|  |  |  |
| --- | --- | --- |
|  | UNIVERZITET U NOVOM SADU  **FAKULTET TEHNIČKIH NAUKA U NOVOM SADU** |  |

Иван Мршуља

**Праћење и препознавање геста шаке комбинацијом неуронских и традиционалних приступа**

Дипломски рад

- Основне академске студије -

Нови Сад, 2022.

|  |  |  |
| --- | --- | --- |
|  | УНИВЕРЗИТЕТ У НОВОМ САДУ  **ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА**  21000 НОВИ САД, Трг Доситеја Обрадовића 6 | Датум: |
|  |
| **ЗАДАТАК ЗА ИЗРАДУ ДИПЛОМСКОГ (BACHELOR) РАДА** | Лист: |
| 1/1 |

*(Податке уноси предметни наставник - ментор)*

| Врста студија: | **Основне академске студије** |
| --- | --- |
| Студијски програм: | **Софтверско инжењерство и информационе технологије** |
| Руководилац студијског програма: | **проф. др Мирослав Зарић** |

|  |  |  |  |
| --- | --- | --- | --- |
| Студент: | **Иван Мршуља** | Број индекса: | **SW 65/2018** |
| Област: | **Електротехничко и рачунарско инжењерство** | | |
| Ментор: | **Др Јелена Сливка, редовни професор** | | |
| НА ОСНОВУ ПОДНЕТЕ ПРИЈАВЕ, ПРИЛОЖЕНЕ ДОКУМЕНТАЦИЈЕ И ОДРЕДБИ СТАТУТА ФАКУЛТЕТА ИЗДАЈЕ СЕ ЗАДАТАК ЗА ДИПЛОМСКИ РАД, СА СЛЕДЕЋИМ ЕЛЕМЕНТИМА:   * проблем – тема рада; * начин решавања проблема и начин практичне провере резултата рада, ако је таква провера неопходна; * литература | | | |

**НАСЛОВ ДИПЛОМСКОГ (BACHELOR) РАДА:**

|  |
| --- |
| **Праћење и препознавање геста шаке комбинацијом неуронских и традиционалних приступа** |

**ТЕКСТ ЗАДАТКА:**

|  |
| --- |
| 1. Анализирати стање у области.  2. Израдити спецификацију захтева софтверског решења.  3. Израдити спецификацију дизајна софтверског решења.  4. Имплементирати софтверско решење према израђеној спецификацији.  5. Тестирати имплементирано софтверско решење.  6. Документовати (1), (2), (3), (4) и (5). |

|  |  |
| --- | --- |
| Руководилац студијског програма: | Ментор рада: |
|  |  |

|  |
| --- |
| Примерак за:  - Студента;  - Ментора |

# КЉУЧНА ДОКУМЕНТАЦИЈСКА ИНФОРМАЦИЈА

|  |  |
| --- | --- |
| Редни број, **РБР**: |  |
| Идентификациони број, **ИБР**: |  |
| Тип документације, **ТД**: | монографска публикација |
| Тип записа, **ТЗ**: | текстуални штампани документ |
| Врста рада, **ВР**: | дипломски рад |
| Аутор, **АУ**: | Иван Мршуља |
| Ментор, **МН**: | Др Јелена Сливка, редовни професор |
| Наслов рада, **НР**: | Наслов рада |
| Језик публикације, **ЈП**: | српски |
| Језик извода, **ЈИ**: | српски / енглески |
| Земља публиковања, **ЗП**: | Србија |
| Уже географско подручје, **УГП**: | Војводина |
| Година, **ГО**: | 2022 |
| Издавач, **ИЗ**: | ауторски репринт |
| Место и адреса, **МА**: | Нови Сад, Факултет техничких наука, Трг Доситеја Обрадовића 6 |
| Физички опис рада, **ФО**: | бр. поглавља / страница / цитата / табела / слика / графикона / прилога |
| Научна област, **НО**: | Софтверско инжењерство и информационе технологије |
| Научна дисциплина, **НД**: | Софтверско инжењерство |
| Предметна одредница /  кључне речи, **ПО**: | Праћење шаке, препознавање геста шаке |
| **УДК** |  |
| Чува се, **ЧУ**: | Библиотека Факултета техничких наука, Трг Доситеја Обрадовића 6, Нови Сад |
| Важна напомена, **ВН**: |  |
| Извод, **ИЗ**: | апстракт – један пасус који добро описује суштину рада – проблем, мотивацију, назнаку решења и резултат. |
| Датум прихватања теме, **ДП**: |  |
| Датум одбране, **ДО**: |  |
| Чланови комисије, **КО**: |  |
| председник | др Име Презиме, звање |
| члан | др Име Презиме, звање |
| ментор | др Име Презиме, звање |
| Потпис ментора | |

# KEY WORDS DOCUMENTATION

|  |  |
| --- | --- |
| Accession number, **ANO**: |  |
| Identification number, **INO**: |  |
| Document type, **DT**: | monographic publication |
| Type of record, **TR**: | textual material |
| Contents code, **CC**: | bachelor thesis |
| Author, **AU**: | Ivan Mršulja |
| Mentor, **MN**: | Jelena Slivka, full professor, PhD |
| Title, **TI**: | Hand tracking and gesture recognition using neural and traditional model pipeline |
| Language of text, **LT**: | Serbian |
| Language of abstract, **LA**: | Serbian / English |
| Country of publication, **CP**: | Serbia |
| Locality of publication, **LP**: | Vojvodina |
| Publication year, **PY**: | 2022 |
| Publisher, **PB**: | author’s reprint |
| Publication place, **PP**: | Novi Sad, Faculty of Technical Sciences, Trg Dositeja Obradovića 6 |
| Physical description, **PD**: | br. poglavlja / stranica / citata / tabela / slika / grafikona / priloga |
| Scientific field, **SF**: | Software Engineering and Information Technologies |
| Scientific discipline, **SD**: | Software Engineering |
| Subject / Keywords, **S/KW**: | Hand tracking, gesture recognition |
| **UDC** |  |
| Holding data, **HD**: | Library of the Faculty of Technical Sciences, Trg Dositeja Obradovića 6, Novi Sad |
| Note, **N**: |  |
| Abstract, **AB**: | Prevod apstrakta na engleski |
| Accepted by sci. Board on, **ASB**: |  |
| Defended on, **DE**: |  |
| Defense board, **DB**: |  |
| president | Ime i prezime, zvanje na eng., PhD |
| member | Ime i prezime, zvanje na eng., PhD |
| mentor | Ime i prezime, zvanje na eng., PhD |
| Mentor's signature | |

**SADRŽAJ**

[КЉУЧНА ДОКУМЕНТАЦИЈСКА ИНФОРМАЦИЈА 4](#_Toc94596412)

[KEY WORDS DOCUMENTATION 5](#_Toc94596413)

[1. UVOD 9](#_Toc94596414)

[2. PREGLED STANJA U OBLASTI 11](#_Toc94596415)

[3. TEORIJSKI POJMOVI I DEFINICIJE 17](#_Toc94596416)

[3.1 Opis algoritma na koji se vaš rad oslanja 17](#_Toc94596417)

[3.2 Opis algoritma na koji se vaš rad oslanja 17](#_Toc94596418)

[4. METODOLOGIJA 19](#_Toc94596419)

[4.1 Prvi modul sistema (izmeniti naslov da bude specifičan za vaše rešenje) 23](#_Toc94596420)

[4.2 Drugi modul sistema 23](#_Toc94596421)

[4.3 Korišćeni alati 23](#_Toc94596422)

[5. EKSPERIMENTI 25](#_Toc94596423)

[5.1 Skup(ovi) podataka 25](#_Toc94596424)

[5.2 Eksperiment 1 26](#_Toc94596425)

[5.3 Evaluacija 27](#_Toc94596426)

[6. REZULTATI (I DISKUSIJA) 29](#_Toc94596427)

[7. DISKUSIJA 31](#_Toc94596428)

[8. ZAKLJUČAK 33](#_Toc94596429)

[9. LITERATURA 35](#_Toc94596430)

[10. BIOGRAFIJA 37](#_Toc94596431)

# UVOD

Могућност рачунара да перципира облик и кретање шаке може бити једна од виталних компоненти која побољшава *user experience* у широком спектру технолошких домена и платформи. Праћење шаке и препознавање гестикулације исте представљају један од основних проблема у овом домену истраживања.

На пример, може да представља основу за разумевање знаковног језика и контролу покрета рукама, а такође може да омогући преклапање дигиталног садржаја и информација из физичког света у системима проширене стварности (augmented reality). Иако је људима природна, робусна перцепција руку у реалном времену је изразито изазован задатак у домену компјутерске визије, јер руке често заклањају саме себе или једна другу (нпр. оклузије прста/длана и дрхтање руку) и немају регионе високог контраста.

У овом раду, рађено је праћење и препознавање геста шаке у домену разумијевања знаковног језика, конкретно, разумијевања алфабета немачког језика.

За рјешавање овог проблема, имплементиран је јеноставан pipeline koji ima dva koraka. Sistem kao ulazne podatke dobija frejmove real-time снимка са веб камере које даје горе поменутом pipeline-у на обрађивање. Prvi korak u obradи слике јесте одређивање 21 кључног oбележја (landmark-а) шаке док други корак представља класификациони модел. Класификациони модел као улаз прима координате сваког од 21 кључног обележја и као излаз даје коначну предикцију геста шаке. Од технологија коришћене су MediaPipe радни оквир (framework) за детекцију кључних обележја шаке као и *scikit-learn* и *Keras* библиотеке за имплементацију класификационих модела.

Евалуација модела вршена је евалуацијом квалитета коначне предикције истог. Квалитет одређивања кључних обележја није евалуиран посебно. Разлог за ово је тај што први корак у проточној обради (детекција и локализација кључних обележја) служи као својеврсни feature extractor те се као сваки други вид претпроцесирања улазне слике не евалуира сам за себе већ у склопу читавог модела проточне обраде. С обзиром да је улазни скуп података релативно балансиран, као мјере перформансе користе се тачност, прецизност, одзив као и микро Ф мјера. Такође, пошто је веома битно да овај модел ради што ближе реалном времену, као мјеру перформансе квалитета рада у реалним условима користи се однос фрејмова које модел може да обради у секунди (Frames Per Second) и микро Ф мјере. Главни закључци који су изведени из евалуације показују да овај приступ има најбољи однос између брзине обраде и квалитета предикције у односу на потпуно трдиционлне и потпуно неуронске приступе (поготово на слбијем хрдверу). Такође, имплементирани приступ се показао супериорно у односу на потпуно неуронске приступе и у домену обучавања јер му је потребно доста мање података за обучавање. На крају, треба напоменути да је мој приступ до одређене мјере отпоран код примјене на различитим нијансама коже као и заклањања шаке што није случај код традиционалних приступа.

Додати пасус са организацијом остатка рада по поглављима.

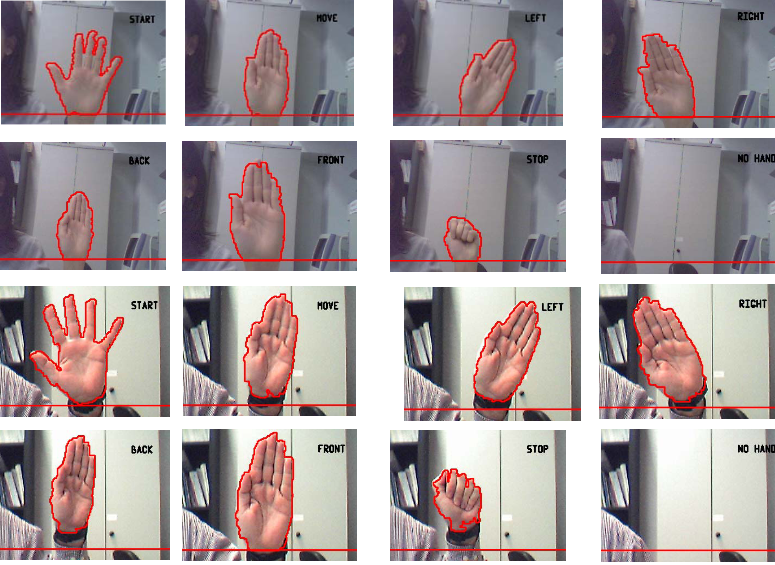
# PREGLED STANJA U OBLASTI

У овом поглављу је наведен историјат развоја приступа за праћење и препознаваље геста шаке као и тренутни state-of-the-art приступи код рјешавања овог проблемског домена. Код избора релевантних радова вођено је рачуна да су представљене методологије имале довољно добре перформансе приликом класификације како би се могли користити у пракси као и да су употребљиви из аспекта интеракције човека-рачунара тј. да се резултат могао добити у реалном времену или близу реалног времена као и да је погодан за реалну примјену. Такође, узети су у обзир само они радови који класификацију раде искључиво коришћењем фрејмова видеа као улаза те су сви Skeleton-Based, Depth-Based i 3D-Based modeli otpisani u startu.

Први приступ за праћење и препознавање геста шаке кој је дао задовољавајућ резултат за реалну употребу био је заснован на pipeline-у који се састојао из три фазе:

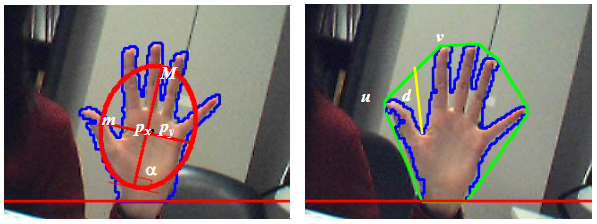
* традиционалне технике сегментације помоћу пробабилистичког модела
* ручног одређивања релевантних обележја
* систем заснован на правилима (rule-based) који је служио за одређивање геста

Аутори су развили овај систем за потребе контролисања кретања аватара у видео игрици те је коначна лабела имала 8 класа: start, move, stop, left, right, front, back, no hand. Прије почетка коришћења апликације (приликом сваког покретања) корисник треба да у исцртани квадрат на екрану услика свој длан како би сегментациони алгоритам могао да се обучи. Сви усликани пиксели длана пребацују се из RGB (Red Green Blue) у HSV (Hue Saturation Value) репрезентацију јер је она доста погоднија за даљу обраду. Функција густине вјероватноће боје коже моделована је гаусовом расподјелом а параметри се одређују стандардном методом максималне вјеродостојности (maximum likelihood estimation). Након процеса обучавања модел је спреман за употребу. За сваки пиксел улазне слике одређује се вјероватноћа да ли припада длану, а затим се користи CCA (Connected Components Аlgorithm) преко којег се пиксели длана групишу у један blob. Ивични пиксели овог blob-a представљају контуре шаке.



Примјер 2.1. Приказ одређивања контура шаке употребом пробабилистичког модела

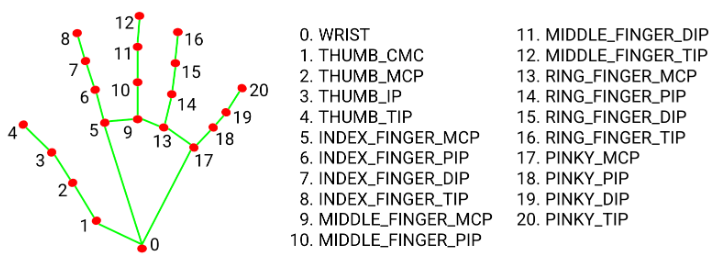
Када сегментациони алгоритам одреди контуре шаке, користи се једноставан хеуристички алгоритам који одређује центар длана. Циљ ове фазе је да се длан представи као елипса и да се одреде мала и велика оса (m, M респективно). С обзиром да као излаз из предходне фазе имамо ивичне пикселе региона, одређивање ових параметара је тривијално и своди се на одређивање линије која представља највеће растојање два пиксела унутар региона (велика оса) и линије која је нормална на њу и спаја два пиксела ивице који се налазе на супротним странама шаке (мала оса). Додатно, стандардним техникама компјутерске визије, рачуна се и конвексни овотач региона. На основу конвексног омотача и репрезентације путем елипсе могуће је одредити низ контурних тачака између два узастопна конвексна врха прста/длана. Ова секвенца формира такозвани дефект конвексности (Convexity Defect) и могуће је израчунати дубину и-тог дефекта конвексности di.



Примјер 2.2 Приказ екстракције обележја са детектованог региона од интереса

Просјек дужина свих дефеката конвексности зједно са дужином мале и велике осе, координатама центра шаке као и орјентација шаке се узимају као обележја у финалној фази која се ослања на неколико ручно написаних if-then-else правила која, на крају, дају коначан резултат процеса класификације. Модел је евалуиран на ручно прикупљеном скупу података на коме су аутори успјели да постигну веома добре перформансе. За метрику перформансе коришћена је тачност која је израчуната за сваку класу те је након тога упросјечена. Просјечна тачност на свих 8 класа износила је одличних 98%. Главна мана овог приступа је та што се приликом сваког покретања мора поново обучавти пробабилистички модел који ради сегментацију слике. Истина је да се овим постигла робусност јер ће модел радити на различитим типовима коже корисника али остаје проблем коришћења у лоше освијетљеној просторији. Наиме, модел се мора користити у соби гдје је освијетљење константно јер промјеном освјетљења добијамо потпуно различите вриједности пиксела те се пробабилистички модел мора поново обучити. Такође, у сцени гдје је позадина сличне боје као и кожа корисника (поготово на камерама лошег квалитета) аутори наводе да се модел понаша непредвидиво. Још једна од великих мана овог приступа, коју сами аутори наводе, је та што је немогуће користити овај модел код гестова приликом којих нису виљиви сви прсти.

У раду [] проблем са поновним обучавањем пробабилистичком модела код употребе на различитим нијансама коже као и проблем позадине са бојом сличној боји коже превазиђен је на начин што је као додатна опрема коришћена рукавица која је офарбана бојом која се јако ријетко може наћи у позадини (и природи уопште). Самим тим, једноставном примјеном сегментације помоћу прага је могуће извући регију од интереса. Поново се ручно имплементираним техникма извлче обележја која се у овом приступу убацују у k-nearest neighbors (KNN) класификатор. Аутори овог рада су добили скоро идентичне резултате као аутори рада [] на јако сличном скупу података с тиме што су превазишли један од проблема увођењем додатне опреме. Са друге стране, идаље је остао велики проблем употребе модела у сцени која је неједнако освијетљена и генерално је остао проблем детекције геста код којег нису видљиви сви прсти. Рјешење за ове проблеме изложено је у [] гдје T. Maung предлаже да, за разлику од свих претходних модела, немамо одвојене фазе за екстракцију обележја и класификацију већ је све било енкапсулирано у једну потпуно повезану неуронску мрежу са hard-limit активационом функцијом. Овај приступ дао је око 90% тачности на ручно прикуплјеном скупу података али је имао велику ману да ради само уколико се шака налази непосредно испред камере. У раду [] предложено је коришћење SSD () модела за локализацију шаке и класификацију геста (конкретно модела из YOLO породице). Иако доста тежак за обучавање, овај модел ријешио је све проблеме претходно набројаних приступа и дуго се сматрао као state-of-the-art рјешење све до појаве MediapipeHands модела. Аутори овог рада су увидјели да је главни проблем у овом проблемском домену управо екстракција обележја и ријешили су да направе pipeline модел који ће комбинацијом неуронских и традиционалних приступа моћи да одреди 21 кључно обележје шаке које би касније било који модел машинског учења могао искористити као улаз за процес класификације геста.



Примјер 2.3 Визуализација сваког кључног обележја заједно са његовом ознаком

**Cilj poglavlja**: pokazujete da ste upoznati sa oblašću kojoj pripada vaš rad. Odnosno, znate koji načini rešavanja postavljenog problema postoje (najsvežiji radovi, metodologije, aplikacije i najbolji postignuti rezultati).

**Organizacija poglavlja.** Prvi pasus treba da da kontekst – čime se bavi ovo poglavlje i koji kriterijum ste primenili kada ste tražili slične radove (po čemu konkretno su slični). Dalje, najčešći način organizacije je hronološki: krećete od najstarijih radova i prikazujete razvoj oblasti kroz vreme**:**

1. Prvo predstavite najvažnije radove iz oblasti. To su obično radovi koji su malo stariji, imaju veliki broj citata i autori su uticajni naučnici iz oblasti. Važno je da se vidi kako se kroz vreme menjala metodologija i kako su se menjali rezultati (primer 3.1). **Važno je da istaknete mane metodologija** da bi se moglo razumeti zašto su stare tehnike zamenjene novim (primer 3.2).
2. Predstavite najskorije radove iz oblasti (prethodnih pet godina). Ako ne umete da procenite kvalitet rada, obratite se mentoru.
3. Na kraju, najdetaljnije opisujete (primer 3.3) najskorije radove koji su najbliži vašem radu. Ovde je cilj da na neki način opravdate metodologiju i da pokažete rezultate sa kojima se kasnije poredite u poglavlju „Rezultati i Diskusija“.

|  |
| --- |
| **(prvo najvažniji radovi)** Prve tehnike za prepoznavanje imenovanih entiteta u tekstu zasnivale su se na ručno kreiranim pravilima [Reference]. Modeli su obučavani/evaluirani na tim i tim korpusima [Reference]. Najbolji rezultati su bili ti i ti (dati tačne mere ali ukratko).  **(zatim skoriji radovi)** U poslednjih X godina/meseci sistemi za to i to uglavnom su zasnovani na transformer modelima [Reference]. |
| **(najvažniji radovi:)**  Detekcija objekta je jedno od najtežih problema u oblasti kompjuterske vizije trenutno, jer obuhvata i klasifikaciju i lokalizaciju samog objekta [11].  Popularni pristupi rešavanju problema klasifikacije slika pre razvoja dubokog učenja (engl. *deep learning*) bili su SIFT *(Scale-Invariant Feature Transform*) [14] i HOG (*Histogram of Oriented Gradients*) [13]. Ipak, ovakvi pristupi su previše jednostavni da bi ponudili robusno i generičko rešenje [12]. Uglavnom se koriste za usko definisane probleme kao što je detekcija lica ili pešaka na ulici. U [13] istrenirani set HOGklasifikatora je korišćen da prepozna vozila na 284 slika i imao je tačnost od 88%.  U radu [15] se problem detekcije objekta deli na: (1) pronalaženje regija od interesa i (2) prosleđivanje pojedinačnih regija od interesa klasifikatoru (na primer, SVM), koji vrši klasifikaciju objekta predstavljenog u regiji od interesa na unapred definisane kategorije. U cilju pronalaženja regije od interesa koristi se tehnika klizećeg prozora. Međutim, ova tehnika se u današnje vreme smatra naivnim pristupom za rešavanje zbog velike vremenske kompleksnosti usled pokušaja klasifikacije ogromnog broja regija od interesa.  **(skoriji radovi:)**  Noviji radovi uvode konvolutivne neuronske mreže kao primer optimizovanog rešenja za problem detekcije.... |

Primer 3.1 Opis prethodnog rada u hronološkom redosledu zarad prikaza.

|  |
| --- |
| Ručno kreirana pravila su previše prilagođena korpusima na osnovu kojih su napravljena i teško se adaptiraju na nove korpuse. Iz tih razloga sledeći pravac u oblasti bio je obučavanje ML modela [Reference]. |

Primer 3.2 Isticanje nedostataka metodologije.

|  |
| --- |
| Metodologija vašeg rada tipično je jedna ili više od sledećih stavki:   * rešavanje specifičnog problema pomoću *state-of-the-art* (SOTA) metodologija za slične probleme, * upoređivanje SOTA modela za rešavanje nekog problema na kome ti modeli još nisu probani, * modifikacija postojeće SOTA metodologije.   U skladu sa tim, ovaj deo pišete tako da vaši izbori budu potkrepljeni literaturom. Ako menjate neku metodologiju, opišite detaljnije rad u kome je ona definisana i istaknite mane zbog kojih je menjate. Ako koristite neku metodologiju, detaljno opišite rad tako da bude jasno zašto je ona najbolja za vaš problem. |

Primer 3.3 Detaljniji opis rada koji je najbliži vašem rešenju.

**Nivo detalja koji ćete prikazati pri opisu radova:**

1. **Rad uključen radi prikaza razvoja oblasti**: dovoljna su jedna ili dve rečenice koje sumiraju stanje oblasti u tom momentu zajedno sa rezultatima i manama. Citirajte više radova u jednoj rečenici, kao što je prikazano u primeru 3.2.
2. **Najskoriji radove značajni za vašu metodologiju**: pišite malo detaljnije – možete izdvojiti ceo paragraf za jedan rad. Međutim, ne biste trebali imati jako puno radova koje ćete opisivati na tom nivou detalja. Opis treba da zadrži samo najvažnije detalje kao što su: cilj rada, metodologija rada, korpus na kome je evaluiran, rezultate i prednosti i mane.

Bez obzira na dužinu opisa (nivo detalja), opis rada mora biti jasan čitaocu koji ga nije čitao –čitalac mora da shvati aspekt ili aspekte rada koji koje prikazujete. Na primer:

* ako želite da istaknete metodologiju nekog rada u kom autori koriste model X koji nije opšte poznat u oblasti, nemojte samo napisati „upotrebili su model X za to i to..“, već objasnite taj model.
* ako želite da istaknete rezultate rada, a rad koristi neku meru performansi koja nije opšte poznata onda objasnite meru i samu vrednost (da li je velika ili mala) da čitalac nema potrebe da čita rad kako bih to saznao.

**Alternativa za organizaciju poglavlja**. Na početku je rečeno da je tipična organizacija hronološka. Poglavlje se umesto toga može organizovati i po potpoglavljima. Ova organizacija ima smisla ako vaš sistem rešava više podproblema u nekoj oblasti. Primer 3.4 prikazuje ovakvu organizaciju.

|  |
| --- |
| Recimo, bavite se prvo klasifikacijom delova dokumenta u sekcije, pa onda prepoznavanjem imenovanih entiteta u tim sekcijama. U tom slučaju, poglavlje organizujete na potpoglavlja koji prikazuju radove koji se bave:   1. klasifikacijom delova dokumenata 2. prepoznavanjem imenovanih entiteta u tekstu 3. kombinovanjem te dve metodologije i slični su vašem radu.   Prva dve potpoglavlja ne treba da budu jako detaljna jer je vaš fokus kombinacija metodologija. Svako potpoglavlje trebalo bi da bude pisana hronološki po uputstvu iznad. |
| U ovom poglavlju predstavićemo postojeća rešenja koja se bave prepoznavanjem registarskih tablica vozila. U cilju prepoznavanja karaktera na registarskim tablicama, potrebno je prvo prepoznati gde se nalaze vozila na slici, a zatim, izdvojiti tablice pojedinačnih vozila i prepoznati karaktere na njima. U skladu sa time, ovo poglavlje je podeljeno na dva potpoglavlja. U potpoglavlju 2.1. predstavljeni su radovi koji se bave detekcijom vozila na slici. Zatim su u potpoglavlju 2.2. predstavljeni radovi koji se bave detekcijom i prepoznavanjem registarske tablice. |

Primer 3.4 Primeri podela poglavlja na potpoglavlja.

|  |
| --- |
| **(Prvi paragraf daje kontekst – koji radovi će biti predstavljeni)** U ovom poglavlju će se razmatrati rešenja koja su relevantna za problem prepoznavanja emocija u muzici.  **(Slede paragrafi gde se opisuje razvoj oblasti hronološki)** Prvi put je 2007. godine klasifikacija muzike po raspoloženjima uključena u MIREX (engl*. Music Information Research Evaluation* *eXchange*), radni okvir za formalno evaluiranje MIR sistema i algoritama [2]. Sistemi sa visokim performansama u ovoj kategoriji pokazuju napredak svake godine, upotrebom isključivo akustičnih obeležja. Neki od sistema su prvobitno dizajnirani za klasifikaciju žanra, a naknadno prisvojeni i u problemu klasifikacije emocija. Prethodno korišćene metode uključuju analizu teksta i meta-podataka pesama, međutim, imali su manje uspeha od metoda koje koriste analizu akustičnih obeležja [2].  U jednoj od prvih publikacija na ovu temu, Li i Ogihara [8] su obučavanjem SVM klasifikatora (engl. *Support Vector Machines*) klasifikovali muziku u 13 kategorija, korišćenjem ručno anotirane biblioteke od 500 pesama u trajanju od 30 sekundi, gde su bili uključeni žanrovi u domenu ambijentalne, klasične, *fusion* i džez muzike. Postignuta je tačnost od svega 45%, što je već tada pokazalo da se ovom problemu treba pristupiti na drugačiji način kako bi se rezultati poboljšali.  Lu i saradnici [9] su pristupili problemu koristeći GMM klasifikator (engl. *Gaussian Mixture Model*) za četiri osnovna kvadranta u VA modelu emocija, koji su korišćeni i u rešenju predstavljenom u ovom radu. Skup podataka je sadržao 800 isečaka klasične muzike u trajanju od 20 sekundi i dostignuta je tačnost 85%. Visoka tačnost je bila posledica korišćenja isključivo klasične muzike pri obučavanju i evaluaciji modela.  Cao i Li su 2009. predložili sistem koji je u to vreme bio prvi u nekoliko kategorija, sa tačnošću 65.7% na problemu klasifikacije muzike po emociji, upotrebom GSV-SVM modela (*Gaussian Super Vector - Support Vector Machine*) [10].  Narednih godina je rasla upotreba neuronskih mreža za rešavanje problema klasifikacije. Sarkar i saradnici [11] su dokazali da duboke neuronske mreže daleko nadmašuju SVM klasifikatore u MER polju istraživanja. Prvobitno rešenje je definisano 2013. i 2015. Bazirano je na LSTM-RNN modelu (engl. *Long Short-Term Memory Recurrent Neural Network*) [12]. Zatim su 2016. KNN (engl. *k-Nearest Neighbour*) i neuronske mreže predložene za prepoznavanje emocije u muzici od strane Hana [13].  **(Sledi pregled radova sličnih predstavljenom rešenju. Primetite da se radovi detaljnije opisuju, uz komentare koji naglašavaju sličnosti i razlike sa rešenjem prikazanim u ovom diplomskom radu)** U rešenju [7] upotrebljena je CNN arhitektura nad skupom podataka od 1000 pesama, trajanja 45 sekundi, iz kojih su generisani spektrogrami. Model se sastoji od tri konvoluciona sloja, tri *pooling* sloja i na kraju dva *fully-connected* sloja, sa *ReLU* aktivacionom funkcijom. Takođe je dodat *dropout* sa vrednošću 0.5kako bi se sprečilo preprilagođavanje (engl. *overfitting*) modela. Upotrebljen je *Adam* optimizator sa *learning rate*-om 0.0001. Izlaz je takođe definisan pomoću VA modela. Najveća tačnost koja je dobijena ovakvim pristupom je 78%. Uzimajući u obzir visoku subjektivnost prepoznavanja emocija, autori su napomenuli da je postignut veoma dobar rezultat, ali da postoji dosta prostora za poboljšanja. Slična arhitektura je primenjena i u ovom radu, sa razlikom u skupu podataka.  U radu [6] opisana je arhitektura neuronske mreže zasnovana na dubokom učenju. Korišćena je baza sa pesmama u trajanju od 30 sekundi, kao i *valence*-*arousal* emocionalni model. Skup podataka iz rada [6] koristi dimenzionalni model za oznaku emocija, nasuprot kategoričkog modela koji je u korišćen u ovom radu. Kategorički model može biti problematičan jer ne postoji konsenzus o broju kategorija, dok se kod dimenzionalnog modela koriste numeričke vrednosti za *valence* i *arousal* obeležja, za čiji odabir su potrebni eksperti. Kao ulaz u sistem su iz zvuka izdvojena MFCC (engl. *Mel Frequency Cepstral Coefficients*) obeležja zajedno sa LMFB obeležjima (engl. *Log Mel-FilterBank energies*) obeležjima, upotrebom jednostavnog CNN modela za njihovu ekstrakciju. Rezultati pokazuju da dodavanje novih obeležja poboljšava performanse klasifikacije. Za obučavanje je upotrebljen LSTM-DNN klasifikator sa postignutim rezultatima 91.93%. Daljom selekcijom obeležja rezultat je poboljšan na čak 99.19%. Samim tim ovakav model predstavlja *state-of-the-art* na ovom problemu.  Prednosti *state-of-the-art* rešenja u odnosu na ovde prikazano rešenje je korišćenje skupova podataka koji su označeni od strane domenskih eksperata ili glasanjem velikog broja ljudi. Takođe je prednost upotreba hardverski zahtevnih arhitektura neuronskih mreža, čiji parametri su optimalno podešeni. |

Primer 3.5 Primer celokupnog poglavlja „Pregled stanja u oblasti“ uz komentare

# TEORIJSKI POJMOVI I DEFINICIJE

**Cilj poglavlja**: predstaviti teorijske osnove neophodne za razumevanje rešenja koje ste primenili u svom diplomskom radu.

**Važno:** Napravite jasno razgraničenje između pojašnjenja algoritama koje upotrebljavate (ovo poglavlje) i načina na koje upotrebljavate ove algoritme zarad konstrukcije vašeg rešenja (poglavlje 1).

Ako delite ovo poglavlje na više celina (koje odgovaraju algoritmima koje opisujete), ovde u uvodu poglavlja opišite šta je cilj poglavlja i najavite podelu na potpoglavlja (u potpoglavlju 4.1 opisan je..., a u potpoglavlju 4.2...).

|  |
| --- |
| Metod mašinskog učenja korišćen za detekciju i prepoznavanje saobraćajnih znakova u ovom radu je konvoluciona neuronska mreža. U ovom poglavlju će prvo biti objašnjene klasične veštačke neuronske mreže (poglavlje 3.1) koje predstavljaju osnovu za razumevanje konvolucionih neuronskih mreža, predstavljenih u poglavlju 3.2.   * 1. Veštačke neuronske mreže   2. Konvolucione neuronske mreže |

Primer 4.1 Primer podele poglavlja „Teorijski pojmovi i definicije“

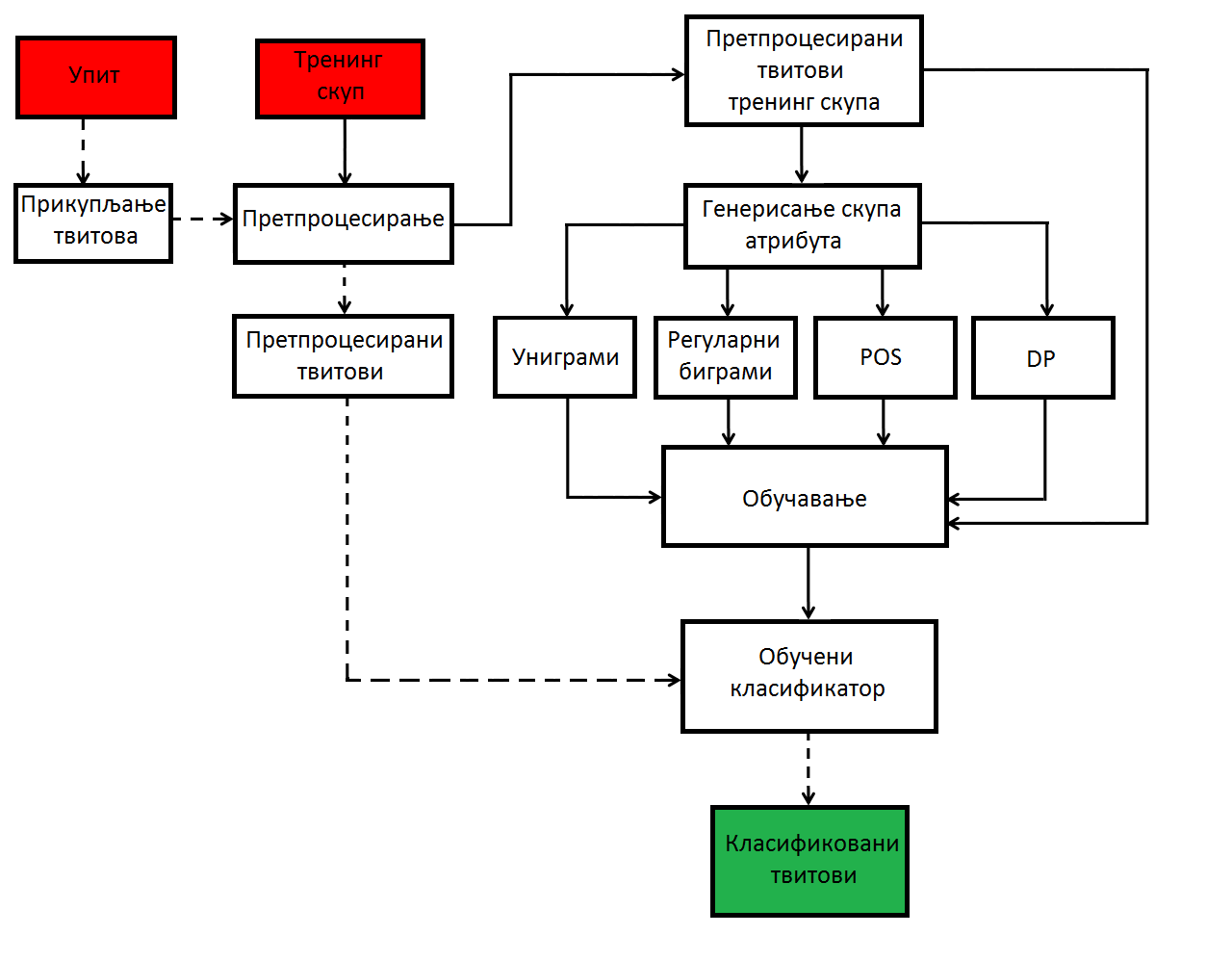
## Opis algoritma na koji se vaš rad oslanja

## Opis algoritma na koji se vaš rad oslanja

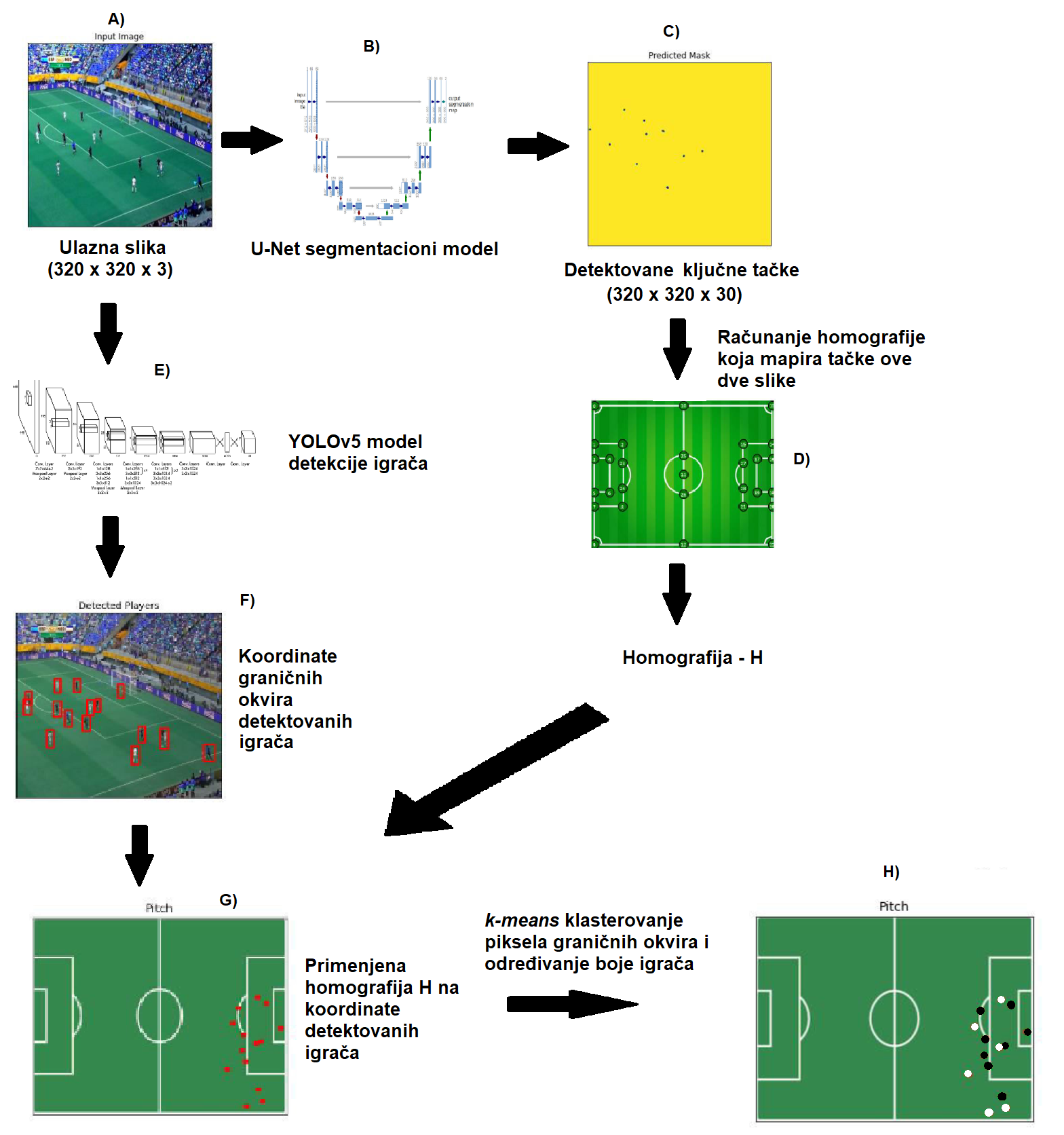
# 

# METODOLOGIJA

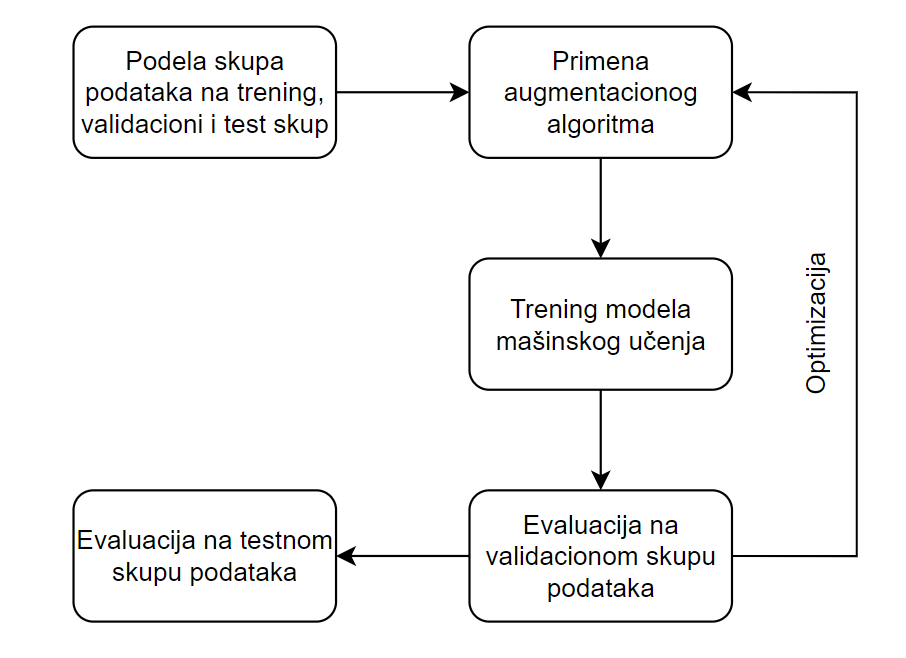
* Krenite sa pasusom koji navodi šta je prikazano u ovom poglavlju. Na primer, „U ovom poglavlju je predstavljena implementacija sistema za...“. Jasno istaknite šta je ulaz u sistem, a šta očekivani izlaz iz sistema.
* U narednom pasusu predstavite arhitekturu rešenja na visokom nivou apstrakcije (npr. dijagram toka podataka u sistemu). Primeri 5.1 - 5.4 daju ideje za predstavljanje arhitekture rešenja. Ovaj dijagram toka podataka podelite na module. Svaki modul razradite u tekstu i, ako je potrebno, prikažite i njegove delove grafički.
* Pojedinačni moduli zahtevaju opsežna objašnjenja, poglavlje „Metodologija“ možete izdeliti na potpoglavlja koja odgovaraju datim modulima. Ako ovo radite, ne zaboravite da dodate rečenicu koja najavljuje ta potpoglavlja.
* Prilikom opisa rešenja ne zaboravite da opišete sledeće stvari (ako ste ih primenili):
  + Kako ste pretprocesirali podatke
  + Kako ste vršili augmentaciju podataka
  + Arhitekturu neuronske mreže koju ste koristili uz obrazloženje zašto ste je odabrali
  + Pomoću koje funkcije gubitka se optimizuje model
  + Koje ste sve kombinacije model/pretprocesiranje isprobali
  + Na koji način ste u svoje rešenje uključili tehnologije predstavljene u poglavlju 4



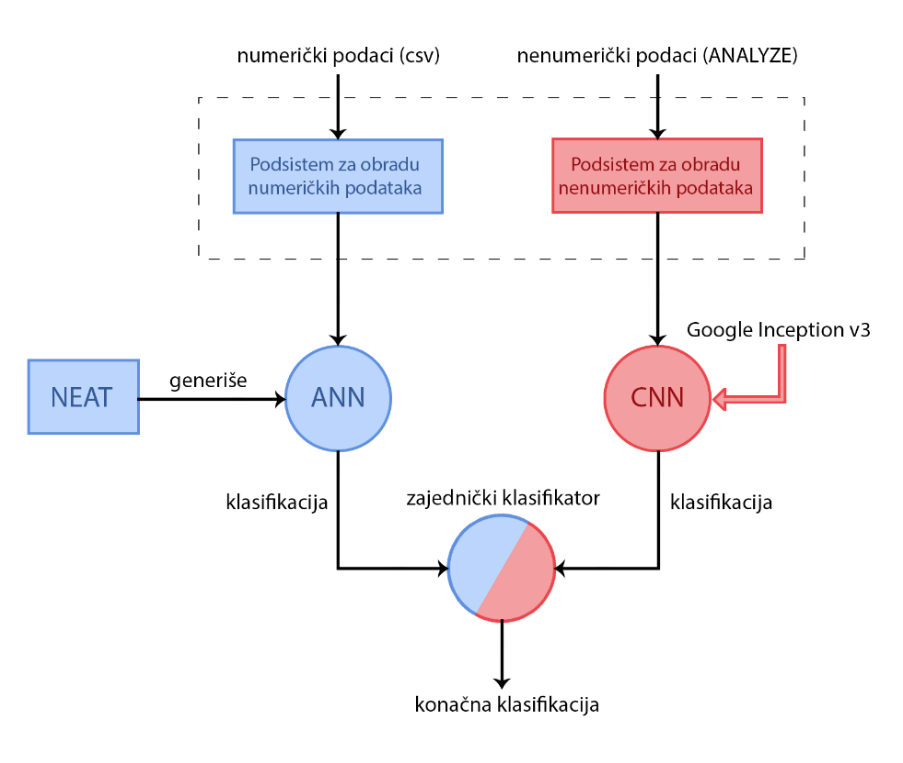
Primer 5.1 Šematski prikaz postupka klasifikacije tvitova.



Primer 5.2 Ulazna slika (A) se prosleđuje U-Net modelu (B) koji za izlaz daje matricu koja predstavlja detektovane tačke koje su vizualizovane na slici (C). Mapiranjem tačaka na njihove ciljne koordinate sa slike (D), dobijamo homografiju H. Na sliku (A) se primenjuje YOLO (E) model za detekciju igrača (F). Primenom homografije na igrače dobijamo sliku (G). K-*means* klasterovanjem piksela dobijamo boje igrača za konačni prikaz (H).



Primer 5.3 Dijagram toka eksperimenta.



5.4 Klasifikacija pacijenata na osnovu: (1) kliničkih testova kojima se detektuje demencija (numerički podaci) i (2) MRI snimaka pacijenta (nenumerički podaci).

## Prvi modul sistema (izmeniti naslov da bude specifičan za vaše rešenje)

* Krenite od toga šta je cilj modula, koji su očekivani izlazi i ulazi.
* Nakon toga specificirajte kako je ovaj cilj implementiran.

## Drugi modul sistema

## Korišćeni alati

Ovo je opciono potpoglavlje koje specificira hardver, biblioteke, alate i programske jezike na koje ste se oslanjali prilikom implementacije rešenja. Alternativno (ako nema puno sadržaja), detalje implementacije možete integrisati u odgovarajuća poglavlja koja opisuju module sistema.

# EKSPERIMENTI

**Cilj poglavlja**: U ovom poglavlju potrebno je opisati sve eksperimente koji su izvršeni u okviru diplomskog rada.

**Važno:** Postavka svakog eksperimenta trebalo bi da sadrži sve detalje koji su potrebni da bi neko drugi mogao da reprodukuje eksperiment.

**Razlika između poglavlja “Eksperimenti” i „Metodologija“**: u metodologiji opisujete arhitekturu sistema koji predstavljate u radu i detalje svakog od njegovih delova modula. Nasuprot toga, u ovom poglavlju opisujete na koji ste način(e) evaluirali taj sistem i/ili svaki od njegovih delova. Na primer, sistem ste možda evaluirali na više različitih skupova podataka ili sa više različitih kombinacija parametara ili ste hteli da vidite uticaj pred ili post-procesiranja pa ste evaluaciju radili sa i bez tih komponenti, itd.

**Organizacija:** U glavnom delu iskažite cilj poglavlja „Eksperimenti“ i pojasnite njegovu podelu na potpoglavlja koja je, tipično, ovakva:

1. Obično, eksperimenti u AI oblasti uključuju jedan ili više obučavajućih skupova. Predstavite ih u posebnom potpoglavlju „Skupovi podataka“.
2. Tipičan sledeći deo eksperimentalne postavke je opis jednog ili više modela mašinskog učenja. Napravite posebno potpoglavlje za svaki eksperiment koji ste izvršili, npr. „Eksperiment 1“, „Eksperiment 2“ (predložite specifičnija imena ako imaju smisla).
3. Opis postupka evaluacije.

**Napomena:** trudite se da ne ponavljate informacije – prvo utvrdite šta je zajedničko za sve vaše eksperimente. Tipično su to skupovi podataka i evaluacija, pa onda možete izdvojiti u dva potpoglavlja, dok ostale detalje pišete za svaki eksperiment posebno u potpoglavljima „Eksperiment 1“, „Eksperiment 2“, itd.

## Skup(ovi) podataka

**Cilj potpoglavlja:** opis skupova podataka korišćenih u svim eksperimentima.

**Organizacija:** Ako imate puno skupova podataka, možete ovo poglavlje podeliti na potpoglavlja koja im odgovaraju. Ako skupovi podataka imaju duža imena, u poglavlju 6.1 uvedite skraćenice. Na primer, K1 (korpus 1).

**Tipični podaci koje opis skupa podataka treba da sadrži:**

1. količina podataka,
2. distribucija klasnog obeležaja,
3. detalji vezani za prikupljanje i anotaciju (labeliranje),
4. da li ste skup podataka preuzeli (da li je javno dostupan, odakle ste ga preuzeli) ili sami kreirali za potrebe diplomskog rada,
5. detalji specifični za vaš skup podataka:
   * Na primer, ako su u pitanju slike onda: veličina, rezolucija, format, kvalitet slika, itd.
   * Na primer, ako je u pitanju tekst onda: prosečna dužina rečenica, dokumenata, entiteta, itd.
6. detalji skupa podataka značajni za problem koji rešavate:
   * Razmišljajte o tome koji detalj skupa vam je potreban da biste u diskusiji objasnili neki rezultat eksperimenta.
   * Na primer, za slike, rezultati mogu biti loši jer je rezolucija slika niska (i onda kod tih rezultata referencirate ovo poglavlje).
   * Na primer, za tekst, rezultati mogu biti loši jer je određena vrste entiteta slabo zastupljena u obučavajućem skupu, rečenice su previše dugačke, itd.

## Eksperiment 1

**Cilj:** prikazujete detalje vezane za ML model(e). Sam model je opisan u prethodnim poglavljima („Teorijske osnove“ i/ili „Metodologija“), dok je ovde cilj da opišete detalje koje ste podesili baš za eksperiment sa ciljem da eksperiment može da se kompletno reprodukuje.

Na primer, ovde prikazujete:

* Vrednosti hiper-parametara. Ako ih ima puno, navedite ih u tabeli.
* Način na koji ste odredili vrednosti hiper-parametara.
* Ako u vašem eksperimentu poredite više modela, onda ovde opisujete konkretne postavke za svaki takav model.
* Ako je cilj vašeg eksperimenta da utvrdite optimalne hiper-parametre ili skup atributa, onda je fokus na opisu načina na koji to radite.
* Hardver ili *cloud* servis koji je korišćen, itd.

## Evaluacija

**Cilj:** prikaz načina evaluacije rešenja. Tipično, ovo obuhvata:

1. Kako ste formirali skup podataka za evaluaciju:

* kako je formiran test skup,
* odnos udela podataka u *test/train* podeli,
* koja vrsta uzorkovanja (*sampling*) je upotrebljena pri podeli na *train*/*test* (npr. *random* ili *stratified*)
* ako je upotrebljena unakrsna validacija, opišite detalje vezane za taj postupak.

1. Mere performansi koje koristite za evaluaciju. Ako su mere neuobičajene i kompleksne onda bi bilo dobro da ih detaljnije objasnite na primeru. Jedan primer takve mere je *Intersection over Union* (IoU) koja se koristi za evaluaciju sistema za detekciju objekata.

# REZULTATI (I DISKUSIJA)

**Cilj:** prikazujete rezultate eksperimenata opisanih u prethodnom poglavlju.

**Napomena oko naslova poglavlja**: ponekad eksperimenti i rezultati mogu biti obimni i/ili zanimljivi pa se o njima može napisati duža diskusija. U tom slučaju je potrebno diskusiju odvojiti u posebno poglavlje „Diskusija“ koje ide nakon rezultata. U nastavku ovog teksta opisani su detalji vezani za poglavlje „Rezultati“, pa onda za poglavlje „Diskusija“, a na vama je da odlučite da li ćete ih pisati u jednom ili u dva odvojena poglavlja.

**Sadržaj poglavlja:**

1. Ovo poglavlje tipično sadrži jednu ili više tabela sa rezultatima eksperimenata. Najbolje je da tabele prikazujete redom po eksperimentima i onda na kraju da date jednu sumarnu tabelu ako to ima smisla. Na primer, ako eksperimenti uključuju evaluaciju nekoliko različitih ML modela, onda u sumarnoj tabeli pokazujete sve njih zajedno i komentarišete kako se porede.
2. Nakon prikaza tabele napišete jedan ili više pasusa u kojima ukratko saopštite i komentarišite rezultate.
   1. Ne ponavljate celu tabelu u tekstu, već istaknite nabolje, najlošije rezultate.
   2. Komentarišite da li su ovi rezultati očekivani (u skladu sa dosadašnjom literaturom) ili iznenađujući.
   3. Na primer, model X se pokazao kao najbolji, što je i očekivano jer je to SOTA model za ovaj problem ili rezultati za klasu X su najbolji, što je iznenađujuće jer je to klasa koje jako slabo zastupljena u obučavajućem skupu.
3. Važno je da donesete odluku o tome da li ćete diskusiju rezultata uključiti u ovo poglavlje ili ćete ih predstaviti u posebnom poglavlju „Diskusija“.
   1. Ako diskusije nema puno, onda ovde dajte svoja objašnjenja za rezultate. U tom slučaju, u nastavku kao posebne pasuse možete dodati delove poglavlja „Diskusija“ (sadržaj poglavlja „Diskusija“ je objašnjen u poglavlju 8).
   2. Razloge zašto su rezultati takvi kakvi su često nije lako utvrditi. Zato, pri diskusiji pišite „pretpostavljamo da su rezultati takvi zato što...“ umesto da čvrsto „tvrdimo da su rezultati...“.

Preporuke oko predstavljanja sadržaja:

* Koristite tabele, a ne grafikone jer se iz tabela vide tačne vrednosti mera performansi. Pored tabela možete imati i grafikon koji je vizuelno predstavlja. Razmislite detaljno o tome šta je to što će on moći da prikaže, a tabela neće.
* Tabele ne bi trebalo da budu previše velike. Uvek je bolje da prikažete više manjih preglednih tabela nego jednu veliku nepreglednu.
* Podebljajte (*bold*) najbolje rezultatе u tabeli.
* Nivo detalja prikaza rezultata bi trebalo da bude što veći. Na primer, ako radite više-klasnu klasifikaciju trebalo bi da pored sumarnih mera (npr. makro F-mera) date i preciznost, odziv i F-meru za svaku klasu posebno.
* Ako imate preveliki broj klasa da biste rezultate prikazivali za svaku klasu posebno (preko 10 ili 15), onda rezultate po klasama možete opisati tako što ćete ih grupisati. Na primer, klase 1,2 i 8 imaju F-mere u rasponu od X do Y, dok klase 3,5,9 imaju....itd.
* Trudite se da sumarne mere što realnije sumiraju rezultate vašeg sistema. Na primer, za više-klasnu klasifikaciju makro F-mera je dobar izbor sumarne mere. Dakle, pored ili u okviru tabele sa rezultatima po klasama možete dati i makro F-meru kao sumarnu.

# DISKUSIJA

**Cilj:** prikazujete svoje rezultate iz svih uglova važnih za cilj vašeg rada.

* Na primer, ako je cilj vašeg rada bio da kreirate novi sistem za neki problem onda je važno da diskutujete kako se vaši rezultati porede sa srodnim rešenjima, pogotovo onim najboljim.
* Ako ne postoje srodna rešenja koja baš rešavaju isti problem kao i vi, sigurno postoje rešenja koja rešavaju sličan problem sa kojima se možete uporediti.
* Na primer, razvijate određen NLP sistem za srpski jezik i niko do sada nije radio to baš za srpski jezik. U tom slučaju, poredite se sa sistemima koji to rada za engleski.

**Sadržaj poglavlja**:

* Komentarišite poređenja sa drugim sistemima.
  + Na primer, rezultati prikazani u radu [x] su bolji jer je korišćen bolji hardver ili veći skup podataka, što ovde nije moguće uraditi usled nedostatka *big data* infrastrukture.
  + Ako je teško porediti performanse usled korišćenja različitih skupova podataka i meri performanse, istaknite tu činjenicu.
* Komentarišite realnu upotrebljivost sistema.
  + Da li su performanse na nivou onih koje postižu ljudi?
  + Da li se rezultati direktno mogu koristiti u produkciji ili su tek prvi korak ka nekom boljem sistemu?
  + Ovde je takođe važno da komentarišete razloge za tvrdnje koje ste izneli.
  + Napomena: ponekad je samim anotatorima skupa podataka teško da se dogovore oko anotacije konkretnog primera[[1]](#footnote-1). Ovo može biti uzrok loših performansi modela. Često u srodnim rešenjima postoje poglavlja koja govore o anotaciji skupa podataka gde su date mere (tipično F-mera) koje pokazuju koliko je teško da se više ljudi složi oko anotacije. U tom slučaju se možete porediti sa tim merama.
* Diskutujte prednosti i ograničenja svog sistema.
  + Na primer, prednost je da je sistem lako proširiv, ali je trenutno ograničenje da radi samo za srpski jezik.
  + Važno je da budete realni, nemojte sakrivati nedostatke. Svrha ovog dela je upravo to da pokažete da ste svesni ograničenja svog sistema.
  + Dajte predloge kako bi se ograničenja mogla prevazići. Na primer, bolji hardver, veći skup podataka, itd.
* (Opciono) u ovo poglavlje uključite i analizu grešaka sistema[[2]](#footnote-2). Za detalje vezane za analizu grešaka pogledajte prateći materijal.

# 

# ZAKLJUČAK

**Cilj**: (1) sumiranje glavnih poenti u radu i diskusija implikacija rešenja i (2) opis mogućih daljih pravaca za unapređenje iznetog rešenja.

**Sadržaj**:

* Sumiranje glavnih poenti u radu:
  + Rešavani problem i motivacija za njegovo rešavanje
  + Opis rešenja na visokom nivou apstrakcije
  + Opis evaluacije rešenja na visokom nivou apstrakcije
  + Podsetiti na sve izvučene zaključke i komentarišite njihovu važnost (glavne poente iz diskusije)
* Iznesite moguće pravce daljeg proširivanja/unapređenja/otklanjanje identifikovanih nedostataka rešenja

|  |
| --- |
| **(Rekapitulacija problema i motivacije za njegovo rešavanje:)** U ovom radu predstavljen je sistem za automatsku detekciju i prepoznavanje saobraćajnih znakova sa slike. Motivacija je bila što bi takav sistem mogao da se koristi prilikom vožnje i da koristi vozaču putem obaveštavanja o važećim saobraćajnim znakovima. Sistem je implementiram u vidu dva odvojena dela: modula za detekciju i modula za prepoznavanje saobraćajnih znakova. Oba modula oslanjaju se na modele konvolucionih neuronskih mreža.  **(Rezime rešenja, načina evaluacije i postignutih rezultata:)**  Modul za detekciju koristi *Faster* R-CNN [4] metod sa *AlexNet* [5] arhitekturom konvolucione mreže. Rezultat evaluacije istreniranog modela je 0.6756 mAP. Za treniranje i testiranje korišćen je skup belgijskih saobraćajnih znakova (engl. BTSD – *Belgium Traffic Sign Dataset*) [6], uz izbacivanje kategorija „nedefinisano“ i „ostalo“, i smanjenje broja negativnih uzoraka.  Modul za prepoznavanje saobraćajnih znakova vrši klasifikaciju znakova dobijenih iz faze detekcije. Za klasifikaciju treniran je model konvolucione neuronske mreže sa *ResNet* [8] arhitekturom od devet rezidualnih blokova. Korišćen je skup nemačkih saobraćajnih znakova za prepoznavanje (engl. GTSRB – *German Traffic Sign Recognition Benchmark*) [9], a preciznost modela nad test skupom iznosi 96.2%.  **(Predlozi za unapređenje rešenja:)** Preciznost konvolucione mreže za prepoznavanje mogla bi se dalje povećati boljim izborom hiper-parametara modela, što bi zahtevalo ponavljanje procesa treniranja. Što se tiče *Faster* R-CNN metoda korišćenog za detekciju, značajno bolje performanse mogu se postići prelaskom na *ResNet* arhitekturu konvolucione mreže. S obzirom na to da *Faster* R-CNN implementacija u korišćenom *Microsoft Cognitive Toolkit* (CNTK) [10] alatu ovo ne podržava, to bi zahtevalo izmenu implementacije ili korišćenje drugog alata. Skup podataka bi takođe mogao da se proširi spajanjem sa nekim drugim, kao što je na primer skup nemačkih saobraćajnih znakova za detekciju (engl. GTSDB – *German Traffic Sign Detection Benchmark*) [32]. |

Primer 9.1 Primer celokupnog zaključka uz komentare

# LITERATURA

|  |
| --- |
|  |
| 1. Manresa, C., Varona, J., Mas, R. and Perales, F.J., 2005. Hand tracking and gesture recognition for human-computer interaction. *ELCVIA Electronic Letters on Computer Vision and Image Analysis*, *5*(3), pp.96-104. 2. Wang, R.Y. and Popović, J., 2009. Real-time hand-tracking with a color glove. *ACM transactions on graphics (TOG)*, *28*(3), pp.1-8. 3. Sharp, T., Keskin, C., Robertson, D., Taylor, J., Shotton, J., Kim, D., Rhemann, C., Leichter, I., Vinnikov, A., Wei, Y. and Freedman, D., 2015, April. Accurate, robust, and flexible real-time hand tracking. In *Proceedings of the 33rd annual ACM conference on human factors in computing systems* (pp. 3633-3642). 4. Melax, S., Keselman, L. and Orsten, S., 2013, March. Dynamics based 3D skeletal hand tracking. In *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games* (pp. 184-184). 5. Zhang, F., Bazarevsky, V., Vakunov, A., Tkachenka, A., Sung, G., Chang, C.L. and Grundmann, M., 2020. Mediapipe hands: On-device real-time hand tracking. *arXiv preprint arXiv:2006.10214*. 6. Yeo, H.S., Lee, B.G. and Lim, H., 2015. Hand tracking and gesture recognition system for human-computer interaction using low-cost hardware. *Multimedia Tools and Applications*, *74*(8), pp.2687-2715. 7. Maung, T.H.H., 2009. Real-time hand tracking and gesture recognition system using neural networks. *International Journal of Computer and Information Engineering*, *3*(2), pp.315-319. 8. Lugaresi, C., Tang, J., Nash, H., McClanahan, C., Uboweja, E., Hays, M., Zhang, F., Chang, C.L., Yong, M.G., Lee, J. and Chang, W.T., 2019. Mediapipe: A framework for building perception pipelines. *arXiv preprint arXiv:1906.08172*. 9. Halder, A. and Tayade, A., 2021. Real-time vernacular sign language recognition using mediapipe and machine learning. *Journal homepage: www. ijrpr. com ISSN*, *2582*, p.7421. 10. Duy Khuat, B., Thai Phung, D., Thi Thu Pham, H., Ngoc Bui, A. and Tung Ngo, S., 2021, February. Vietnamese sign language detection using Mediapipe. In *2021 10th International Conference on Software and Computer Applications* (pp. 162-165). 11. Le, T.H., Jaw, D.W., Lin, I.C., Liu, H.B. and Huang, S.C., 2018, May. An efficient hand detection method based on convolutional neural network. In *2018 7th International Symposium on Next Generation Electronics (ISNE)* (pp. 1-2). IEEE. |
|  |  |

# BIOGRAFIJA

Ovde navedite osnovne informacije o sebi koje će predsednik komisije pročitati prilikom otvaranja odbrane završnog/diplomskog rada (primer 11.1 ).

|  |
| --- |
| Petar Petrović je rođen dd.mm.gggg. u Novom Sadu, gde je stekao svoje osnovno i srednje obrazovanje. Školske xxxx/yy godine se upisuje na Fakultet tehničkih nauka na studijski program X. Položio je sve ispite predviđene planom i programom i stekao uslov za odbranu završnog rada. |

Primer 11.1 Primer biografije

1. Na primer, cilj nam je da razvijemo AI model čiji je cilj da detektuje da li je određena Java klasa previše kompleksna (pati od tzv. *God Class code smell*-a). Samim programerima (anotatorima) je teško da usaglase stavove oko toga koje klase su „previše kompleksne“. Stoga skup podataka može imati nekonzistentne anotacije, čineći da obučeni ML model ima lošije performanse. [↑](#footnote-ref-1)
2. Analizu grešaka sistema možete opciono staviti i u poglavlje „Rezultati“. U tom slučaju, u diskusiji kratko prodiskutujte rezultate te analize. [↑](#footnote-ref-2)