veličina serije može varirati u zavisnosti od problema i skupa podataka, a optimalan *batch_size* se često određuje eksperimentisanjem i prilagođavanjem prema konkretnim potrebama i ograničenjima.

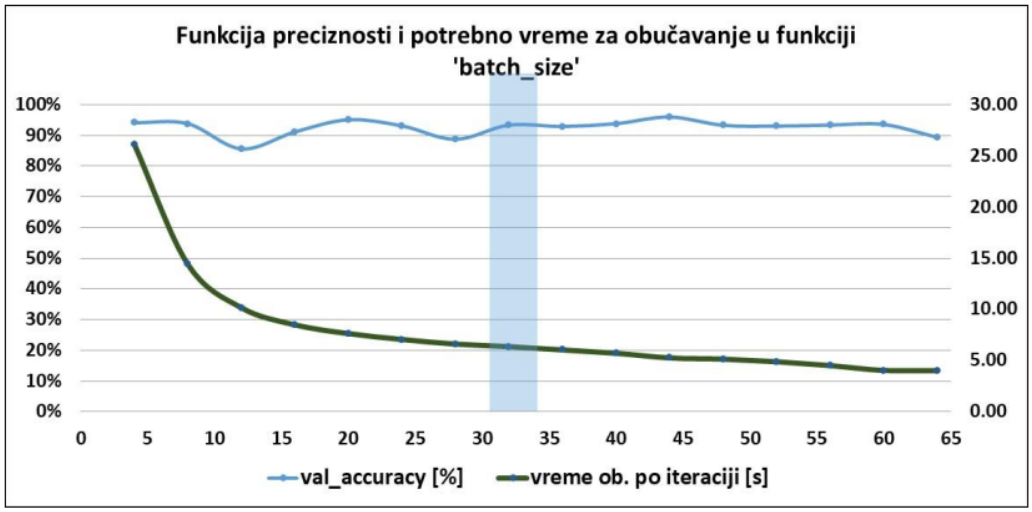
Analiziranjem podataka iz tabele Tabela 11Tabela 1, kao i sa grafikona prikazanim na slikama Slika 54 i Slika 55, dolazi se do zaključka da je optimalna veličina za parametar ***batch_size* = 32**.

Tabela 11. Promene rezultata obučavanja modela u zavisnosti od veličine „*batch_size*“

Parametri obučavanja modela: 40 iteracija (epochs) – 2D model: (40, 258)							
batch_size	**accuracy**	**loss**	**val_loss**	**val_accuracy**	**val_accuracy [%]**	**∑ vreme obučavanja [s]**	**vreme ob. po iteraciji [s]**
4	0.1664	0.9444	0.2130	0.9407	94.1%	1045	26.13
8	0.1331	0.954	0.2254	0.9375	93.8%	576	14.40

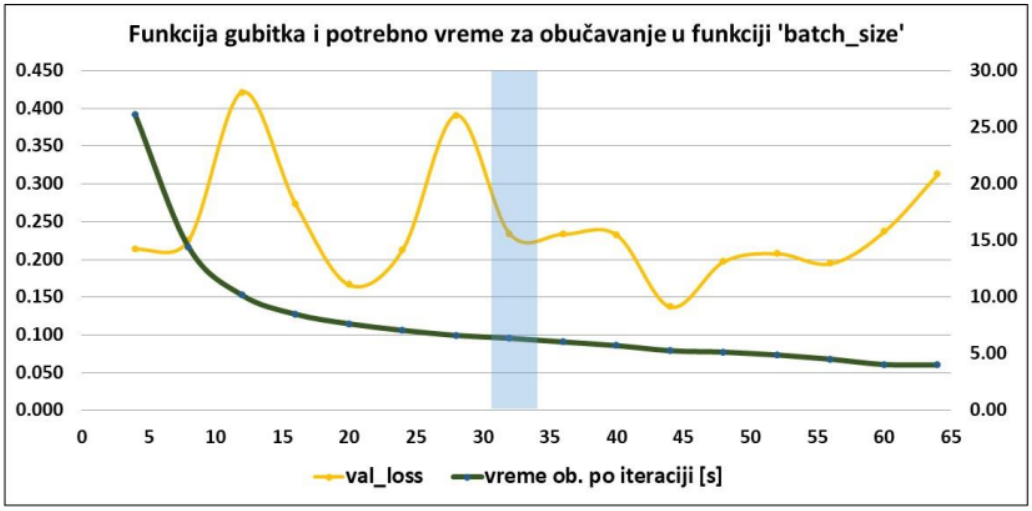
		2					
12	0.1048	0.9652	0.4213	0.8549	85.5%	406	10.15
16	0.1183	0.9569	0.2731	0.9104	91.0%	339	8.48
20	0.1242	0.9576	0.1662	0.9504	95.0%	305	7.63
24	0.1117	0.9577	0.2128	0.9304	93.0%	282	7.05
28	0.1209	0.9569	0.3906	0.8865	88.7%	264	6.60
32	0.1235	0.9558	0.2338	0.9329	93.3%	254	6.35
36	0.1349	0.9537	0.2333	0.9284	92.8%	241	6.03
40	0.1110	0.9623	0.2322	0.9375	93.8%	228	5.70
44	0.0901	0.9710	0.1369	0.9594	95.9%	210	5.25
48	0.1339	0.9561	0.1969	0.9323	93.2%	205	5.13
52	0.1141	0.9616	0.2076	0.9297	93.0%	195	4.88
56	0.1001	0.9660	0.1942	0.9329	93.3%	180	4.50
60	0.1271	0.9524	0.2366	0.9349	93.5%	161	4.03
64	0.1508	0.9487	0.3125	0.8930	89.3%	160	4.00

Funkcija preciznosti modela prikazana na slici Slika 54 oslikava kontinualni blagi porast pri povećanju veličine „batch_size“, dok se u isto vreme, potrebno vreme za obučavanje smanjuje. Zaključuje se da je dobar odnos vremena potrebnog za obučavanje i ostvarene preciznosti postignuto oko vrednosti 32 za batch_size parametar.



Slika 54. Preciznost modela u funkciji *batch_size* parametra

Slično kao i za funkciju preciznosti, funkcija koja pokazuje gubitak ostvaruje optimalan rezultat u procesu obučavanja modela kada je **batch_size vrednost između 30 i 35.** Odnos funkcije gubitka, vremena potrebnog za obučavanje modela u odnosu na batch_size parametar je prikazana na slici Slika 55.



Slika 55. Gubitak modela u funkciji *batch_size* parametra

5.3. Potencijalne implementacije modela mašinskog učenja

Kreirani modeli mašinskog učenja se u zavisnosti od vrste podataka, namene, karakteristika i zahtevnosti mogu primeniti na velikom broju fizičkih uređaja (hardver), kreirati na različitim programskim platformama (softver), mogu se postaviti na različite lokacije i koristiti za različite namene.

5.3.1. Hardver za model mašinskog učenja

Uzimajući u obzir veliku preciznost i nizak stepen gubitka odabranog modela, a bez velikih hardverskih zahteva, model je prilagodljiv za rad na različitim uređajima. Obzirom na korišćenje MediaPipe platforme za prepoznavanje ključnih tačaka na rukama, licu i telu demonstratora, za primenu u realnom vremenu, bez kašnjenja u obradi podataka, za optimalne performanse, preporučene su sledeći uređaji:

- **Laptop** ili **Desktop računar**, sa modernim procesorima (i5, i7, Ryzen 5, Ryzen 7), sa preporučenim minimumom radne memorije od 8GB. Poseban grafički procesor može pomoći bržoj obradi slika i video zapisa, pošto je moguća raspodela zadataka između centralnog i grafičkog procesora.
- **Tablet** ili **Mobilni** telefon srednje ili visoke kategorije, sa kvalitetnim kamerama i velikom procesorskom snagom bi bili sposobni da bez kašnjenja i u realnom vremenu koriste pripremljeni model i prepoznaju gestove srpskog znakovnog jezika.

5.3.2. Softver za model mašinskog učenja

Za razvoj aplikacija na osnovu generisanog *Keras* modela, mogu se koristiti različite softverske platforme, jezici i alati. U zavisnosti od primene, uređaja za koji se program pravi, mogu se koristiti različita rešenja:

- Za **desktop** ili **laptop aplikacije** najčešće se koriste razvojna okruženja kao što su Visual Studio, PyCharm, Eclipse i slično. Sa odgovarajućim dodacima za Python i Keras, aplikacije se mogu praviti korišćenjem postojećih modula za kreiranje grafičkih interfejsa Tkinter ili PyQt.
- Za **mobilne aplikacije**, u zavisnosti od toga da li se prave za Android ili iOS, mogu se koristiti Android Studio ili Xcode respektivno. Pre primene, Keras model se može konvertovati u format kompatibilan sa TensorFlow Lite koji efikasnije radi na mobilnim platformama.

- Za **internet platforme** i druge internet lokacije se model može upotrebiti korišćenjem TensorFlow.js JavaScript biblioteke. Ova biblioteka je specijalno namenjena za korišćenje modela na klijentskoj strani bez potrebe za serverom.

5.3.3. Lokacije za postavljanje modela

U skladu sa svojom namenom, aplikacije koje se kreiraju za unapred generisane modele mašinskog učenja se mogu postaviti i pokretati na različitim lokacijama:

- **Lokalno** pokretanje aplikacija, na uređaju za koji je program namenjen je najčešći i najbolji način za zahtevne sisteme koji zahtevaju rezultate u realnom vremenu. Video igre ili edukativni materijal se najčešće prave sa namerom da se izvršavaju lokalno, na uređaju za koji su pravljene.
- **„Cloud" platforme** (nedefinisane internet lokacije „u oblaku") kao što su Amazon AWS, Google Cloud ili Microsoft Azure nude mogućnost postavljanja na njih modela u obliku API (Application Programming Interface). Postavljanjem modela na internet, omogućeno je centralizovano ažuriranje i održavanje aplikacija za koje je to potrebno.
- **„Edge Computing"** podrazumeva distribuirani računarski model koji približava aplikacije izvorima podataka sa ciljem da se korisnik postavi fizički bliže serveru. Za razliku od podataka koji su na serverima u oblaku, ovi podaci su blizu korisnika, pa su i kašnjenja signala (*latency*) manja a performanse bolje. Primer za ovakvu postavku su IoT uređaji i lokalni serveri. Na uređaje Cisco Webex Video Codec (Kit, Pro, EQ, …), Microsoft Kinect ili druge, postavljanje modela direktno na uređaj, obezbeđuje lokalnu obradu bez kašnjenja koja su prisutna u komunikaciji sa serverima lociranim na internetu.

5.3.4. Moguće aplikativne primene

Modeli mašinskog učenja mogu se primenjivati u širokom spektru aplikacija i programa, prožimajući različite industrije i discipline. Namena projektovanog i kreiranog modela je prepoznavanje gestova srpskog znakovnog jezika, pa se u tom kontekstu, mogu praviti različita softverska rešenja, zasnovanih na tom modelu:

- **Edukativni interaktivni materijali** i aplikacije za **trening i obuku** koji koriste kameru i snimanje gestova korisnika, mogu biti korišćeni za pomoć u učenju srpskog znakovnog jezika za osobe sa oštećenjem sluha.
- **Video igre** za računare ili mobilne uređaje se takođe mogu kreirati na osnovu generisanog modela, sa ciljem da se najmlađoj populaciji približi srpski znakovni jezik i omogući interaktivno učenje uz zabavu.
- **Komunikacione platforme** poput email klijenata, različitih aplikacija za instant poruke ili video konferencijskih sistema može omogućiti automatsko prepoznavanje i prevod gestova srpskog znakovnog jezika u realnom vremenu, olakšavajući komunikaciju sa gluvim ili slušno oštećenim osobama.
- **Korisnička podrška** u koju spadaju chatbot-ovi i drugi virtuelni asistenti može omogućiti interakciju sa korisnicima koji koriste znakovni jezik, čime se povećava inkluzivnost i dostupnost usluga.
- **Interaktivni kiosci i samouslužni terminali** mogu omogućiti korisnicima sa oštećenjem sluha da u tržnim centrima, bankama ili drugim javnim mestima iskažu svoje potrebe korišćenjem znakovnog jezika i dobiju potrebne informacije.
- **Integrativne aplikacije** za postojeća hardverska rešenja kao što su Microsoft Kinect, Cisco Codec, Polycom itd., mogu omogućiti korisnicima komunikaciju sa udaljenim partnerima upotrebom gestova znakovnog jezika. Gestovi se mogu prepoznavati, prevoditi i titlovati za sagovornika.

5.4. Python kod za korišćenje modela mašinskog učenja pri prepoznavanju gestova slova srpskog znakovnog jezika

Za potrebe ove disertacije, kreiran je funkcionalan primer, u Python programskom jeziku, koji koristi prethodno pripremljenu bazu podataka sa dvodimenzionalnim vrednostima gestova srpskog znakovnog jezika. Detalji o korišćenoj Python verziji i pripadajućih uvezenih modula su prikazani u Prilogu A. Aplikacija koja koristi unapred pripremljen model je kreirana (Slika 56):

- Za desktop/laptop računar (preporučenih karakteristika procesora i5, i7 i radne memorije od 8GB, sa web kamerom)
- Korišćenjem Python (3.10 verzije) programskog jezika
- Program je instaliran lokalno, bez potrebe za internet konekcijom
- Namena aplikacije je za demonstraciju i učenje gestova srpskog znakovnog jezika

```
def ucitaj_model(model_name):
    szj_model = load_model(model_name)
    print(f"Model: {model_name} loaded.")
    return szj_model

if __name__ == "__main__":
    odstampaj_zaglavlje()
    szj_model = ucitaj_model("model_3.keras")
    szj_prepoznavanje_gestova(szj_model)
```

Slika 56. Početak programa za prepoznavanje srpskog znakovnog jezika

Kao što je prikazano u izvornom kodu, unapred kreiran *Keras* model („model_3.keras“) se prvo učitava, a nakon toga poziva funkcija za prepoznavanje gestova srpskog znakovnog jezika („szj_prepoznavanje_gestova(szj_model)“). Funkcija za prepoznavanje gestova snimljenih web kamerom računara je prikazana na slici Slika 57.

```
def szj_prepoznavanje_gestova(ssl_model):
    frame_count = 0
    recording_in_progress = False
    ruke_u_kadru_nisu_bile_count = 5
    # Broj frejmova potrebno da prodje BEZ ruku u kadru, da bi se snimao sledeci
    ruke_u_kadru_bile = False
    ruke_u_kadru_pauza = 10 # PAUZA u frame-ovima izmedju snimanja klipova

    # Isečen ulaz sa kamere, kako bi snimljeni video klipovi bili manji
    frame_left_margin = 280
```

```
    frame_right_margin = 1000

    # Detalji za CODEC i snimanje - Dell Latitude 7420 Camera: 1280x720@30fps
    targeted_frame_count = 40  # UKUPAN BROJ FRAME-ova koji se snima
    cap = cv2.VideoCapture(0, cv2.CAP_DSHOW)
    cap.set(cv2.CAP_PROP_FRAME_WIDTH, video_width)
    cap.set(cv2.CAP_PROP_FRAME_HEIGHT, video_height)
    cap.set(cv2.CAP_PROP_FORMAT, -1) # Preuzimanje videa u izvornom (RAW) obliku

    predvidjanje1 = ""
    predvidjanje2 = ""
    predvidjanje3 = ""
    slovo1 = " "
    slovo2 = " "
    slovo3 = " "

    # Postavljanje MediaPipe parametara
    with mp_holistic_model.Holistic(min_detection_confidence=0.5,
                                    min_tracking_confidence=0.5) as holistic:
        while cap.isOpened():
            ret, frame = cap.read()  # VIDEO uživo sa web kamere
            frame = frame[0:video_height, frame_left_margin:frame_right_margin]
                          #crop frame to middle 3/5 of total video width

            # Detektovanje ključnih tačaka
            image, results = mediapipe_keypoint_locations(frame, holistic)

            cv2.rectangle(image, (0,0), (video_width,30), (144,144,188), -1)
            cv2.putText(image, f"{predvidjanje1}% {slovo1}
                        | {predvidjanje2}% {slovo2}
                        | {predvidjanje3}% {slovo3}", (200, 23),
                        cv2.FONT_HERSHEY_SIMPLEX, 0.8, (0,0,0), 2, cv2.LINE_AA)

            # POCETAK SNIMANJA JEDNOG KADRA čim se primeti ruka u kadru
            ruke_u_kadru = (results.right_hand_landmarks) or
                           (results.left_hand_landmarks)
            if ruke_u_kadru:
                ruke_u_kadru_bile = True
                ruke_u_kadru_nisu_bile_count = 0
            else:
                ruke_u_kadru_bile = False
                ruke_u_kadru_nisu_bile_count += 1

            # ČIM SE RUKA DETEKTUJE U KADRU, KREĆE SNIMANJE I PRAVI SE FAJL
            if (ruke_u_kadru) and (not recording_in_progress):
                    print("Detektovana ruka u kadru!")
                    all_keypoints = []
                    frame_count = 0
                    recording_in_progress = True

            # U TOKU SNIMANJA KADRA - Promena status bar-a
            if (recording_in_progress) and (frame_count < targeted_frame_count):
                recording_progress_bar = int(video_height -
                                     frame_count/targeted_frame_count*video_height)
                cv2.putText(image, "Snimanje u toku!", (256, 700),
                            cv2.FONT_HERSHEY_SIMPLEX, 1, (11,11,255),
                            2, cv2.LINE_AA)
                cv2.line(image, (20,video_height-recording_progress_bar),
                         (20,video_height), (0,0,255), 40)
                frame_count += 1
```

```
            # PREPOZNAVANJE ključnih tačaka
            one_frame_keypoints = recognize_keypoints(results)
            all_keypoints = np.append(all_keypoints, one_frame_keypoints)

        # PAUZA na kraju snimanja kadra, da ne krene ako je ruka još u kadru
        elif (frame_count >= targeted_frame_count) and
              (ruke_u_kadru_nisu_bile_count < ruke_u_kadru_pauza):
            cv2.putText(image, "Sklonite ruke iz kadra...", (200, 700),
                        cv2.FONT_HERSHEY_SIMPLEX, 1,
                        (255,255,255), 2, cv2.LINE_AA)

        elif (frame_count >= targeted_frame_count):
            recording_in_progress = False
            frame_count = 0
            print("Prepoznavanje slova uz pomoć modela mašinskog učenja:")
            prepoznaj_ovo = all_keypoints.reshape((1, 40, 258))
            # Prosledjivanje niza za prepoznavanje i dobijanje rezultata
            result = (szj_model.predict(prepoznaj_ovo))

            # Prebacivanje rezultata u jedno-dimenzionalni niz (30,) i
            # rasporedjivanje u promenljive
            numpy_niz_rezultata = result.flatten()
            predvidjanje1 = round(max(numpy_niz_rezultata)*100)
            slovo1 = sign_dict_keys_list[np.argmax(numpy_niz_rezultata)]
            print(f"Verovatnoca: {predvidjanje1}% | Redni broj slova:
                  {np.argmax(numpy_niz_rezultata)}
                  | Prepoznato slovo: {slovo1}")
            numpy_niz_rezultata[np.argmax(numpy_niz_rezultata)] = 0
            predvidjanje2 = round(max(numpy_niz_rezultata)*100)
            slovo2 = sign_dict_keys_list[np.argmax(numpy_niz_rezultata)]
            print(f"Verovatnoca: {predvidjanje2}% | Redni broj slova:
                  {np.argmax(numpy_niz_rezultata)} |
                  Prepoznato slovo: {slovo2}")
            numpy_niz_rezultata[np.argmax(numpy_niz_rezultata)] = 0
            predvidjanje3 = round(max(numpy_niz_rezultata)*100)
            slovo3 = sign_dict_keys_list[np.argmax(numpy_niz_rezultata)]
            print(f"Verovatnoca: {predvidjanje3}% | Redni broj slova:
                  {np.argmax(numpy_niz_rezultata)}
                  | Prepoznato slovo: {slovo3}")

        # Prikaži na ekranu u prozoru sa slikom
        cv2.imshow('Serbian Sign Language | AZBUKA <Esc> to Exit', image)
        # Izlaz iz programa pritiskom na <Esc> taster
        if cv2.waitKey(10) & 0xFF == 27:
            break
    cap.release()
    cv2.destroyAllWindows()
```

Slika 57. Funkcija za prepoznavanje gestova srpskog znakovnog jezika

U prvom delu funkcije postavljaju se parametri važni za korišćenje web kamere i prosleđivanje slika i snimka za dalju obradu („Detalji za CODEC i snimanje“). Pored parametara vezanih za OpenCV (Bradski, G., 2008) Python modul, definišu se i neophodne promenljive koje će se koristiti za prikazivanje rezultata prepoznavanja srpskog znakovnog jezika.

U kodu koji sledi, postavljaju se parametri za funkcije iz MediaPipe modula. Koristi se „holistic“ model, koji omogućava istovremenu detekciju lica, šaka i poza u realnom vremenu. Korišćeni MediaPipe „holistic“ model integriše nekoliko prethodno razvijenih modela, kao što su *MediaPipe Hands* (ruke), *MediaPipe Pose* (telo) i *MediaPipe Face Mesh* (lice) u jedan efikasan funkcionalni proces (*pipeline)*. Korišćenjem samo jednog modula umesto tri, smanjuje se neophodna potrebna procesorska snaga i kompleksnost koja bi bila potrebna za integraciju više modela. Ovaj pristup omogućava aplikacijama da dobiju sveobuhvatne informacije o položaju snimljenog subjekta u vidu položaja i pokreta ruku, lica i tela, a to sve koristeći samo jedan model.

Snimak sa web kamere (označen komentarom „VIDEO uživo sa web kamere“) se prosleđuje funkciji označenoj komentarom „Detektovanje ključnih tačaka“, a ukoliko se u rezultatima (*results*) prepozna prisustvo bar jedne šake, započinje se čuvanje narednih 40 slika u privremenu, radnu memoriju. Vrednosti prepoznatih ključnih tačaka sa zabeleženih 40 snimaka se dalje prosleđuju funkciji unapred generisanog modela za prepoznavanje srpskog znakovnog jezika „*szj_model.predict(prepoznaj_ovo)*“, koja vraća 30 vrednosti sa prognoziranim verovatnoćama (od 0 do 1). Najveća vrednost predstavlja prognozu modela koje je slovo subjekt prikazao ispred web kamere. Program prikazuje tri najveće vrednosti, sa procentima verovatnoće koju model generiše.

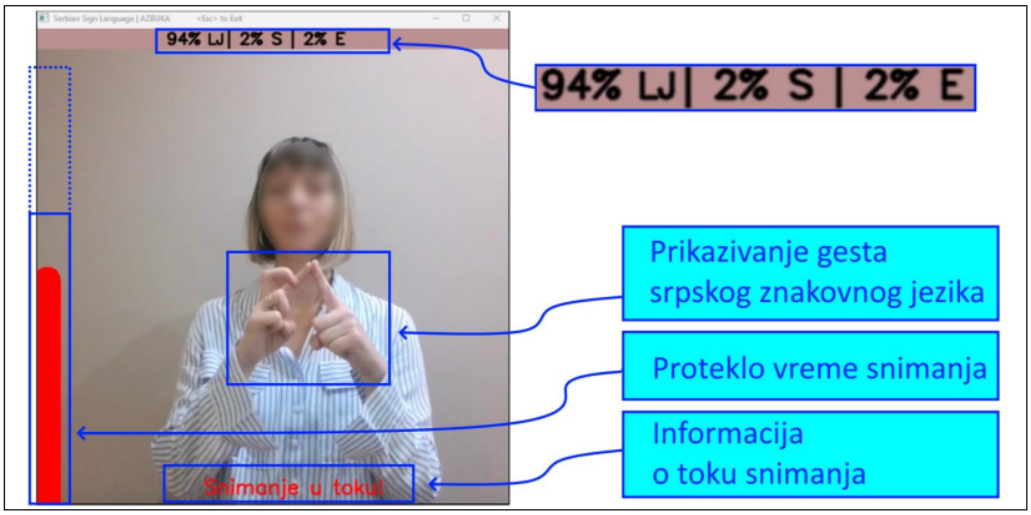
5.5. Praktična upotreba i rezultati rada aplikacije za prepoznavanje gestova srpskog znakovnog jezika

Korisnički interfejs razvijene aplikacije za prepoznavanje gestova srpskog znakovnog jezika je prikazan na slici Slika 58. Kamera bez prekida snima i u trenutku kada prepozna ruku koja je ušla u kadar, narednih 40 slika uzima kao video koji će se analizirati i na osnovu koga će se vršiti predviđanje. U levoj strani prozora se nalazi traka crvene boje, koja se vremenom smanjuje i pomaže demonstratoru da proceni brzinu kojom treba da prikaže gest. Kao standard za eksperiment u ovoj disertaciji, uzima se 40 slika, za čiju obradu se vreme razlikuje u zavisnosti od performansi računara. Ovo vreme se neće bitno razlikovati, tako da je potrebno prilagođavanje osobe koja prikazuje gestove minimalno. U centralnom delu ekrana se prikazuje sam gest – u vidu pokreta. Neki gestovi zahtevaju velike pokrete

rukama, dok su drugi statični, ali se za sve njih i baza podataka i sam postupak prepoznavanja zasniva na snimanju i čuvanju video klipova.

U vrhu prozora aplikacije, prikazani su rezultati predviđanja prethodnog snimljenog gesta. Rezultati ostaju prikazani sve dok se ne završi snimanje i prepoznavanje narednog gesta.

U donjem delu prozora, prikazane su informacije o toku snimanja, pa korisnik može biti upozoren da je „Snimanje u toku“ ili da je neophodno da „Pomeri ruke iz kadra“ u slučaju da je 40 slika već zapamćeno, a ruke su i dalje u kadru.



Slika 58. Korisnički interfejs programa za prepoznavanje gestova srpskog znakovnog jezika

Slika 59 prikazuje (u skraćenom obliku) proces prognoziranja aplikacije i predviđanja slova koje je pokazao demonstrator svojim pokretom. Vidi se i protok vremena koji je prikazan smanjivanjem crvene linije u levom delu prozora aplikacije i na samom kraju predviđanja je prikazan rezultat. Model je u ovom slučaju prognozirao da je prikazano slovo M sa 82% tačnosti, slovo N sa 18% tačnosti a 0% tačnosti daje slovu Š i svim ostalim slovima azbuke.



Slika 59. Prepoznavanje gesta za slovo „M“ sa 82% pouzdanosti

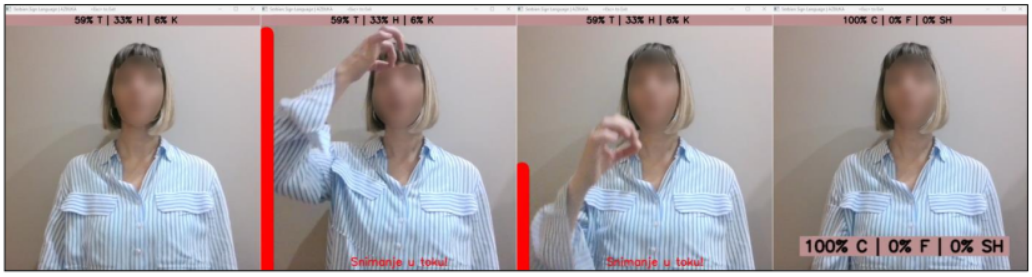
Slika 60 prikazuje gest slova „J“, koji se prikazuje samo jednom rukom, ali je neophodno da se pokaže ceo gest, pošto je za njega obavezan pokret. Za sve gestove za koje je obavezan i pokret, rezultati su malo lošiji od statičkih gestova, što se može videti u Tabeli

Tabela 12.



Slika 60. Prepoznavanje gesta za slovo „J“ sa 82% pouzdanosti

Slika 61 takođe prikazuje rad aplikacije pri obradi video zapisa dinamičkog slova „C (Ц)“. Model je u prikazanom slučaju prognozirao da je u pitanju slovo C sa verovatnoćom od 100%. Rezultati se razlikuju, a zavise kako od načina i brzine prikazanog pokreta, tako i od boje i kolorita odeće i snimljene pozadine. Šarena pozadina i boja odeće demonstratora mogu primetno smanjiti kvalitet i preciznost prepoznavanja ključnih tačaka na rukama, licu i telu učesnika u eksperimentu.



Slika 61. Prepoznavanje gesta za slovo „C (Ц)“ sa 100% pouzdanosti

Slika 62 pokazuje rezultat prepoznavanja statičkog slova „U“, za koje je potrebno koristiti obe ruke. U prikazanom slučaju, model je izuzetno uspešno prepoznao slovo, sa pouzdanošću od 98%. Model daje 2% verovatnoće slovu „P“. Sličnost između dva slova se nalazi u dva vertikalno postavljena prsta.



Slika 62. Prepoznavanje gesta za slovo „U (У)“ sa 98% pouzdanosti

Slika 63 oslikava rad aplikacije u slučaju prikazivanja statičkog slova „K“, za koje je potrebno koristiti obe ruke. Prikazani rezultat sa 67% tačno prepoznaje predstavljeni gest. Program je procenio da je prikazani gest slovo „V“ sa 22% verovatnoće, najverovatnije zbog sličnosti postavke prstiju.



Slika 63. Prepoznavanje gesta za slovo „K“ sa 67% pouzdanosti

Prepoznavanje slova „E“, koje se pokazuje sa samo jednom rukom, bez potrebe za pokretom, prikazano je na slici Slika 64. Rezultat sa prikazanog primera od 94% je izuzetno visok, a sličnost sa slovom „L“ je očigledna, pa ga je zbog toga model sa 5% predložio kao drugi izbor.



Slika 64. Prepoznavanje gesta za slovo „E“ sa 94% pouzdanosti

Slika 65 prikazuje rad programa pri prepoznavanju slova Lj (Љ). Rezultat od 99% verovatnoće demonstrira izuzetno visoku preciznost na prikazanom slučaju. Za prikazivanje slova Lj su potrebne dve ruke, ali nije potreban pokret.



Slika 65. Prepoznavanje gesta za slovo „Lj = Љ“ sa 99% pouzdanosti

Praktična primena programa za prepoznavanje pokreta koji opisuju slova azbuke srpskog znakovnog jezika donosi rezultate prikazane u tabeli

Tabela 12. Svaki od pokreta je prikazan po 20 puta, a u tabelu je uneta srednja vrednost prognoziranih rezultata. Zaključuje se da su rezultati lošiji za slova azbuke za koja je potreban pokret, za koje je potrebno komplikovanije postavljanje prstiju i koja imaju sličnu postavku ruku.

Tabela 12. Eksperimentalno izmeren procenat uspešnog prepoznavanja slova srpskog znakovnog jezika na osnovu prikazanih gestova (na uzorku od 20 snimaka po slovu).

Slovo	Broj ruku	Obavezan pokret	Procenat uspešnosti predviđanja slova na osnovu prikazanog pokreta
Г	1	НЕ	92%
Ђ	1	НЕ	89%
Е	1	НЕ	88%
И	1	НЕ	93%
Ј	1	ДА	76%
Л	1	НЕ	94%
О	1	НЕ	85%
Р	1	ДА	79%
С	1	НЕ	77%
Ћ	1	ДА	87%
Ц	1	ДА	85%
А	2	НЕ	82%
Б	2	НЕ	90%
В	2	НЕ	81%
Д	2	НЕ	83%
Ж	2	НЕ	91%
З	2	НЕ	92%
К	2	НЕ	80%
Љ	2	НЕ	68%
М	2	НЕ	84%
Н	2	НЕ	82%
Њ	2	НЕ	74%
П	2	НЕ	73%
Т	2	НЕ	78%
У	2	НЕ	86%
Ф	2	НЕ	72%
Х	2	НЕ	78%
Ч	2	НЕ	73%
Џ	2	ДА	79%
Ш	2	НЕ	82%

6. Zaključak

Celokupno istraživanje koje je direktno dovelo do rezultata prikazanih u ovoj disertaciji je trajalo oko godinu i po dana. Ono je obuhvatilo prikupljanje video snimaka za bazu sa gestovima, kreiranje modela mašinskog učenja na osnovu prikupljenih podataka i programiranje aplikacije koja koristi taj model. Rezultat je funkcionalni program baziranog na modelu mašinskog učenja koji je sposoban da samostalno prepozna prikazane gestove slova iz azbuke sprskog znakovnog jezika. Pored programa i modela mašinskog učenja, razvijena je obimna i kvalitetna baza podataka koja u sebi sadrži informacije o snimljenim ključnim tačkama ruku, lica i tela, za vreme prikazivanja gestova slova srpskog znakovnog jezika. Ova baza podataka je dostupna i omogućen je pristup istoj od strane naučne zajednice, sa ciljem njenog daljeg razvoja, ali i za implementaciju u realizaciji budućih projekata koji za cilj imaju razvoj ove oblasti. Kreirani i predstavljeni funkcionalni program i model mašinskog učenja, predstavljeni su detaljno kroz ovu disertaciju, čime su postignuti naučni doprinosi koji ujedno potvrđuju prethodno postavljene hipoteze.

6.1. Hipoteze

Opšta hipoteza koja čini osnovu istraživanja u disertaciji je potvrđena: „korišćenjem alata baziranim na veštačkoj inteligenciji može se unaprediti i olakšati komunikacija ljudi oštećenog sluha“. Na osnovu obimne i kvalitetne baze podataka srpskog znakovnog jezika kreirana je i detaljno objašnjena finalna funkcionalna aplikacija. Korišćenjem kreirane baze, kojoj se može pristupiti preko interneta, mogu se kreirati različiti modeli mašinskog učenja, a nakon toga i alati koji unapređuju i olakšavaju komunikaciju ljudi oštećenog sluha. Ove aplikacije, zbog principa na kome rade, su interaktivni, pa se informacije za razliku od postojećih rešenja, prosleđuju dvosmerno – od čoveka ka računaru i od računara ka čoveku. Prepoznavanje gestova srpskog znakovnog jezika upotrebom generisanog modela mašinskog učenja, omogućava implementaciju ovakvog sistema u različite aplikacije i uređaje. Model se može inkorporirati u komunikacione programe i platforme, kao što su Viber, WhatsApp, Teams, Webex, FaceTime, Telegram, itd. Pored navedenih programa i platformi, ovakve aplikacije i modeli se mogu iskoristiti za potrebe transkripcije znakovnog jezika i u drugim komunikacionim i video-konferencijskim uređajima proizvođača Cisco, Polycom, Logitech, itd. Zbog svoje dostupnosti,

mogućnosti proširenja i široke primenljivosti, osnovna baza podataka se može iskoristiti i kao osnova za potpuno nova programska rešenja, edukativnog, komunikacionog, poslovnog ili drugog tipa. Takođe je moguće da se kreirani program iskoristi i za proširenje prepoznavanja drugih gestova ili jezika.

Posebna hipoteza koja proizilazi iz opšte hipoteze je takođe potvrđena: „modeli mašinskog učenja mogu značajno da pomognu u razvoju algoritama za prepoznavanje znakovnog jezika“. Sprovođenjem eksperimenata sa jednodimenzionalnim, dvodimenzionalnim ili drugim oblicima podataka, menjanjem parametara modela i merenjem dobijenih rezultata, ostvaruje se napredak u radu modela u kontekstu razumevanja neophodnih izmena, kako bi se unapredili rezultati i povećala njegova funkcionalnost. Napredak u razvoju modela je primetan u praćenim i predstavljenim rezultatima njihovog funkcionisanja. Većina modela koristi statičke slike za kreiranje modela i samo prepoznavanje pokreta. Različite platforme pomoću kojih istraživači pronalaze ključne tačke na telu demonstratora takođe mogu da prouzrokuju razlike u generisanim rezultatima. Većina aplikacija i modela je zasnovana samo na statičkim slikama, ograničavajući svoje mogućnosti na samo deo gestova. Razmatranjem različitih potencijalnih rešenja, modela i algoritama, u ovoj disertaciji je projektovano, kreirano i usvojeno rešenje koje koristi dinamičke video zapise za tu namenu. Na taj način se omogućava prepoznavanje slova koje u svom prikazivanju sadrže pokret, što se smatra značajnim napretkom u odnosu na modele koji su bazirani na prepoznavanju statičkih slika.

Pojedinačna hipoteza 1: „Smanjuje se mogućnost greške u komunikaciji sa slušno oštećenim osobama“ je u potpunosti potvrđena. Komunikacija sa slušno oštećenim osobama nosi svoje izazove, probleme i ograničenja. Korišćenjem sistema sličnih ovom koji je prikazan u disertaciji, može se u velikoj meri pomoći slušno oštećenim osobama da koriste znakovni jezik sa većim stepenom tačnosti. Takođe, omogućava se celokupnoj slušno oštećenoj zajednici da kroz funkcionalni program koji omogućava prepoznavanje slova azbuke kroz pokret, nauči i usavrši razumevanje i prikazivanje obrađenih gestova.

Zasnovan na obimnoj bazi podataka koja opisuje pokrete slova azbuke, razvijeni model mašinskog učenja testiran je nakon obučavanja i dobijeni rezultati indikuju 93% tačnosti prognoziranja ishoda. Ovako visoka tačnost rezultata potvrđuje kvalitet i preciznost modela, ali može da upozori i na značajan nivo prilagođenosti

postojećim podacima. Povećan nivo prilagođenosti setu podataka za obučavanje se u određenim okolnostima može tumačiti kao potencijalno ograničenje. Testiranje generisanog modela korišćenjem skupa podataka za testiranje, samo delimično može da ukaže na performanse modela. Kako bi se utvrdilo koliko je model zaista precizan i u kojoj meri je upotrebljiv u praksi, vrši se testiranje u realnom vremenu, uz pomoć demonstratora koji prikazuju gestove ispred kamere. Dva demonstratora su pokazivala svako slovo srpskog znakovnog jezika po 10 puta, pri čemu su beleženi i izlazni rezultati modela (

Tabela 12). Prosečna preciznost predviđanja programa u realnim okolnostima je 82%, što se može smatrati sa visokim stepenom preciznosti. Zbog visoke preciznosti kreirane aplikacije i prikazanim mogućnostima u realnoj upotrebi, smatra se da je i ova hipoteza takođe potvrđena.

Pojedinačna hipoteza 2: „Unapređuje se položaj slušno oštećenih osoba u društvu“. Kreirana aplikacija i model mašinskog učenja omogućuju osobama oštećenog sluha da lakše komuniciraju i koriste poslovne i svakodnevne usluge i aplikacije na gotovo identičan način kao i ostali korisnici. Automatizacija prepoznavanja i transkripcije slova srpskog znakovnog jezika, koju ovakvi sistemi donose, može smanjiti potrebu za tumačima znakovnog jezika, što može biti korisno u situacijama kada ga nije moguće obezbediti. Moguća je implementacija modela za prepoznavanje gestova znakovnog jezika u široko rasprostranjenim programima i uređajima:

- Viber
- WhatsApp
- Microsoft Teams
- Cisco Webex
- Cisco Codecs (Pro, Kit, EQ, itd.)
- Microsoft Kinect uređaj
- Polycom video conferencing sistem
- Logitech web kamera
- Ostalim Windows, Linux ili MacOS aplikacijama, itd.

Mogućnost slušno oštećene zajednice da koriste za njih prilagođene, široko rasprostranjene sisteme i programe, zajedno sa ostalom populacijom, obezbeđuje njenim pripadnicima ravnopravniji položaj u društvu. Na ovaj način je i ova hipoteza objašnjena, potvrđena i dokazana.

Pojedinačna hipoteza 3: „Proširuje se spektar delatnosti u kojima se mogu efikasno angažovati ljudi oštećenog sluha“. Ova hipoteza je dokazana, direktno se nadovezuje na prethodnu, a vezana je za spektar delatnosti koje implementirani i kreirani alati, sistemi i programi omogućavaju. Implementacijom modela mašinskog učenja u različite sisteme, slušno oštećena zajednica može da:

- Učestvuje u video-konferencijskim sastancima tako što pokazuje gestove srpskog jezika
- Komunicira gestovima u okviru komunikacionih programa: Viber, WhatsApp, itd.
- Pruži i dobije informacije sa informacionih panela na javnim mestima
- Učestvuje u različitim kanalima poslovne i privatne komunikacije, itd.

Mogućnosti koje slušno oštećena zajednica dobija korišćenjem ovakvih sistema dodatno opremljenim modulima za prepoznavanje gestova, značajno utiče na proširenje spektra delatnosti u kojima se njeni pripadnici mogu efikasno angažovati i ravnopravno učestvovati. Sve pomenute činjenice obezbeđuju potvrdu da je i treća posebna hipoteza potvrđena.

6.2. Kompleksnost istraživačkog procesa: mogućnosti, ograničenja, rizici, prednosti i nedostaci

Mogućnosti sistema za prepoznavanje znakovnog jezika su različite i zavise od načina njegove primene i implementacije. Glavni cilj je da ovi sistemi omoguće komunikaciju između osoba oštećenog sluha koji koriste znakovni jezik sa osobama koje ga ne koriste. Oni se mogu prilagoditi tako da prepoznaju rukopis (praćenjem pokreta olovke), ekspresiju lica, budu u mogućnosti da razumeju opšti kontekst razgovora, konvertuju gestove u pisani ili govorni jezik, itd. Sposobnost sistema da omogući dvosmernu, interaktivnu komunikaciju je izuzetno značajno za socijalno osnaživanje slušno oštećene zajednice u smislu unapređenja njihove interakcije u savremeno poslovno okruženje ali i svakodnevne aktivnosti. Mogućnost da se ovakvi sistemi aktiviraju na različitim fizičkim platformama je takođe od velikog značaja. Mogu se koristiti na mobilnim telefonima, laptop ili desktop računarima, tabletima, itd.

Ograničenja ovakvih sistema su brojna. Između ostalih, ogledaju se u velikoj raznolikosti znakovnih jezika. Vidne su regionalne i lokalne razlike i u okviru gestova i pokreta samo jednog jezika, što predstavlja veliki izazov za razvoj jedinstvenog i univerzalnog sistema. Mnogi znakovi zahtevaju precizne pokrete i položaje prstiju, kao i specifične ekspresije lica ili orjentaciju tela. Precizno prepoznavanje takvih detalja pomoću modela mašinskog učenja i aplikacija koje ih koriste je složen zadatak. Netačnost i nepreciznost predviđanja modela zavisi i od načina na koji su demonstratori prikazivali znakove. Od velike je važnosti da učesnici koji pokazuju

gestove koji će se koristiti za obučavanje modela budu stručni i da na precizan i jasan način prikažu sva slova znakovnog jezika, kako bi model bio ispravno obučen. Ovo podrazumeva ozbiljnu kontrolu snimljenih video zapisa, pošto je neophodno da se oni neodgovarajućeg obavezno moraju ukloniti iz baze. Ukoliko ostanu u bazi, poremetiće proces obučavanja i model će u primeni pokazivati pogrešno prikazane gestove kao ispravne, a dobro prikazane kao neispravne. Od najveće je važnosti da baza podataka bude velika, precizna, stručno napravljena i raznolika.

Rizici koje sistemi za prevođenje gestova znakovnog jezika u pisani ili aktivni govor mogu dovesti do nesporazuma, nejasnoća i grešaka u komunikaciji. U medicinskim, pravnim ili hitnim slučajevima, greške u komunikaciji mogu imati ozbiljne posledice. Posledice nesporazuma mogu direktno negativno uticati na zdravlje, dovesti do nepovoljnih pravnih posledica ili drugih problema. Pored grešaka, zbog korišćenja kamere ili drugih senzora, često se postavlja pitanje o tome na koji se način prikupljaju, čuvaju i koriste osetljivi podaci korisnika. Prekomerna zavisnost od tehnoloških rešenja može potencijalno ograničiti i smanjiti broj osoba koje razvijaju lične veštine komunikacije i osloniti se samo na računar. I pored široke dostupnosti tehnologije, postoji rizik da ovakvi sistemi ne budu dostupni svim korisnicima, pa osobe u manje razvijenim područjima i zajednicama ne budu u mogućnosti da koriste ovakve sisteme.

Prednosti korišćenja računarskih sistema za prepoznavanje znakovnog jezika su brojne, a odnose se kako na osobe koje koriste znakovni jezik, tako i na širu zajednicu. Omogućavaju lakšu komunikaciju između slušno oštećenih i onih koji koriste znakovni jezik sa osobama koje ne znaju znakovni jezik. Smanjuje se jezička barijera i poboljšava se socijalna inkluzija. Ovakvi računarski sistemi se mogu koristiti kao obrazovni alati i platforme za učenje znakovnog jezika, a takođe i za pružanje obrazovnog interaktivnog materijala na znakovnom jeziku. Sistemi za prepoznavanje znakovnog jezika povećavaju samostalnost i nezavisnost osoba sa oštećenjem sluha, omogućavajući im komunikaciju u različitim svakodnevnim, inače izazovnim, situacijama. Olakšana im je komunikacija u posetama lekarima, kupovini, bankarskim poslovi, itd. Široka i sveobuhvatna dostupnost ovakvih sistema omogućava trenutnu komunikaciju bez potrebe za ljudskim tumačem, što je posebno korisno u hitnim ili neočekivanim situacijama. Mogućnost snimanja i arhiviranja komunikacije na znakovnom jeziku može takođe biti korisna za obrazovne, pravne ili medicinske

svrhe. Mogu da predstavljaju izuzetno vrednu podršku pri učenju znakovnog jezika, čime se promoviše veće razumevanje, prihvatanje, tolerancija i inkluzija među različitim zajednicama.

Nedostaci ovakvih sistema sprečavaju i usporavaju široku primenu u programima, na internet platformama i u specijalizovanim uređajima. Najveći izazov predstavlja postizanje visoke tačnosti prepoznavanja u realnoj upotrebi, posebno u složenim ili promenljivim uslovima osvetljenja, raznobojnoj boji odeće i pozadine, kao i stilova i brzine izvođenja gestova. Znakovni jezici i pored velikog napora za standardizaciju su i dalje puni nijansi i različitosti i ne uključuju samo pokrete ruku, već i izraze lica, položaj tela i druge neverbalne aspekte komunikacije. Ove karakteristike znakovnih jezika takođe dovode do nepotpunog ili netačnog prevođenja. Razvijanje univerzalnog sistema za prepoznavanje različitih znakovnih jezika iz navedenog razloga takođe nije moguće. Efikasno prepoznavanje gestova znakovnih jezika često zahteva naprednu tehnologiju, kao što su kvalitetne kamere i brzi procesori, što može biti skupo i nepraktično za korisnike u zemljama u razvoju. Bez razumevanja celokupnog konteksta rečenice znakovnog jezika, posmatrajući svaki gest kao nezavistan, može generisati pogrešno predviđanje rezultata. Iako se brzina i efikasnost rada ovakvih sistema stalno unapređuju, postizanje tečne i prirodne komunikacije u realnom vremenu je i dalje izazov. Korišćenje kamera, senzora za snimanje i čuvanje ovakvih podataka dovodi do zabrinutosti u pogledu privatnosti i etičkih pitanja, posebno u javnim, poslovnim, školskim, zdravstvenim ili drugim zajedničkim prostorima. Svaki korisnik demonstrira znakovni jezik na svoj način, tako da sistemi moraju da budu sposobni da se prilagode individualnim varijacijama u izvođenju znakova. Ovakva generalizacija i uopštenost u smislu prepoznavanja ličnih posebnih pokreta može biti teška za implementaciju.

Pravci budućih istraživanja

Nastavak istraživanja vezanih za razvoj sistema za prepoznavanje gestova sprskog i drugih znakovnih jezika je ključan za unapređenje komunikacije između osoba koje znakovni jezik koriste i onih koji ga ne koriste. Pravce budućih istraživanja je moguće grupisati u:

a) Teorijska naučna istraživanja
b) Praktična naučna istraživanja

Teorijska naučna istraživanja se mogu zasnovati na izmenama postojećih modela i računarskog koda, kao i ispitivanju na tim podacima dobijenih, novih rezultata. Promenom parametara, funkcija, oblika i forme podataka, aktivacionih funkcija za obučavanje modela mašinskog učenja i drugih promenljivih, moguće je generisati više različitih grupa rezultata i analizirati kako izmene utiču na performanse.

Praktična naučna istraživanja se zasnivaju na konkretnim izmenama, dodacima i korekcijama modela mašinskog učenja i aplikacije koja se na tom modelu zasniva. Moguće su višestruke izmene i unapređenja postojećeg istraživanja, kao i razvoj novih, alternativnih programa i modela, zasnovanih na ovom:

- Promena u obliku, rezoluciji, broju slika i formi prikupljanja video klipova
- Izmene u načinu obrade slika i video zapisa, kao i njihovom kvalitetu
- Promena platforme koja će biti korišćena za prepoznavanje ključnih tačaka na šakama, licu i telu
- Eksperimentisanje sa različitim alatima i platformama za razvoj i kreiranje modela mašinskog učenja
- Ispitivanje mogućnosti i postavki modela sa ciljem da se obučavanje modela dovede na najviši mogući nivo pouzdanosti i preciznosti
- Eksperimentisanje sa parametrima novih modela mašinskog učenja
- Postavljanje programa i modela na različite platforme i prilagođavanje radu na različitim uređajima
- Moguće je korišćenje aplikacije i modela predstavljene u ovoj disertaciji za druge gestove, reči i jezike
- Spajanje više pojedinačnih postojećih računarskih modula u jedan sveobuhvatni programski sistem
- Postavljanje celokupnog sistema na neku od „*Cloud*“ platformi
- Razvoj u različitim programskim jezicima
- Unapređenje kvaliteta i obima baze podataka

6.3. Strukturni aspekti naučnog istraživanja: Deskripcija, Klasifikacija, Otkriće i Objašnjenje

U okviru nivoa **naučne deskripcije** za potrebe ove disertacije, opisane su karakteristike srpskog znakovnog jezika, kao i pokreti ruku, lica i tela koji označavaju

slova azbuke srpskog znakovnog jezika. Obuhvaćen je i pregled dosadašnjih naučnih istraživanja, tehnoloških rešenja i metodologija za prepoznavanje različitih znakovnih jezika, sa posebnim fokusom na one koji su već u primeni i dokazani su praksi. Disertacija se takođe bavi detaljnom analizom i unapređenjem usvojenih rešenja, funkcija, modela i aplikacije.

Na nivou **naučne klasifikacije**, u okviru doktorata posvećenog aplikacijama i modelima za prepoznavanje srpskog znakovnog jezika, važno je bilo razviti i primeniti sistematičan pristup za kategorizaciju slova srpskog znakovnog jezika. Posebna pažnja je posvećena analizi podataka, kao i brisanju video zapisa koji ne zadovoljavaju uspostavljen nivo kvaliteta ili su snimljeni greškom, nenamerno.

Korišćeni metod **naučne deskripcije** obuhvatio je pregled i prikaz korišćene tehnologije koja se koristi u procesu prepoznavanja, uključujući hardver (kamera) i softver (algoritmi za obradu slika i tehnike mašinskog učenja). Svaki od predstavljenih elemenata u disertaciji doprinosi i utiče na unapređenje opštih performansi, preciznosti i efikasnosti sistema prepoznavanja slova.

U ovom radu je nivo **naučnog otkrića** prikazan kroz analizu i identifikaciju novih pristupa, tehnika i algoritama koji poboljšavaju tačnost, brzinu i efikasnost prepoznavanja gestova za slova srpskog znakovnog jezika. Otkrića na ovom nivou su podržana eksperimentalnim testiranjem i validacijom, dokazujući praktičnu primenljivost i doprinos u oblasti prepoznavanja gestova srpskog znakovnog jezika.

Naučno objašnjenje prikazano u ovoj disertaciji se fokusira na razradi teorijskih osnova i principa koji stoje iza metoda i principa algoritama korišćenih u sistemu prepoznavanja. Diskutovano je o različitim metodama i njihovim ograničenjima, kao i mogućim budućim unapređenjima i daljem smeru razvoja ovakvih i sličnih sistema, programa i platformi.

Literatura

[1] Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jozefowicz, R., Jia, Y., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Schuster, M., Monga, R., Moore, S., Murray, D., Olah, C., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., & Zheng, X. (2015). „TensorFlow: Large-scale machine learning on heterogeneous systems“. http://www.tensorflow.org (pristupljeno 3.3.2024).

[2] Aldalur A., Pick L.H., Schooler D., „Navigating Deaf and Hearing Cultures: An Exploration of Deaf Acculturative Stress“, The Journal of Deaf Studies and Deaf Education, Volume 26, Issue 3, July 2021, Pages 299–313, Dostupno na: https://doi.org/10.1093/deafed/enab014, 2021.

[3] Amangeldy, N.; Kudubayeva, S.; Kassymova, A.; Karipzhanova, A.; Razakhova, B.; Kuralov, S., „Sign Language Recognition Method Based on Palm Definition Model and Multiple Classification“, Sensors 2022, 22, 6621, Dostupno na: https://doi.org/10.3390/s22176621, 2022.

[4] Anand Raju, Malini Mahendiran, Shaik Shahrukh Ahmed, „Newfangled 3d human pose estimation using MediaPipe with foreground object detection“, AIP Conf, 2640 (1): 020005. Dostupno na: https://doi.org/10.1063/5.0109949, 2022.

[5] anie, S.; Varol, G.; Momeni, L.; Bull, H.; Afouras, T.; Chowdhury, H.; Fox, N.; Woll, B.; Cooper, R.; McParland, A.; et al. BBC-Oxford British Sign Language Dataset. arXiv 2021, arXiv:2111.03635. https://doi.org/10.48550/arXiv.2111.03635.

[6] Bajić, D.R.; Nikolić, G.; Gordić, M.; Mouvet, K.; Herreweghe, M.V. Language attitudes towards Serbian Sign Language and experiences with deaf education in Serbia. DiGeSt J. Divers. Gend. Stud. 2021, 8, 1

[7] Batta R., „Machine Learning Algorithms - A Review“, International Journal of Science and Research (IJSR), ISSN: 2319-7064, 2018.

[8] Bradski, G. (2000). The OpenCV Library. Dr. Dobb's Journal of Software Tools.

[9] Bradski, G.& Kaehler, A. „Learning OpenCV: Computer vision with the OpenCV library.“, O Reilly Media Inc., 2008.

[10] Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, “OpenPose: Realtime multi-person 2D pose estimation using part affinity fields,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 43, no. 1, pp. 172–186, 2019. 3.1, 4.1

[11] Caruana R., Lawrence S., & Giles L., „Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping“, Advances in Neural Information Processing Systems, 2000.

[12] Chollet, F., & others. (2015). Keras. GitHub. https://github.com/fchollet/keras (pristupljeno: 3.3.2024)

[13] Colić, V., Šta pokušavamo reći maloj deci, Dnevnik, Novi Sad, 2008.

[14] Danielle, B. et al., „Sign Language Recognition, Generation, and Translation: An Interdisciplinary Perspective“, 2019.

[15] Desai R. J., Wang S. V., Vaduganathan M., Evers T., Schneeweiss S., „Comparison of machine learning methods with traditional models for use of administrative claims with electronic medical records to predict heart failure outcomes“, JAMA network open, 3(1), e1918962-e1918962, 2020.

[16] Dimić N, Isaković Lj., O znakovnom jeziku, Univerzitet u Beogradu – Fakultet za specijalnu edukaciju i rehabilitaciju. Izdavački centar (ICF), 2018.

[17] Dimić, D. N. (2002): Metodika artikulacije, Defektološki fakultet, CIDD, Beograd; 2002.

[18] Ding Z., Huang Y., Yuan H., & Dong H.. „Introduction to reinforcement learning. Deep reinforcement learning: fundamentals, research and applications“, Computer Sciences Department University of Wisconsin, Madison, 47-123, 2020.

[19] Esakki, G., Panayides, A., Jatla, V., & Pattichis, M., „Adaptive Video Encoding For Different Video Codecs“, IEEE Access, Dostupno na: https://doi.org/10.1109/ACCESS.2021.3077313, 2021.

[20] Feurer M., Hutter F., „Hyperparameter optimization. Automated machine learning: Methods, systems, challenges“, 3-33, 2019.

[21] Figma Online UI Design Tool, dostupno na: http://www.figma.com [pregledano 10.12.2023]

[22] Fox, N.F.; Woll, B.W.; Cormier, K.C. Best practices for sign language technology research. Univers. Access Inf. Soc. 2023.

[23] Furst, J., Alkoby, K., Berthiaume, A., Chomwong, P., Davidson, M. J., Konie, B., Lancaster, G., Lytinen, S., McDonald, J., Roychoudhuri, L., Toro, J., Tomuro, N.,

& Wolfe, R. „Database design for American Sign Language. School of Computer Science“, Telecommunications and Information Systems, DePaul University, 2001.

[24] Gatalica, A., Bogati svet gluvih i nagluvih. Objavljeno na sajtu Evropskog pokreta u Srbiji, 2011.

[25] Gu, Y.; Oku, H.; Todoh, M., „American Sign Language Recognition and Translation Using Perception Neuron Wearable Inertial Motion Capture System“, Sensors 2024, 24, 453. Dostupno na: https://doi.org/10.3390/s24020453, 2024.

[26] Gupta S., Saluja K., Goyal A., Vajpayee A., Tiwari V., „Comparing the performance of machine learning algorithms using estimated accuracy“, Measurement: Sensors, 24, 100432, 2022.

[27] H. R. V. Joze and O. Koller, “MS-ASL: A large-scale data set and benchmark for understanding American Sign Language,” in British Machine Vision Conference (BMVC), 2019.

[28] Hand Landmarks Detection Guide for Python: MediaPipe—Google for Developers. Dostupno na: https://developers.google.com/mediapipe/solutions/vision/hand_landmarker/python [pregledano 14.01.2024].

[29] Humphries T., Mathur G., Napoli D.J., Padden C., Rathmann C., “Deaf Children Need Rich Language Input from the Start: Support in Advising Parents“. Children 2022, 9, 1609. Dostupno na: https://doi.org/10.3390/children9111609, 2022.

[30] Isaković, L., Kovačević, T., & Šešum, M. (2023), „Uloga i značaj prstne azbuke kod gluvih i nagluvih“, Zbornik radova-12, Međunarodni naučni skup Specijalna edukacija i rehabilitacija danas Beograd, 27–28. 109-105, 2023.

[31] J. Huang, W. Zhou, H. Li, and W. Li, “Attention-based 3D-CNNs for large-vocabulary sign language recognition,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 29, no. 9, pp. 2822–2832, 2019. 2.2, 2

[32] Javid A.M., Das S., Skoglund M., Chatterjee S., „A ReLU dense layer to improve the performance of neural networks“, ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 2810-2814). IEEE, 2021.

[33] Jayatilleke G., Ranawaka G., Wijesekara C., Kumarasinha M., „Development of mobile application through design-based research“, Asian Association of Open Universities Journal. 13. 10.1108/AAOUJ-02-2018-0013, 2018.

[34] Jeon H., Choi H., Noh D., Kim T., Lee D. „Wearable Inertial Sensor-Based Hand-Guiding Gestures Recognition Method Robust to Significant Changes in the Body-Alignment of Subject“. Multidisciplinary Digital Publishing Institute (MDPI), Basel, Switzerland, Mathematics. 2022, 10, 4753. Dostupno na: https://doi.org/10.3390/math10244753

[35] Kagirov, I., Ivanko, D., Ryumin, D., Axyonov, A., & Karpov, A. , „TheRuSLan: Database of Russian Sign Language“, Proceedings of the Twelfth Language Resources and Evaluation Conference (pp. 6079–6085). European Language Resources Association. Dostupno na: https://aclanthology.org/2020.lrec-1.746, 2020.

[36] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, „Deep Residual Learning for Image Recognition“, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770-778, 2016.

[37] Kothadiya, D.; Bhatt, C.; Sapariya, K.; Patel, K.; Gil-González, A.-B.; Corchado, J.M. Deepsign: Sign Language Detection and Recognition Using Deep Learning. Electronics 2022, 11, 1780. Dostupno na: https://doi.org/10.3390/electronics11111780, 2022.

[38] Kovačević- Isaković- Arsić 2019: Tamara Kovačević, Ljubica Isaković, Radomir Arsić. Bilingvalni pristup u obrazovanju gluve i nagluve dece. Zbornik radova FIlozofskog fakulteta u Prištini (Kosovska Mitrovica), 49 (4), 107-124.

[39] Kovačević, T., Isaković, Lj. (2020). Usvajanje znakovnog jezika kod gluve i nagluve dece predškolskog uzrasta. Baština. 513-535. doi: 10.5937/bastina30-267

[40] Krishna Chowdary, Akhil & Sandeep, Gunna. „Sign language recognitionusing deep cnn“, SRM Institute of science and technology, Chennai, 2022.

[41] Kumari, D., Anand, R.S., „Isolated Video-Based Sign Language Recognition Using a Hybrid CNN-LSTM Framework Based on Attention Mechanism“, Electronics 2024, 13, 1229. Dostupno na: https://doi.org/10.3390/electronics13071229, 2024.

[42] Lilly, J. P., „Deaf People and Society: Psychological, Sociological, and Educational Perspectives“. The Journal of Deaf Studies and Deaf Education, 26(2), 296. Dostupno na: https://doi.org/10.1093/deafed/enaa040, 2021.

[43] Lugaresi, C., Tang, J., Nash, H., McClanahan, C., Uboweja, E., Hays, M., Zhang, F., Chang, C.-L., Yong, M. G., Lee, J., Chang, W.-T., Hua, W., Georg, M., &

Grundmann, M. „MediaPipe: A Framework for Building Perception Pipelines“, arXiv. Dostupno na: https://arxiv.org/abs/1906.08172, 2019.

[44] Marković, M.M. Osobe sa Invaliditetom u Srbiji; Statistical office of the Republic of Serbia: Belgrade, Serbia, 2014; p. 35.

[45] Mijatović, S., & Ristić, I. Znakovni jezik - Razvoj i karakteristike. Zbornik radova sa međunarodne konferencije Aktuelnosti u logopediji, okupacionoj terapiji i socijalnom radu: Izmedu tradicije i tranzicije 28 - 29. 11. 2019. Beograd (Str. 25-39).

[46] Munoz-Baell, I. M., & Ruiz, M. T. „Empowering the deaf. Let the deaf be deaf“, Journal of Epidemiology & Community Health. Dostupno na: https://doi.org/10.1136/jech.54.1.40, 2000.

[47] Natarajan B., Rajalakshmi E., Elakkiya R., Kotecha K., Abraham A., Gabralla L. A., Subramaniyaswamy V., „Development of an end-to-end deep learning framework for sign language recognition, translation, and video generation“, IEEE Access, 10, 104358-104374, 2022.

[48] Nhangumbe M., Nascetti A., Georganos S., Ban Y, „Supervised and unsupervised machine learning approaches using Sentinel data for flood mapping and damage assessment in Mozambique“, Remote Sensing Applications: Society and Environment, 32, 101015, 2023.

[49] O. M. Sincan and H. Keles, “AUTSL: A large scale multi-modal Turkish Sign Language dataset and baseline methods,” IEEE Access, vol. 8, pp. 181 340–181 355, 2020.

[50] Panagakis Y., Kossaifi J., Chrysos G.G., Oldfield J., Nicolaou M. A., et al., „Tensor methods in computer vision and deep learning“, Proceedings of the IEEE, 109(5), 863-890, 2021.

[51] Papastratis, I., Chatzikonstantinou, C., Konstantinidis, D.; Dimitropoulos, K., Daras, P. Artificial Intelligence Technologies for Sign Language. Sensors 2021, 21, 5843. https://doi.org/10.3390/s21175843

[52] Pedregosa, F., Varoquaux, Ga"el, Gramfort, A., Michel, V., Thirion, B., Grisel, O., … others. (2011). „Scikit-learn: Machine learning in Python“. Journal of Machine Learning Research, 12(Oct), 2825–2830.

[53] Python Software Foundation. Python Language Reference, verzija 3.1x. Dostupno na: https://www.python.org, pristupljeno 1.1.2024.

[54] Radaković M., Marjanović M., Ristić I., Kuleto V., Ilić M.P., Dabić-Miletić S., „The Serbian Sign Language Alphabet: A Unique Authentic Dataset of Letter Sign Gestures“, Multidisciplinary Digital Publishing Institute (MDPI), Basel, Switzerland, Mathematics. 2024; 12(4):525. Dostupno na: https://doi.org/10.3390/math12040525

[55] Radaković, M. U., Marjanović, M. D., „Analiza primera kreiranja modela veštačke inteligencije korišćenjem programskog jezika Python - benefiti i izazovi“, Tehnika, 78(2), 187-194. https://doi.org/10.5937/tehnika2302187R, 2023.

[56] Radaković, M., „Audio signal preparation process for deep learning application using Python“, Sinteza 2021 - International Scientific Conference on Information Technology and Data Related Research (pp. 146-152). Singidunum University, 2021.

[57] Ristić I., Kovačević J., „Institucionalizacija vaspitanja i obrazovanja gluvih i nagluvih“, Socijalna politika u Srbiji na raskršću vekova, 2019.

[58] Sapkota B., Gurung M. K., Mali P., Gupta R., „Smart glove for sign language translation using Arduino“, In 1st KEC Conference Proceedings (Vol. 1, pp. 5-11), 2018.

[59] Sapre A., Vartak S., „Scientific Computing and Data Analysis using NumPy and Pandas“, International Research Journal of Engineering and Technology, 7, 1334-1346., 2020.

[60] Savić, Lj., Neverbalna komunikacija gluvih i njena interpretacija, Savez gluvih i nagluvih Jugosla- vije, Beograd, 2002.

[61] Sharma, P.; Tulsian, D.; Verma, C.; Sharma, P.; Nancy, N., „Translating Speech to Indian Sign Language Using Natural Language Processing“, Future Internet 2022, 14, 253. Dostupno na: https://doi.org/10.3390/fi14090253, 2022.

[62] Shin, J.; Matsuoka, A.; Hasan, M.A.M.; Srizon, A.Y. American Sign Language Alphabet Recognition by Extracting Features from Hand Pose Estimation. Sensors 2021, 21, 5856. Dostupno na: https://doi.org/10.3390/s21175856, 2021.

[63] Shiri F. M., Perumal T., Mustapha N., Mohamed R., „A comprehensive overview and comparative analysis on deep learning models: CNN, RNN, LSTM, GRU“, arXiv preprint arXiv:2305.17473, 2023.

[64] Siebert J.; Groß J.; Schroth C., „A Systematic Review of Packages for Time Series Analysis“, Eng. Proc. 2021, 5, 22. Dostupno na: https://doi.org/10.3390/engproc2021005022, 2021.

[65] Tan J., Yang J., Wu S., Chen G., & Zhao J., „A critical look at the current train/test split in machine learning“, arXiv preprint arXiv:2106.04525, 2021.

[66] Teran-Quezada, A.A.; Lopez-Cabrera, V.; Rangel, J.C.; Sanchez-Galan, J.E., „Sign-to-Text Translation from Panamanian Sign Language to Spanish in Continuous Capture Mode with Deep Neural Networks“, Big Data Cogn. Comput. 2024, 8, 25. Dostupno na: https://doi.org/10.3390/bdcc8030025, 2024.

[67] V. Athitsos, C. Neidle, S. Sclaroff, J. Nash, A. Stefan, Quan Yuan, and A. Thangali, “The American Sign Language lexicon video dataset,” in CVPRW, 2008.

[68] Vashisth, H.K.; Tarafder, T.; Aziz, R.; Arora, M.; Alpana, „Hand Gesture Recognition in Indian Sign Language Using Deep Learning“, Eng. Proc. 2023, 59, 96. Dostupno na: https://doi.org/10.3390/engproc2023059096, 2023.

[69] Virtanen P., Gommers R., Oliphant T. E., et al., „SciPy 1.0 Fundamental Algorithms for Scientific Computing in Python“, arXiv. https://ar5iv.org/abs/1907.10121, 2019.

[70] Wang Q., Ma Y., Zhao K., Tian Y., „A comprehensive survey of loss functions in machine learning“, Annals of Data Science, 1-26, 2020.

[71] World Health Organization (Svetska zdravstvena organizacija), „Gluvoća i gubitak sluha“, Preuzeto sa https://www.who.int/health-topics/hearing-loss. Datum pristupa sajtu: 20.02.2024.

[72] World Medical Association, „World Medical Association Declaration of Helsinki: Ethical principles for medical research involving human subjects“, JAMA, 310(20), 2191-2194. Dostupno na: https://doi.org/10.1001/jama.2013.281053, 2013.

[73] Xu H., van Genabith J., Xiong D., Liu Q., „Dynamically adjusting transformer batch size by monitoring gradient direction change“, arXiv preprint arXiv:2005.02008, 2020.

[74] Yang S., Yu X., Zhou Y., „LSTM and GRU neural network performance comparison study: Taking yelp review dataset as an example“, International workshop on electronic communication and artificial intelligence (IWECAI) (pp. 98-101). IEEE, 2020.

[75] Zakon o upotrebi znakovnog jezika. Službeni glasnik Republike Srbije. Broj 38/2015. Dostupno na: http://demo.paragraf.rs/WebParagrafDemo/?did=272323 [pregledano 14.02.2024].

[76] Zhang L., Wen J., Li Y., Chen J., Ye Y., Fu Y., Livingood W., „A review of machine learning in building load prediction“, Applied Energy, 285, 116452, 2021.

[77] Zhou J., Gandomi A.H., Chen F., Holzinger A., „Evaluating the quality of machine learning explanations: A survey on methods and metrics“, Electronics, 10(5), 593, 2021.

Prilozi

Prilog A. Verzije instaliranih Python modula u kreiranoj aplikaciji

```
PS H:\Doktorat\SZJ Program> python --version
Python 3.10.11

PS H:\OneDrive\Doktorat\SZJ Program> pip freeze

absl-py==1.3.0
altgraph==0.17.4
appdirs==1.4.4
astunparse==1.6.3
attrs==22.2.0
audioread==3.0.0
beautifulsoup4==4.11.1
cachetools==5.2.0
certifi==2022.12.7
cffi==1.15.1
charset-normalizer==2.1.1
contourpy==1.0.6
cycler==0.11.0
decorator==5.1.1
distlib==0.3.7
filelock==3.13.1
flatbuffers==23.5.26
fonttools==4.38.0
gast==0.4.0
google-auth==2.15.0
google-auth-oauthlib==1.2.0
google-pasta==0.2.0
grpcio==1.51.1
h5py==3.7.0
idna==3.4
joblib==1.2.0
keras==2.15.0
kiwisolver==1.4.4
libclang==14.0.6
librosa==0.9.2
llvmlite==0.39.1
Markdown==3.4.1
MarkupSafe==2.1.1
matplotlib==3.6.2
mediapipe==0.9.0.1
ml-dtypes==0.2.0
numba==0.56.4
numpy==1.23.5
oauthlib==3.2.2
opencv-contrib-python==4.7.0.68
opt-einsum==3.3.0
packaging==22.0
pandas==1.5.2
pefile==2023.2.7
Pillow==9.3.0
platformdirs==4.0.0
pooch==1.6.0
protobuf==3.20.3
pyasn1==0.4.8
```

```
pyasn1-modules==0.2.8
PyAudio==0.2.13
pycparser==2.21
pyinstaller==6.3.0
pyinstaller-hooks-contrib==2024.0
pyparsing==3.0.9
python-dateutil==2.8.2
pytz==2022.7
pywin32-ctypes==0.2.2
requests==2.28.1
requests-oauthlib==1.3.1
resampy==0.4.2
rsa==4.9
scikit-learn==1.2.0
scipy==1.9.3
six==1.16.0
sklearn==0.0.post1
soundfile==0.11.0
soupsieve==2.3.2.post1
SpeechRecognition==3.9.0
tensorboard==2.15.1
tensorboard-data-server==0.7.2
tensorboard-plugin-wit==1.8.1
tensorflow==2.15.0
tensorflow-estimator==2.15.0
tensorflow-intel==2.15.0
tensorflow-io-gcs-filesystem==0.29.0
termcolor==2.1.1
threadpoolctl==3.1.0
typing_extensions==4.4.0
urllib3==1.26.13
virtualenv==20.24.7
Werkzeug==2.2.2
wrapt==1.14.1
```

Spisak akronima

AI (Artificial Intelligence) – veštačka inteligencija

ML (Machine Leaerning) - mašinsko učenje

Adam (Adaptive Moment Estimation) -

AdaGrad (Adaptive Gradient Algorithm) –

RMSProp (Root Mean Square Propagation) -

SDG (Stochastic Gradient Descent) –

Adamax (Adam Max) – Optimizator koji se zasniva na Adam optimizatoru

Adadelta (Ada Delta) – Zasnovan na optimizatoru AdaGrad, sa drugačijim parametrima

Nadam (Nesterov-accelerated Adaptive Moment Estimation) -

RNN (Recurrent Neural Network) – Rekurentna neuronska mreža

CNN (Convolutional Neural Network) – Konvolucijska neuronska mreža

Simple RNN (Simple Recurrent Neural Networks) – jednostavne rekurentne neuronske mreže

LSTM (Long Short-Term Memory) - dugoročna kratkoročna memorija (tip RNN-a)

Bidirectional LSTM (Bidirectional Long Short-Term Memory) – vrsta LSTM

GRU (Gated Recurrent Unit) - jedinica sa kontrolisanim protokom

Dense – potpuno povezani neuronski sloj

MSE - Mean Squared Error – standardna devijacija

FTRL (Follow The Regularized Leader) – funkcija optimizatora koja prati regularizovanog lidera

val_loss (validation loss) – Matematički „gubitak“ na skupu podataka za proveru rada modela

overfitting – preveliko prilagođavanje modela mašinskog učenja obezbeđenog skupu podataka

API (Application programming interface) – programski interfejs aplikacije (omogućava komunikaciju između različitih sofverskih rešenja).

Spisak slika

Slika 1. Proces kreiranja modela mašinskog učenja (Radaković M., et al., 2024).....33

Slika 2. Koncept kreiranja baze podataka srpskog znakovnog jezika 34

Slika 3. Programski parametri video snimanja i korišćenja kamere.....35

Slika 4. Dizajn Android mobilne aplikacije DR MR 36

Slika 5. Firebase pregled postavljenih zadataka 37

Slika 6. Dizajn funkcionalnog dela Android aplikacije DR MR 37

Slika 7. Baza slika i snimaka napravljenih mobilnim telefonom.....38

Slika 8. Pozicija, veličina i udaljenost demonstratora od kamere 41

Slika 9. Parametri MediaPipe modela i okvir snimanja.....42

Slika 10. Kadrovi koji prikazuju gest SZJ slova „J“ (Radaković M., et al., 2024).....42

Slika 11. Prepoznate ključne tačke na rukama i telu gestom prikazanog slova „J“....43
Slika 12. Algoritam kreiranja baze podataka (Radaković M., et al., 2024).....44
Slika 13. Broj snimljenih video zapisa SZJ 45
Slika 14. Struktura fajlova i foldera 45
Slika 15. Prikaz gesta slova „Д“ (D) sa vrednostima ključnih tačaka na slici broj 3 od 40 46
Slika 16. Broj ključnih tačaka koji nisu prepoznati (Radaković M., et al., 2024).....47
Slika 17. Prikaz transformacije podataka iz matrica u jednodimenzionalni niz 50
Slika 18. Funkcija za konverziju vrednosti ključnih tačaka u jednodimenzionalni niz 51
Slika 19. Python izvorni kod za čitanje snimljenog NumPy zapisa iz baze podataka 52
Slika 20. Neispravno snimljeni video zapisi.....53
Slika 21. Funkcija za pronalaženje neispravnih video zapisa 55
Slika 22. Procenat neprepoznatih ključnih tačaka po slovu, po snimku.....56
Slika 23. Prikaz broja video snimaka pre provere i nakon brisanja neprihvatljivih snimaka 57
Slika 24. Pre i nakon čišćenja za dvoručne gestove.....58
Slika 25. Pre i nakon čišćenja za jednoručne gestove.....59
Slika 26. Broj odbačenih snimaka i konačni broj usvojenih snimaka za bazu podataka 60
Slika 27. Poređenje kreirane baze podataka SZJ sa drugim bazama 62
Slika 28. Odnos između veštačke inteligencije, mašinskog učenja i dubokog učenja 63
Slika 29. Klasifikacija mašinskog učenja 64
Slika 30. Programska osnova za kreiranje eksperimentalnih modela.....71
Slika 31. Kreirane datoteke sa svim grupisanim podacima i nazivima 72
Slika 32. Učitavanje prethodno generisanog fajla sa svim podacima 72
Slika 33. Centralni deo programa za kreiranje modela mašinskog učenja 73

Slika 34. Primer jednog od projektovanih modela mašinskog učenja 74

Slika 35. Grafički prikaz modela neuronske mreže 75

Slika 36. Struktura modela: Simple RNN, LSTM, GRU, Bi-LSTM (Apaydin et al., 2020) 76

Slika 37. Imaginarna ilustracija proizvoljnog modela mašinskog učenja generisana pomoću veštačke inteligencije 78

Slika 38. Kompajliranje generisanog modela mašinskog učenja 79

Slika 39. Programski kod za obučavanje modela mašinskog učenja 83

Slika 40. Procena performansi modela u 30 zadatih prolaza (epoha) 84

Slika 41. Programski kod za procenu performansi potpuno obučenog modela 84

Slika 42. Rezultat evaluacije kreiranog modela iz primera 84

Slika 43. Snimanje modela za naknadno korišćenje 85

Slika 44. Struktura računarskog koda za generisanje modela mašinskog učenja 85

Slika 45. Najprecizniji model zasnovan na jednodimenzionalnim podacima 1-D (10.320) 90

Slika 46. Promena vrednosti parametara tokom kreiranja 1-D modela mašinskog učenja 90

Slika 47. Najprecizniji model zasnovan na dvodimenzionalnim podacima 2-D (258×40) 92

Slika 48. Promena vrednosti parametara tokom kreiranja 2-D modela mašinskog učenja 92

Slika 49. Poređenje funkcije gubitka za 1-D i 2-D model 96

Slika 50. Poređenje funkcije preciznosti za 1-D i 2-D model 97

Slika 51. Poređenje preciznosti i grešaka između modela zasnovanih na jednodimenzionalnim i dvodimenzionalnim podacima 97

Slika 52. Poređenje tačnosti i gubitaka modela za 1-D i 2-D podatke 98

Slika 53. Razlike u funkcijama gubitka i preciznosti modela zasnovanog na dvodimenzionalnom skupu podataka 100

Slika 54. Preciznost modela u funkciji *batch_size* parametra 102

Slika 55. Gubitak modela u funkciji *batch_size* parametra 102

Slika 56. Početak programa za prepoznavanje srpskog znakovnog jezika 106

Slika 57. Funkcija za prepoznavanje gestova srpskog znakovnog jezika..... 108

Slika 58. Korisnički interfejs programa za prepoznavanje gestova srpskog znakovnog jezika 110

Slika 59. Prepoznavanje gesta za slovo „M“ sa 82% pouzdanosti 111

Slika 60. Prepoznavanje gesta za slovo „J“ sa 82% pouzdanosti..... 111

Slika 61. Prepoznavanje gesta za slovo „C (Ц)“ sa 100% pouzdanosti..... 111

Slika 62. Prepoznavanje gesta za slovo „U (У)“ sa 98% pouzdanosti 112

Slika 63. Prepoznavanje gesta za slovo „K“ sa 67% pouzdanosti 112

Slika 64. Prepoznavanje gesta za slovo „E“ sa 94% pouzdanosti 113

Slika 65. Prepoznavanje gesta za slovo „Lj = ЉЪ“ sa 99% pouzdanosti..... 113

Spisak tabela

Tabela 1. Prikaz naučno istraživačkih radova iz oblasti baza podataka znakovnih jezika 20

Tabela 2. Prikaz naučno istraživačkih radova iz oblasti mašinskog učenja i znakovnih jezika 24

Tabela 3. Snimanje 30 slova srpske Azbuke..... 44

Tabela 4. Procena rada modela 67

Tabela 5. Najčešće korišćeni optimizatori za sekvencijalne modele (Chollet, F. et.al, 2015) 79

Tabela 6. Optimizatori koji su manje pogodni za primenu u sekvencijalnim modelima 80

Tabela 7. Poređenje performansi kreiranih modela za jednodimenzionalni skup podataka 85

Tabela 8. Poređenje performansi kreiranih modela za dvodimenzionalni skup podataka 87

Tabela 9. Numeričko poređenje gubitaka dva modela 95

Tabela 10. Numeričko poređenje preciznosti dva modela 96

Tabela 11. Promene rezultata obučavanja modela u zavisnosti od veličine „*batch_size*“ 100

Tabela 12. Eksperimentalno izmeren procenat uspešnog prepoznavanja slova srpskog znakovnog jezika na osnovu prikazanih gestova (na uzorku od 20 snimaka po slovu). 114

RAZVOJ I EVALUACIJA MODELA ZASNOVANOG NA VEŠTAČKOJ INTELIGENCIJI ZA PREPOZNAVANJE SLOVA SRPSKOG ZNAKOVNOG JEZIKA

ORIGINALITY REPORT

1%
SIMILARITY INDEX

PRIMARY SOURCES

1	nardus.mpn.gov.rs Internet	93 words — < 1%
2	huggingface.co Internet	23 words — < 1%
3	hdl.handle.net Internet	22 words — < 1%
4	repozitorij.unipu.hr Internet	21 words — < 1%
5	medium.com Internet	20 words — < 1%
6	velog.io Internet	17 words — < 1%
7	runingcoding.tistory.com Internet	11 words — < 1%
8	elevate.airserbia.com Internet	10 words — < 1%
9	github.com Internet	10 words — < 1%

10	hal.archives-ouvertes.fr Internet	10 words — < 1%
11	eteze.ni.ac.rs Internet	9 words — < 1%
12	lutpub.lut.fi Internet	9 words — < 1%
13	www.analyticsvidhya.com Internet	9 words — < 1%
14	www.wikiwand.com Internet	9 words — < 1%
15	academic.oup.com Internet	8 words — < 1%
16	accedacris.ulpgc.es Internet	8 words — < 1%
17	pastebin.com Internet	8 words — < 1%
18	repositorio.utmachala.edu.ec Internet	8 words — < 1%
19	repozitorij.efst.unist.hr Internet	8 words — < 1%
20	vbs.rs Internet	8 words — < 1%
21	"Decision Intelligence", Springer Science and Business Media LLC, 2023 Crossref	7 words — < 1%

22	"Gesture Recognition", Springer Science and Business Media LLC, 2017 Crossref	7 words — < 1%
23	Divyang Baudhanwala, Darshan Mehta, Vijendra Kumar. "Machine learning approaches for improving precipitation forecasting in the Ambica River basin of Navsari District, Gujarat", Water Practice & Technology, 2024 Crossref	6 words — < 1%
24	notebook.community Internet	6 words — < 1%
25	www.yumpu.com Internet	6 words — < 1%

EXCLUDE QUOTES	ON	EXCLUDE SOURCES	OFF
EXCLUDE BIBLIOGRAPHY	ON	EXCLUDE MATCHES	OFF