Analiza videozapisa nogometnih utakmica tehnikama računalnog vida

Pavić, Tihomir

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 660

ANALIZA VIDEOZAPISA NOGOMETNIH UTAKMICA TEHNIKAMA RAČUNALNOG VIDA

Tihomir Pavić

Zagreb, lipanj 2024.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 660

ANALIZA VIDEOZAPISA NOGOMETNIH UTAKMICA TEHNIKAMA RAČUNALNOG VIDA

Tihomir Pavić

Zagreb, lipanj 2024.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

Zagreb, 4. ožujka 2024.

DIPLOMSKI ZADATAK br. 660

Pristupnik:	**Tihomir Pavić (0036519545)**
Studij:	Računarstvo
Profil:	Znanost o podacima
Mentor:	izv. prof. dr. sc. Tomislav Hrkać

Zadatak: **Analiza videozapisa nogometnih utakmica tehnikama računalnog vida**

Opis zadatka:

Suvremene tehnike računalnoga vida pružaju mogućnost automatske analize videosnimaka nogometnih i općenito sportskih natjecanja, te automatiziranog dobivanja vrijednih informacija o značajnim događajima u susretu (gol, zaleđe, kretanje igrača ili lopte i slično). Izvlačenje takvih informacija može pridonijeti poboljšanju kvalitete i brzine sudačkih odluka, mogućnosti praćenja statistika igrača ili kao pomoć u taktičkom smislu. U okviru diplomskoga rada potrebno je proučiti najznačajnije tehnike računalnog vida i metode dubokog učenja primjenjive za analizu videozapisa nogometnih utakmica. Programski ostvariti sustav koji će na temelju ulaznog videozapisa nogometnog susreta generirati odabrane informacije o snimljenom susretu. Pripremiti bazu videozapisa za učenje i ispitivanje sustava, analizirati ponašanje ostvarenog sustava te prikazati i ocijeniti ostvarene rezultate. Radu priložiti izvorni i izvršni kod razvijenih postupaka, ispitne podatke i rezultate, uz potrebna objašnjenja i dokumentaciju te navesti korištenu literaturu.

Rok za predaju rada: 28. lipnja 2024.

Zahvaljujem mentoru izv. prof. dr. sc. Tomislavu Hrkaću na pomoći pri izradi ovog rada i na pružanju ove jedinstvene prilike.

Zahvaljujem obitelji i prijateljima koji su uvijek bili uz mene na ovom putovanju do diplome, a posebno Kristini na stalnoj vjeri u mene, posebnoj podršci i razumijevanju tijekom cijelog studija.

Sadržaj

Uvod

Nogomet je jedan od najpopularnijih sportova diljem svijeta te svakog dana privlači veliku pažnju brojnih sportskih fanova. Tijekom godine organizirana su brojna nogometna natjecanja od kojih su najpopularnija europska i južnoamerička natjecanja, međutim u posljednjih nekoliko godina veliku pažnju izazivaju i natjecanja s ostalih područja poput Saudijske Arabije i Sjedinjenih Američkih Država. S obzirom na količinu utakmica i natjecanja, javlja se i bogat izvor podataka koji je predstavlja ključ za razvoj sustava umjetne inteligencije tj. računalnog vida i dubokog učenja.

Računalni vid je grana umjetne inteligencije koja se bavi prikupljanjem, obradom te razumijevanjem slika ili videozapisa na način kako to vide ljudi. Nastoje se razviti algoritmi i tehnike koje računalima omogućuju da interpretiraju vizualne informacije iz digitalnih slika ili videozapisa. Postoji mnogo problema koje računalni vid nastoji riješiti. Neki od tih problema su detekcija objekata na slici, praćenje objekata sa videozapisa, klasifikacija slika ili videozapisa u definirane razrede, klasifikacija svakog piksela slike u zaseban razred (semantička segmentacija), povećanje rezolucije slika ili videozapisa pa sve do generiranja umjetnih slika koje izgledaju veoma realno.

Tijekom prethodnog desetljeća, područje dubokog učenja doživjelo je velik napredak te se danas koristi u gotovo svim spomenutim problemima koje računalni vid nastoji riješiti. Duboko učenje i neuronske mreže svojom ekspresivnošću omogućuju učenje složenih obrazaca i značajki iz velikih skupova podataka što ima izuzetan utjecaj u području računalnog vida jer otvaraju mogućnost razumijevanja slika i videozapisa na način na koji to razumiju ljudi [12].

Razumijevanje videozapisa jedan je od najvećih izazova računalnog vida posljednjih godina. S obzirom na količinu videozapisa koje pruža nogometno podneblje, upravo u tom području pokrenuta su brojna istraživanja u razumijevanju videozapisa nogometnih utakmica što bi moglo donijeti velike napretke u strategijskim analizama, pogreškama tijekom igre pa čak i donošenju sudačkih odluka. S druge strane, razumijevanje nogometnih akcija pomoću računalnog vida moglo bi uvelike pomoći brojnim medijskim kućama prilikom automatskog generiranja sažetaka utakmica.

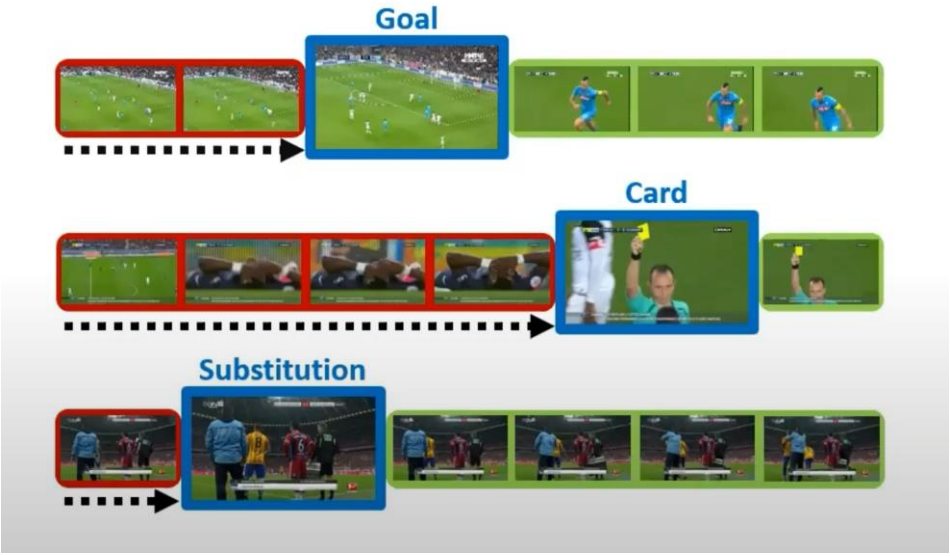
Ovaj rad bavi se upravo problematikom detektiranja glavnih nogometnih akcija koje se mogu dogoditi na utakmici poput gola, kartona, zamjene, kaznenog udarca i druge. Cilj je unutar videozapisa predvidjeti trenutak u kojem se određena akcija dogodila te odrediti koja je to akcija kako bi se iz cijelog videozapisa mogao generirati sažetak na temelju detektiranih akcija. Bez obzira na dostupnost videozapisa utakmica, popriličan je izazov pronaći označeni skup podataka nad kojim bi se modeli dubokog učenja učili te su napravljena brojna istraživanja upravo s ciljem prikupljanja i označavanja skupa podataka koji bi mogli doprinijeti daljnjem napretku računalnog vida u nogometu. Upravo tako je nastao SoccerNet [1], veliki skup podataka namijenjen razumijevanju nogometnih videozapisa. Skup podataka sadrži više od 500 snimljenih nogometnih utakmica te pruža oznake videozapisa ovisno o zadatku koji se želi raditi poput lokalizacije terena, praćenja igrača, klasifikacija brojeva na dresovima, prepoznavanje akcija lopte i slično. Također, SoccerNet [1] nudi oznake za prepoznavanje nogometnih akcija kojim se bavi ovaj rad što omogućava izradu i testiranje modela koji će biti korišteni u ovom radu.

U poglavlju (1) bit će razmotreni elementarni koncepti problema detektiranja akcija u videu. U poglavlju (2) detaljno opisan korišteni skup podataka u ovom radu. Nakon skupa podataka, u poglavlju (3) opisana je cijela metodologija primijenjena za detektiranje nogometnih akcija. Na samom kraju u poglavlju (4) opisan je eksperimentalni proces i rezultati metode.

1. Elementarni koncepti

1.1. Prepoznavanje akcija tehnikama računalnog vida

Prepoznavanje akcija (engl. *action spotting*) u videu područje je računalnog vida kojem je danas posvećeno mnogo pažnje, a cilj je identificiranje i lokalizacija specifičnih radnji unutar video sekvence. Ovaj zadatak ima brojne primjene kao što su video nadzor, analiza sportskih događaja, autonomna vožnja, sažimanje videa i drugo. Kao primjer možemo razmotriti izazov kojim se bavi ovaj rad, a to je detekcija specifičnih nogometnih akcija na video snimkama utakmica s ciljem sažimanja cijele utakmice u video koji sadrži samo odabrane akcije. Ako se radi o dužem videu, potrebno ga je izrezati u manje dijelove te za svaki dio odrediti sadrži li on neku od odabranih akcija. Ako sadrži, potrebno je odrediti točan vremenski segment željene akcije u tom dijelu videa koji se procesira. Na slici (Slika 1.1) možemo vidjeti primjer tri kraća video isječka koji sadrže akciju koju bismo željeli detektirati.



Slika 1.1 Primjer video sekvence koja sadrži akciju [2]

Na temelju slike možemo izdvojiti i određene izazove koje zadatak prepoznavanja akcije nosi, a to je da se ne radi samo o prepoznavanju akcije, već i vremenska lokalizacija koja može biti poprilično zahtjevna. Akcija se u video sekvenci može nalaziti u bilo kojem

vremenskom trenutku te je bitno biti robustan na otkrivanje akcije na različitim mjestima videa. Računalo vidi video kao slijed slika koje je potrebno izdvojiti ovisno o tome koliko slika po sekundi želimo prikazati. Također, izazov je i razumijevanje sadržaja videa jer je potrebno razumjeti niz slika kako bi se odredilo sadrži li uopće sekvenca videa željenu akciju.

1.2. Metode i pristupi problemu prepoznavanja akcije

Razvijeni su brojni pristupi rješavanju ovog problema poput nekih tradicionalnih metoda koji su se oslanjali na ručno kreirane značajke i statističke modele za prepoznavanje radnji. Tehnike poput optičkog toka, histograma gradijenata, detektiranje ključnih točaka teško su se nosile s kompleksnim radnjama i općenito razumijevanjem situacije videa. Nadalje, pristupi koji su donijeli velike napretke su pristupi temeljeni na dubokom učenju. Kao i u svim područjima računalnog vida, duboko učenje donijelo je revoluciju te se razvija i napreduje iz dana u dan. Prvi pristup koji je donio značajan napredak bio je uvođenje konvolucijskih neuronskih mreža koje imaju ekspresivnost pronalaska i izdvajanja relevantnih značajka iz slike što predstavlja ključ za obradu kompleksnih i visoko dimenzionalnih podataka poput slika. Konvolucijski slojevi su ti koji prolaskom filtera po slici, radeći operaciju konvolucije, kreiraju mape značajki. Svaki filter ima mogućnost uhvatiti drugačije vizualne uzorke poput rubova, tekstura pa i kompleksnijih struktura u dubljim slojevima. Upravo mape značajki koje konvolucijska neuronska mreža kreira, mogu se iskoristiti za sekvencijalnu obradu kako bi se zaključile vremenske informacije koje su nužne kod prepoznavanja akcije u videu.

Nadalje, uz konvolucijske neuronske mreže koje tradicionalno rade operaciju konvolucije nad prostornim dimenzijama, s obzirom da kod problema prepoznavanja akcija imamo i vremensku dimenziju (slijed slika u vremenu koje čine video isječak), nameće se ideja 3D konvolucijskih neuronskih mreža koje proširuju operaciju konvolucije na vremensku dimenziju.

Također, još jedna od sve popularniji metoda u računalnom vidu su transformerski modeli koji koriste mehanizam pažnje što se može iskoristiti na način da se model fokusira na relevantne dijelove video sekvence, poboljšavajući sposobnost prepoznavanja i lokalizacije akcije u video sekvenci.

1.3. Prepoznavanje nogometnih akcija iz videosnimaka nogometnih utakmica

Glavni cilj ovog rada je prepoznavanje nogometnih akcija iz videosnimke cijele nogometne utakmice kako bi se kreirao video sažetak koji sadrži samo ključne i najzanimljivije trenutke utakmice. S obzirom da jedna nogometna utakmica traje oko 90 minuta, potrebno je osmisliti način na koji će se procesirati tako veliki videozapis. S obzirom da se ključne nogometne akcije ne događaju često unutar utakmice, nameće se ideja procesiranja uzastopnih kraćih sekvenci videa dok se ne prođe cijela utakmica. Time se zapravo pokušava odrediti postoji li tražena nogometna akcija unutar svake video sekvence te ako postoji u kojem trenutku se ona dogodila. Ako video sekvenca ne sadrži jednu od željenih akciju, tada bismo ju trebali označiti kao klasu pozadine i nastaviti s procesiranjem sljedeće video sekvence. Detaljno objašnjenje obrade i analize utakmica i video sekvenci bit će prikazano u drugom poglavlju ovog rada.

Metoda koja se koristi u ovom radu inspirirana je radom „Regresssion and Masking for Soccer Event Spotting“ [2]. U srcu pristupa su konvolucijske neuronske mreže koje omogućuju razumijevanje scene te detektiranje željene akcije unutar predane video sekvence. Kao što je spomenuto, video utakmice je razbijen na video sekvence u trajanju od 20 sekundi. Svaka video sekvenca najprije se predaje na ulaz konvolucijske neuronske mreže koja služi kao ekstraktor značajki. U slučaju ovog rada radi se o arhitekturi ResNet-152 čije težine su predtrenirane na poznatom ImageNet skupu podataka [7]. S obzirom da je video niz slika, potrebno je odrediti koliko slika će se predati na ulaz modela unutar sekvence od 20 sekundi. U slučaju ovog rada, izdvajaju se dvije slike po sekundi (engl. *frames per second*), čime sekvencu koja dolazi na ulaz neuronske mreže čine 41 slijednih slika. Značajke koje uzimamo su značajke dobivene iz zadnjeg sloja globalnog sažimanja arhitekture ResNet-152 gdje za svaku sliku dobivamo 1D vektor od 2048 značajki. Dodavši tome vremensku dimenziju, tj. sekvencu od 41 slika, vektor značajki koji dobivamo ima dimenzije (41, 2048). Nakon što su značajke izdvojene slijedi nova konvolucijska mreža čiji je zadatak prepoznati postoji li nogometna akcija u sekvenci i gdje se ona nalazi. Dakle, ideja je uzeti značajke koje su dobivene ekstrakcijom, poslati ih na ulaz nove neuronske mreže koja treba klasificirati svaku video sekvencu ovisno o prisutnosti nogometne akcije te predvidjeti relativnu vremensku poziciju unutar predane video sekvence. Upravo tu ideju i samu arhitekturu te dodatne neuronske mreže predložili

su autori članka „Regresssion and Masking for Soccer Event Spotting“ [2]. Ipak, ovaj pristup ima jedno bitno ograničenje, a to je da može detektirati prisutnost jedne akcije unutar video sekvence što u nogometu ne mora biti slučaj. Detalji implementacije i izvedbe cijelog sustava opisani su u poglavlju (3) ovog rada.

Izuzetno veliku ulogu u detektiranju nogometnih akcija ima skup podataka. Poprilično je teško pronaći skup podataka s dovoljnom količinom videa koji ima odgovarajuće oznake kako bismo mogli učiti neki od spomenutih modela. Međutim, skup podataka SoccerNet [1] pruža veliku količinu označenih nogometnih utakmica nad kojim je moguće učiti modele dubokog učenja. Također, s obzirom na specifičnost cilja ovog rada i nekih ograničenja metode, potrebno je odraditi određeno procesiranje skupa podataka što će detaljno biti objašnjeno u poglavlju (2) ovog rada. Sam proces obrade podataka i prilagodbe oznaka bio je u velikom fokusu ovog rada kako bi se postigli očekivani i željeni rezultati.

2. Skup podataka

Skup podataka ima jako veliku ulogu u izradi sustava koji koriste duboko učenje. Kako bismo nadzirani model mogli učiti, potrebni su ispravno označeni podaci u prikladnom formatu ovisno o zadatku. U današnje vrijeme nije lagano doći do označenih podataka nogometnih videozapisa jer se radi o velikoj količini materijala koju je potrebno označiti što zahtjeva mnogo vremena i resursa. Dugi niz godina istraživačko područje računalnog vida u nogometu je stagniralo zbog nedostatka podataka. Jedna od prekretnica bio je skup podataka SoccerNet [1] koji pruža veliku količinu označenih nogometnih utakmica za različite zadatke i izazove u području računalnog vida. Neki od zadataka, za koje spomenuti skup podataka pruža označeni skup podataka su: praćenje igrača, prepoznavanje brojeva na dresovima, lokalizacija nogometnog terena, automatsko kalibriranje kamere, praćenje akcija lopte poput dodavanja i druge. Također, SoccerNet [1] pruža označeni skup podataka za prepoznavanje nogometnih akcija, koje su u fokusu ovog rada, što omogućuje razvoj i testiranje korištenih metoda. U nastavku je detaljnije razmotren skup podataka koji je korišten u ovom radu te procesiranje samog skupa.

2.1. Skup podataka za prepoznavanje nogometnih akcija

SoccerNet [1] pruža označeni skup podataka za prepoznavanje nogometnih akcija koji se sastoji od ukupno 500 nogometnih utakmica, čiji videozapisi su podijeljeni u dva poluvremena [3]. Dakle, za svaku utakmicu dolaze dva videozapisa uz jednu datoteku oznaka nogometnih akcija koje su se dogodile. Utakmice koje se nalaze u skupu podataka su utakmica 5 glavnih svjetskih nogometnih liga: Engleske, Francuske, Italije, Španjolske i Njemačke te utakmice natjecanja UEFA Lige prvaka, sve po sezonama 2014/2015, 2015/2016 i 2016/2017. Također, bitno je spomenuti da skup podataka pruža videozapise u dvije rezolucije, 720p i 224p. U ovom radu korišteni su videozapisi u rezoluciji od 224p zbog resursnih ograničenja. Također, dostupne su i izdvojene značajke dobivene ekstrakcijom iz originalnih videozapisa koristeći ResNet-152 model. Okviri (engl. *frames*) su izdvojeni frekvencijom 2 okvira po sekundi. Što se tiče oznaka, skup je označen s ukupno 17 nogometnih akcija: penal *(„Penalty“),* početni udarac *(„Kick-off“)*, gol

(„Goal“), zamjena *(„Substitution“),* zaleđe *(„Offside“),* šut u okvir gola *(„Shots on target“)*, šut izvan okvira gola *(„Shots off target“),* izbijanje lopte *(„Clearance“)*, lopta izvan terena *(„Ball out of play“),* ubačaj rukom *(„Throw-in“),* prekršaj *(„Foul“),* neizravni slobodni udarac *(„Indirect free-kick“),* izravni slobodni udarac *(„Direct free-kick“)*, korner *(„Corner“),* žuti karton *(„Yellow card“)*, crveni karton *(„Red card“)* te drugi žuti karton koji prelazi u crveni karton *(„Yellow->red card“).*

Način na koji su nogometne akcije označene prikazuje slika (Slika 2.1). Dakle, za svaku utakmicu oznake su dane unutar datoteke u .json formatu koji je jednostavan za obradu i programski pristup oznakama. U svakoj datoteci nalazi se nekoliko meta informacija o samoj utakmici, ali ono najbitnije nalazi se unutar ključa *„annotations“.* Kao što je prikazano na slici (Slika 2.1), unutar ključa *„annotations“* nalazi se lista oznaka gdje svaka oznaka sadrži sljedeće informacije:

- *„gameTime“* – sadrži informaciju o poluvremenu utakmice te točnom vremenu u minutama i sekundama u kojem se dogodila označena akcija.
- *„label“* – ime jedne od 17 mogućih akcija koja je označena konkretnom oznakom
- *„position“* – točno vrijeme akcije u milisekundama
- *„team“* – sadrži informaciju o kojem timu se radi. Može poprimiti vrijednosti: „*away“, „home“* ili *„not applicable“*
- *„visibility“* – sadrži informaciju o tome je li akcija u tom trenutku vidljiva na kameri.

Od spomenutih oznaka ključne su *„gameTime“*, *„label“* i *„position“* koje daju informacije koje se žele predvidjeti u ovom radu, a to su:

- Koja akcija se dogodila?
- U kojem trenutku se akcija dogodila?

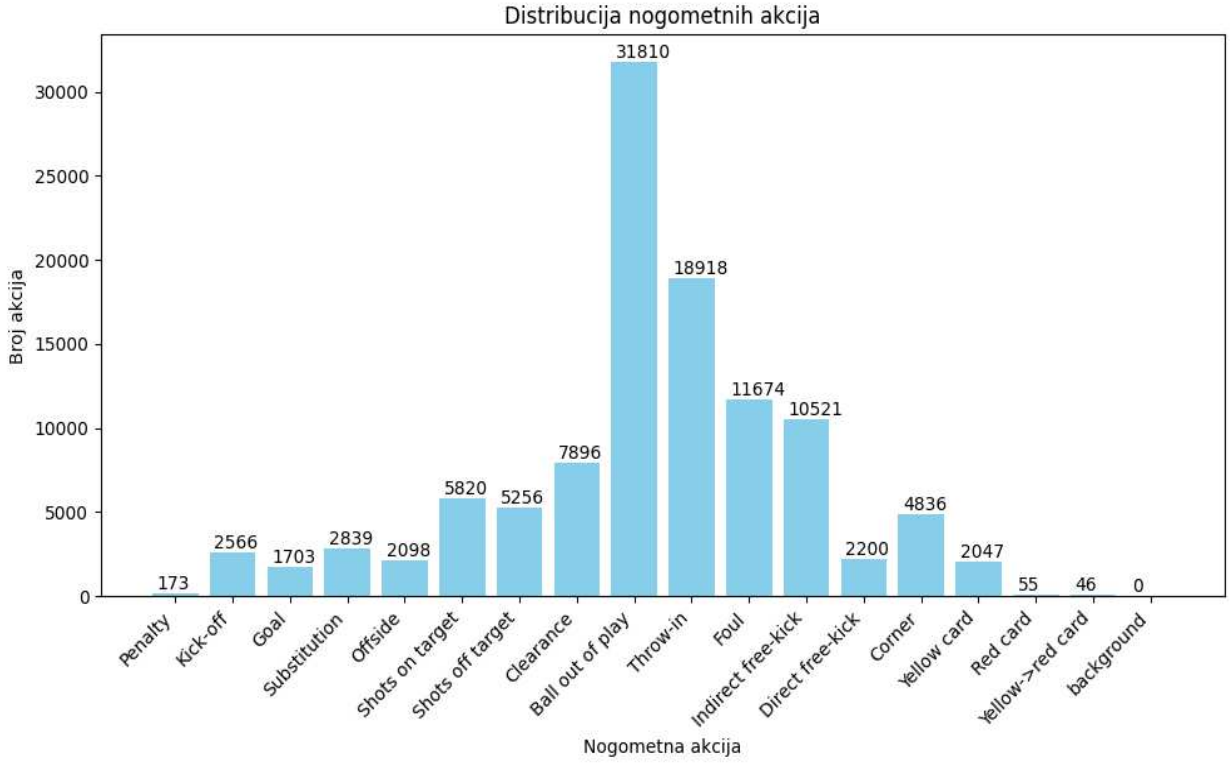
Oznaka „*visibility*“ može biti zbunjujuća, no ponekad postoji situacija da se u videu utakmice prikazuje ponovna snimka prethodno postignutog gola, dok u tom trenu igrači izvode početni udarac sa centra terena pa u tom slučaju oznaka *„Kick-off*“ postaje nevidljiva te *„visibility“* tada poprima vrijednost *„not shown“.*

Slika 2.1 Primjer oznaka nogometnih akcija

Jasno je da se neke oznake pojavljuje češće od drugih što može izazvati nebalansiranost skupa prilikom učenja modela. Recimo, u tijeku utakmice se češće događa prekršaj ili izlazak lopte s terena u odnosu na penal ili crveni karton. Također, postoje akcije koje se mogu pojaviti zaredom u kratkom vremenskom intervalu. Ako imamo oznaku penala, koja predstavlja trenutak izvođenja penala, ukoliko je penal realiziran, oznaka gola dolazi gotovo u istoj sekundi izvođenja penala.

Kako bi se stekao dojam količine pojedinih označenih nogometnih akcija u korištenom skupu podataka, na slici (Slika 2.2) prikazana je distribucija oznaka unutar svih 500 dostupnih utakmica. Iz grafa se može vidjeti kako dominira oznaka *„Ball out of play“,* dok oznake poput *„Red card“, „Yellow->red card“* te *„Penalty“* imaju izuzetno malo podataka što može prouzročiti problem prilikom učenja i prepoznavanja tih akcija. Također, bitno je spomenuti klasu/oznaku *„background“* koja se ne nalazi unutar datoteke,

a njome smatramo sve ostale trenutke utakmice u kojima se nije dogodila niti jedna druga akcija koja se nalazi unutar skupa oznaka.



Slika 2.2 Distribucija oznaka nogometnih akcija

S obzirom da je cilj ovog rada kreiranje sažetaka nogometne utakmice, potrebno je razmotriti neke od ključnih nogometnih akcija koje bi se trebale nalaziti u sažetku. Recimo, iako se nogometna akcija *„Ball out of play“* po distribuciji na slici (Slika 2.2) pojavljuje značajno najviše puta, ta akcija nije ključan događaj koji bi se trebao nalaziti u sažetku. U suprotnom, akcija *„Penalty“* je nešto što može donijeti preokret u utakmici te zapravo i najveća prilika za gol te svakako spada u jednu od ključnih akcija. Međutim, u cijelom skupu podataka ta se akcija dogodila samo 173 puta što je značajno manje u odnosu na akciju *„Goal“* te će biti potrebne tehnike proširenja skupa podataka kako bi model bio u stanju naučiti i prepoznati tu akciju.

U sljedećem poglavlju (2.2) je opisano procesiranje podataka te odabir ključnih nogometnih akcija nad kojim će model biti naučen. Također, objašnjen je postupak promjene oznaka nekih akcija kako bi model mogao na dobar način predvidjeti akciju s obzirom na ranije spomenuto ograničenje metode, a to je mogućnost prepoznavanja samo jedne akcije unutar sekvence od 20 sekundi.

2.2. Priprema podataka

U ovom poglavlju bit će objašnjena cijela priprema podataka i video sekvenci kako bismo lakše razumjeli metodologiju i rad sustava objašnjenog u poglavlju (3). Kako bi sustav bio što točniji pravilna priprema podataka igra veliku ulogu. Zbog spomenutog ograničenja metode i detektiranja samo jedne akcije po sekvenci, najprije ćemo razmotriti odabir i promjenu oznaka koja je napravljena za potrebe ovog rada kako bi metoda postigla što bolje performanse prilikom kreiranja sažetka nogometnog videa.

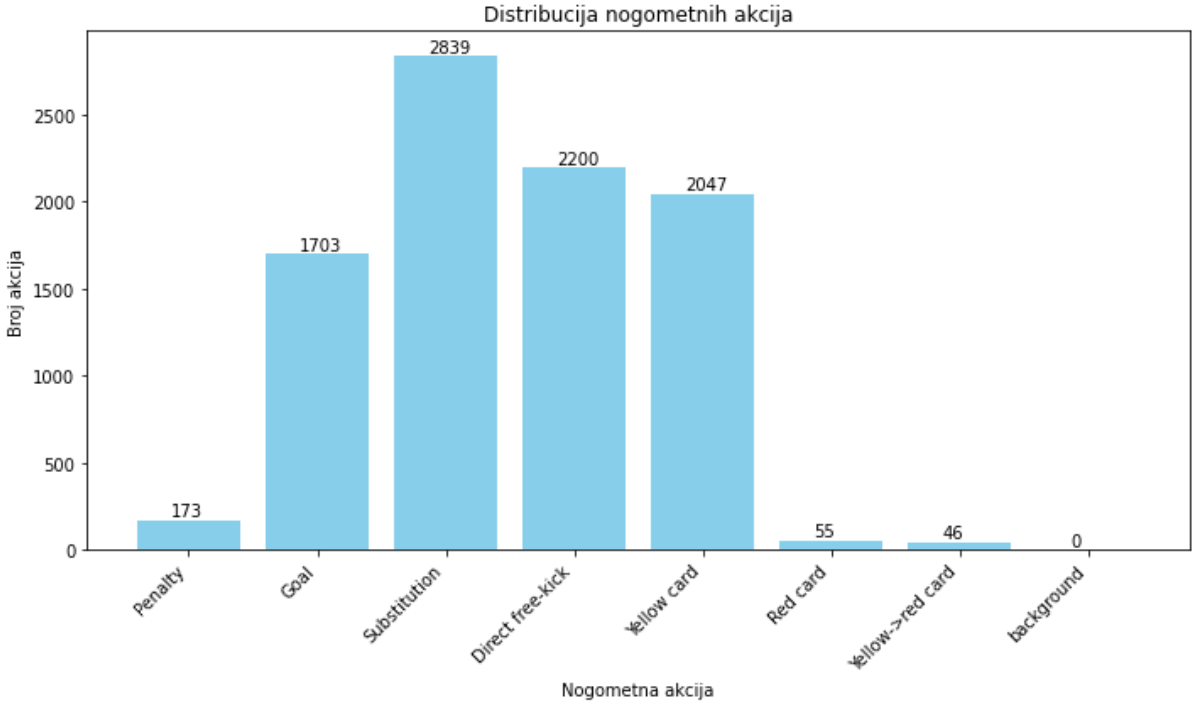
2.2.1. Odabir i procesiranje oznaka nogometnih akcija

Za potrebe ovog rada napravljeno je određeno procesiranje oznaka koje označavaju ukupno 17 nogometnih akcija kroz 500 nogometnih utakmica. Kao što je spomenuto, svako poluvrijeme utakmice sastoji se od posebnog videa koji u prosjeku traje 45 minuta, a svaki taj video najprije će biti podijeljen u video sekvence od 20 sekundi (41 uzastopnih okvira/slika izdvojenih frekvencijom 2 okvira po sekundi). Metoda koja se koristi u ovom radu i detaljno će biti objašnjena u poglavlju (3), zamišljena je na način da neuronska mreža na ulaz dobije upravu tu video sekvencu od 41 okvira te nastoji prepoznati sadrži li ona jednu od traženih nogometnih akcija te u kojem trenutku sekvence se ona dogodila. Naprimjer, ako na ulaz modela pošaljemo sekvencu unutar koje postoji oznaka da je postignut gol u 10. sekundi sekvence, očekivani izlaz modela je klasifikacija te sekvence u klasu *„Goal“* te prepoznavanje da se akcija dogodila točno na pola predane sekvence. S druge strane, ako na ulaz modela dolazi sekvenca unutar koje se ne nalazi niti jedna od željenih akcija, tada je očekivani izlaz modela klasifikacije te sekvence u klasu *„background“*. U oba spomenuta slučaja ne nastaju problemi te metoda kojom se pristupa u ovom radu može raditi na željeni način. Međutim, problem koji se pojavljuje je što se unutar jedne video sekvence može nalaziti više nogometnih akcija koje bi trebalo detektirati, poput *„Penalty“* i *„Goal“*, a metoda kojom pristupamo može detektirati samo jednu akciju. Postavlja se pitanje kako to riješiti te kako prilagoditi dane podatke i oznake.

Postoje 2 konkretna razloga promjene i prilagodbe oznaka u ovom radu:

1. Prethodno spomenuti problem nemogućnosti detekcije više akcija unutar jedne sekvence
2. Nisu sve akcije poželjne u sažetku koji se sastoji od samo ključnih nogometnih akcija

Uzevši u obzir dostupnost oznaka unutar SoccerNet skupa podataka [1], kako bi se u konačnom video sažetku nalazile samo ključne akcije koje su se dogodile tijekom utakmice, potrebno je reducirati skup oznaka na skup koje bi metoda/model trebala prepoznati. Također, smanjenjem skupa oznaka, bitno se smanjuje vjerojatnost pojave više akcija unutar iste sekvence jer ključne nogometne akcije ipak nisu toliko česte, no i dalje će postojati. Taj problem ćemo razjasniti u nastavku. Akcije tj. oznake koje su odabrane kao ključne oznake su: *„Penalty“, „Goal“, „Direct free-kick“, „Substitution“, „Yellow card“, „Red card“ i „Yellow->red card“.*

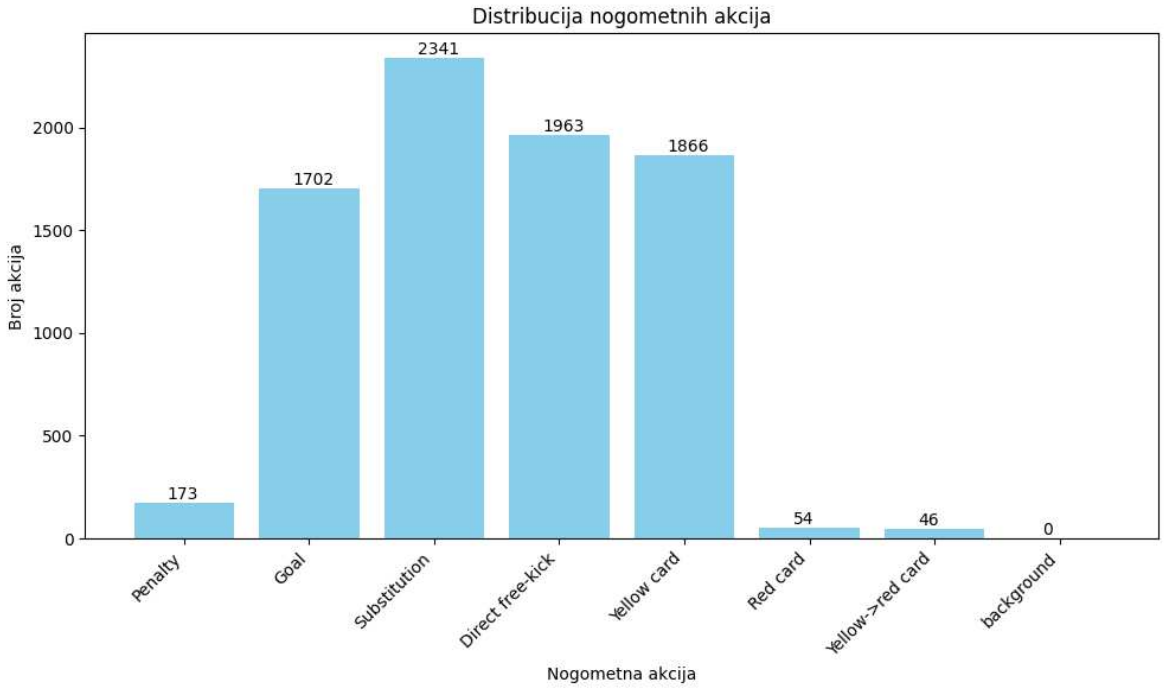


Slika 2.3 Distribucija nogometnih akcija nakon odbacivanja oznaka

Slika (Slika 2.3) prikazuje distribuciju zadržanih nogometnih akcija koje se smatraju ključnim u kreiranju nogometnog sažetka. Ponekad, akcije poput *„Shots on target“*, *„Shots of target“* ili *„Corner“* mogu biti zanimljive, međutim, kako bi što više smanjili

mogućnost pojavljivanja više akcija unutar jedne video sekvence, odlučeno je da se te akcije ne zadržavaju. Dodatni razlog je što kod kreiranja video sažetka, ukoliko je postignut gol iz udarca, akcija gola će se nalaziti u sažetku, a samim time i akcije koje se nalaze oko nje unutar nekog vremenskog intervala. U velikoj većini situacija, to su akcije poput kornera ili upravo udarca prema golu koje prethode akciji gola, ako je postignut iz tih situacija. Bitno je spomenuti da sve oznake koje su odbačene iz skupa oznaka, naprosto više ne postoje te se trenuci u utakmici gdje su se nalazile te akcije smatraju kao da ne postoji željena akcija što će se tretirati kao klasa *„background“*.

Također, ranije je spomenuto kako postoje oznake akcija koje su označene kao *„not shown“*. Kako bi se izbjeglo zbunjivanje modela da klasificira nešto što nije vidljivo kamerom, odlučeno je da će se u skupu podataka zadržati oznake koje su označene kao *„visible“*. Distribucija oznaka nakon zadržavanja samo vidljivih oznaka prikazana je na slici (Slika 2.4)



Slika 2.4 Distribucija nogometnih akcija nakon zadržavanja vidljivih oznaka

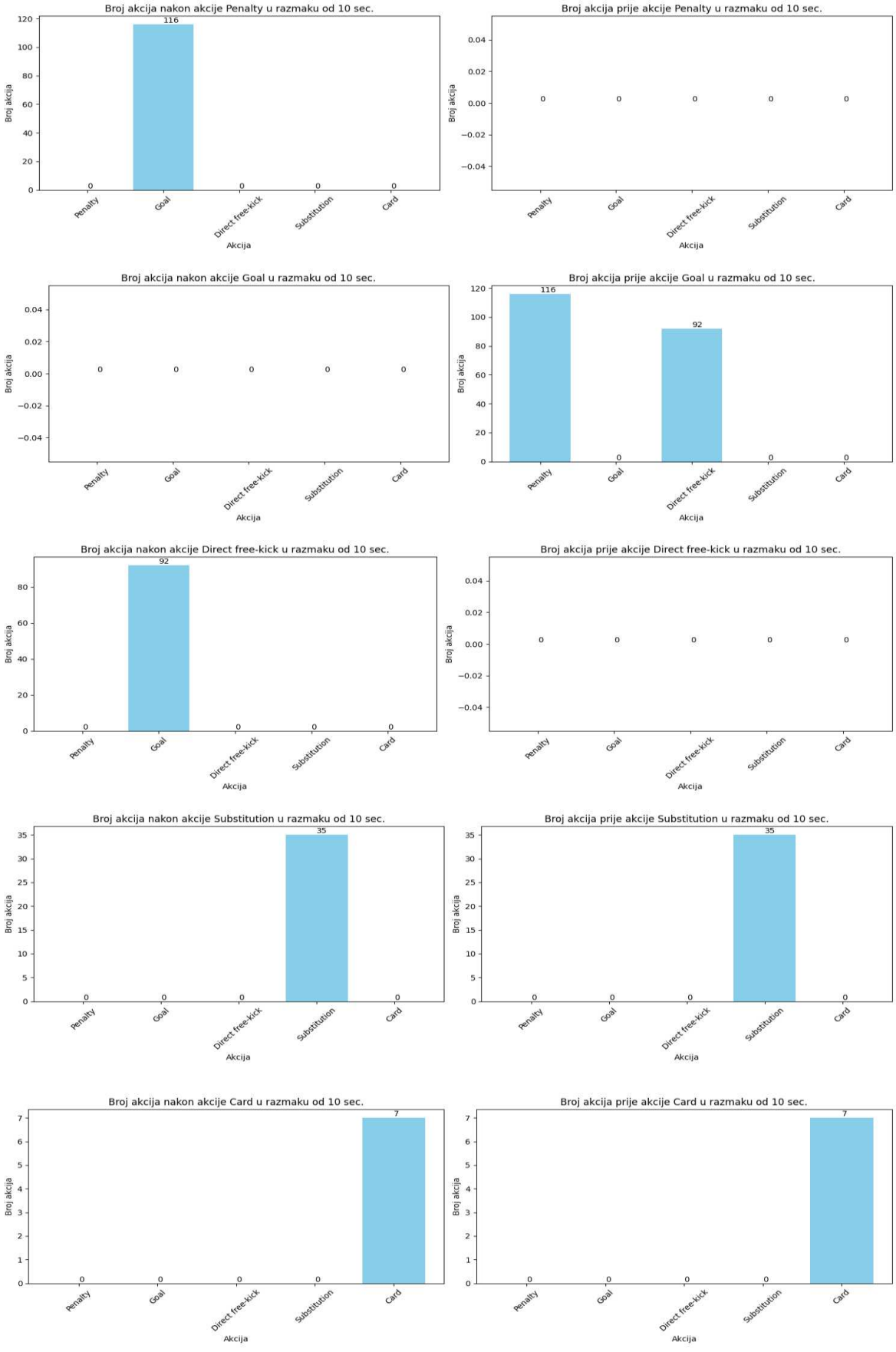
S obzirom da su akcije *„Red card“* i *„Yellow->red card“* veoma rijetke unutar cijelog skupa, odlučeno je da će se te dvije akcije te akcija *„Yellow card“* tretirati zajednički kao akcija *„Card“*. Spajajući te tri klase dobivamo novu klasu koja ima 1966 oznaka što će doprinijeti većoj balansiranosti podataka.

Nadalje, kako bi oznake i podaci bili što kvalitetniji za proces učenja modela, napravljena je detaljna analiza preostalih nogometnih akcija u smislu vremenskih intervala unutar kojih se oznake međusobno nalaze zbog ograničenja mogućnosti detektiranja jedne akcije unutar video sekvence. Kako bismo to provjerili, oko svake nogometne akcije uzet je vremenski interval od 10 sekundi, odnosno 10 sekundi neposredno prije i 10 sekundi neposredno nakon svake nogometne akcije. Analizom su utvrđeni sljedeći problemi:

1. Postoje duplicirane oznake unutar iste sekunde te je odlučeno da će se zadržati jedna od tih oznaka
2. Kod oznake *„Card“* vrlo je čest slučaj podjele više kartona u kratkom vremenskom periodu
3. Situacija kada igrač dobije karton neposredno nakon postignutog gola
4. Kada je iz slobodnog udarca ili penala postignut gol, nakon oznaka *„Direct free-kick“* ili *„Penalty“* dolazi oznaka *„Goal“*
5. Postoje situacije kada se neposredno prije izvođenja slobodnog udarca ili neposredno nakon obavlja zamjena igrača

Uz prethodno opisane situacija postoje i dodatne mogućnosti koje se mogu pojaviti tijekom utakmice. Gledajući od strane kreiranja nogometnih sažetaka, nije problem što akcije slijede jedna iza druge jer je u tom slučaju dovoljno detektirati jednu od njih te će druga (ako je unutar pokrivenog vremenskog intervala nakon prve akcije) sama po sebi nalaziti u sažetku. Međutim, od strane modela koji može detektirati samo jednu akciju po sekvenci to može stvarati problem, pogotovo tijekom učenja jer je tada teže odrediti što je model dao na izlazu u odnosu na to što je trebao dati. Pozitivna stvar je što takvih slučaja nema puno te se mogu u ovom slučaju tretirati kao svojevrsne stršeće vrijednosti koje je potrebno procesirati kako bi video sekvence koje će biti predane na ulaz modela bile što čišće i proces učenja što bolji.

Na slici (Slika 2.5) prikazan je broj oznaka koje se nalaze neposredno prije ili neposredno nakon određene akcije u intervalu od 10 sekundi, nakon što su određene stršeće vrijednosti otklonjene. Stršeće vrijednosti su otklonjene na način da je iz skupa podataka odbačeno poluvrijeme nogometne utakmice unutar kojih se je takva oznaka nalazila, a odbačeno je ukupno 17 poluvremena čime se neznatno izgubi i nekolicina drugih oznaka. Na grafovima na slici (Slika 2.5) može se vidjeti koje od akcija koje slijede drugu akciju su zadržane.



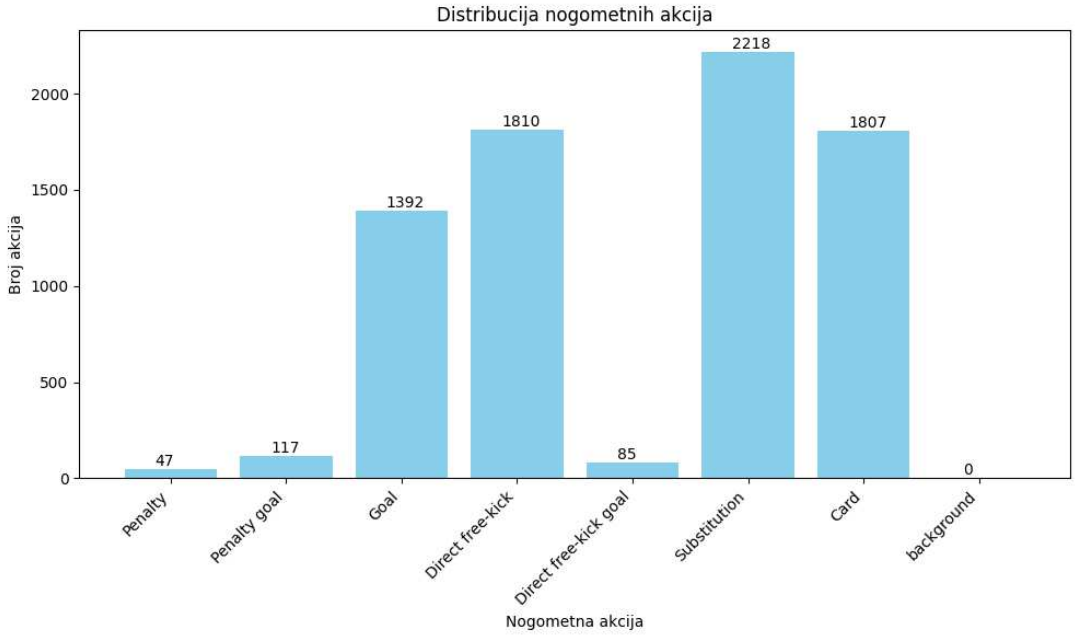
Slika 2.5 Broj akcija u intervalu prije i nakon svake od akcija

Vidimo da neposredno prije akcije *„Penalty“* i *„Direct free-kick“* te nakon akcije *„Goal“* u intervalu od 10 sekundi ne postoji niti jedna druga akcija. S druge strane, vidimo da postoji 116 akcija *„Goal“* nakon akcije *„Penalty“* u intervalu 10 sekundi, a naknadno je utvrđeno da se sve akcije *„Goal“* nalaze zapravo maksimalno 2 sekunde od akcije *„Penalty“* što ukazuje da je zabijen gol iz penala. Slično možemo uočiti i za akciju *„Direct free-kick“*. Što se tiče akcija *„Substitution“* i *„Card“,* vidimo da se u intervalu od 10 sekundi prije ili nakon nalazi ponovno ista akcija što u ovom slučaju ne uzimamo kao problem jer čak i ako se unutar video sekvence nađu dvije iste akcije, model bi trebao ispravno klasificirati tu sekvencu, a dovoljno dobrim bismo smatrali da predvidi vrijeme jednog od tih događaja pošto će se zasigurno oba naći u krajnjem sažetku.

Dakle, jedini problem koji je ostao je pojavljivanje akcije *„Goal“* koja slijedi akcije *„Penalty“* ili *„Direct free-kick“*. Taj problem je riješen na sljedeći način:

- Izvest ćemo novu oznaku *„Penalty goal“* na način da oznaku *„Goal“* koja slijedi oznaku *„Penalty“* pretvorimo u *„Penalty goal“*
- Izvest ćemo novu oznaku *„Direct free-kick goal“* na način da oznaku *„Goal“* koja slijedi 4 sekunde nakon oznake *„Direct free-kick“* pretvorimo u *„Direct free-kick goal“*

Na taj način izvedene su dvije nove dodatne oznake, a konačna distribucija oznaka prikazana je na slici (Slika 2.6).

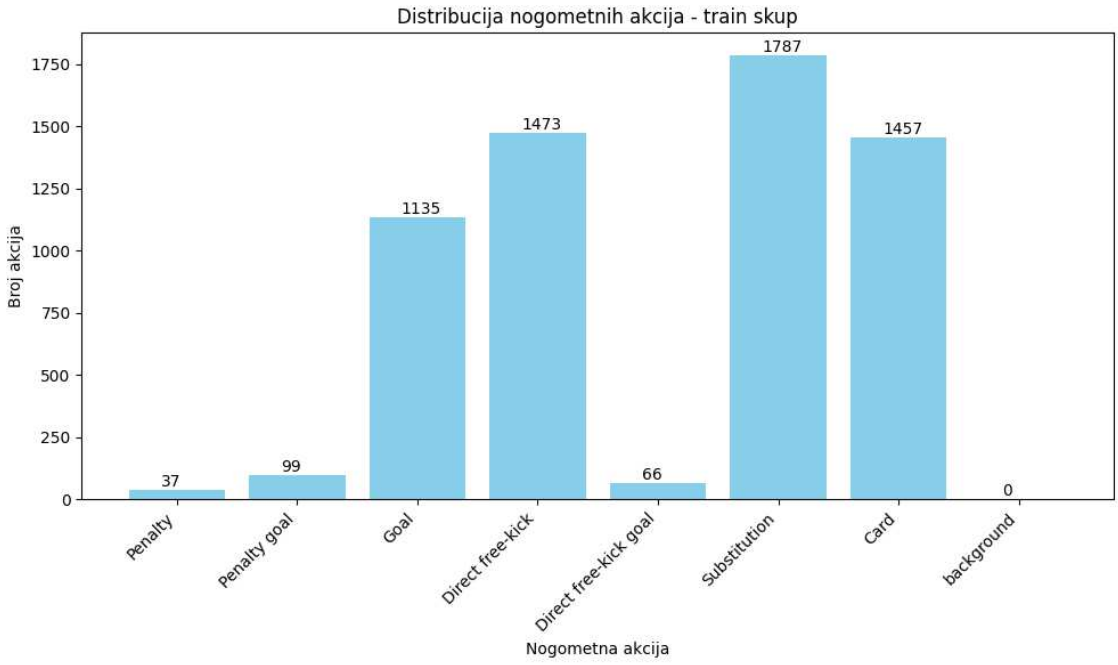


Slika 2.6 Konačna distribucija akcija nakon procesiranja

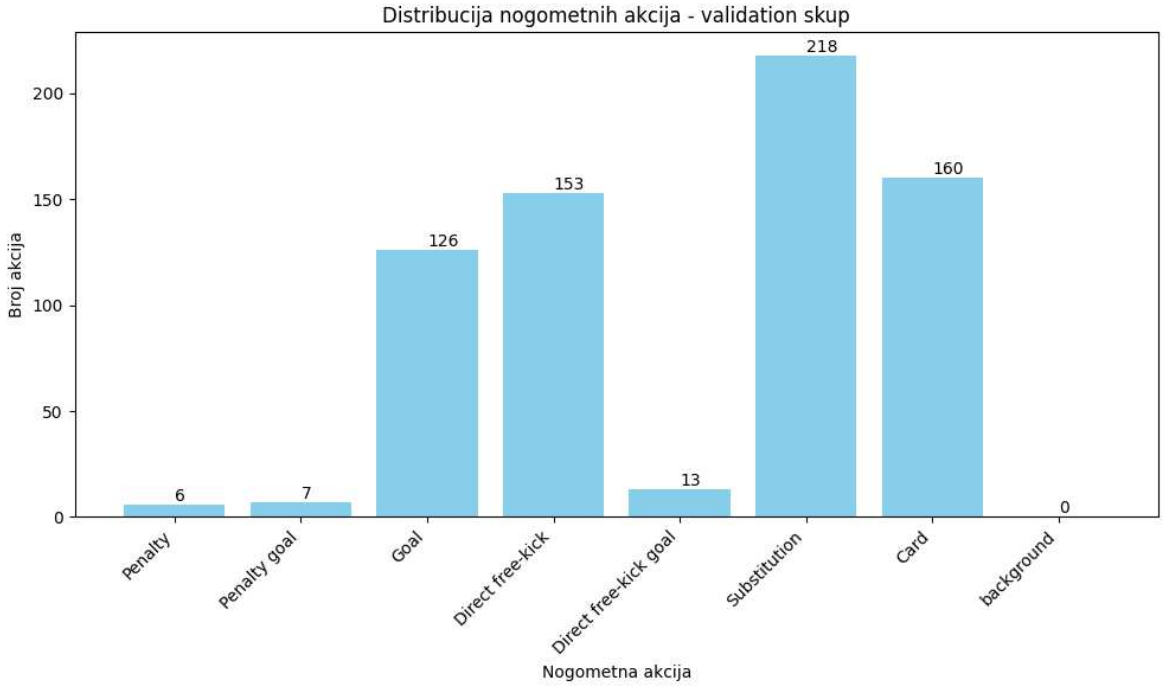
Posljednji korak koji je nužan kod svakog algoritma dubokog učenja je podjela podataka na skup za učenje, skup za validaciju i skup za testiranje kako bi se metoda mogla evaluirati na ispravan način. S obzirom da su u procesu procesiranja podataka odbačena neka poluvremena, dogodio se i slučaj odbacivanja dvije cijele utakmice pa tako od početnih 500 utakmica sada imamo 498 utakmica. Skup je nasumično podijeljen na sljedeći način:

- Skup za učenje: 400 utakmica
- Skup za validaciju: 49 utakmica
- Skup za testiranje: 49 utakmica

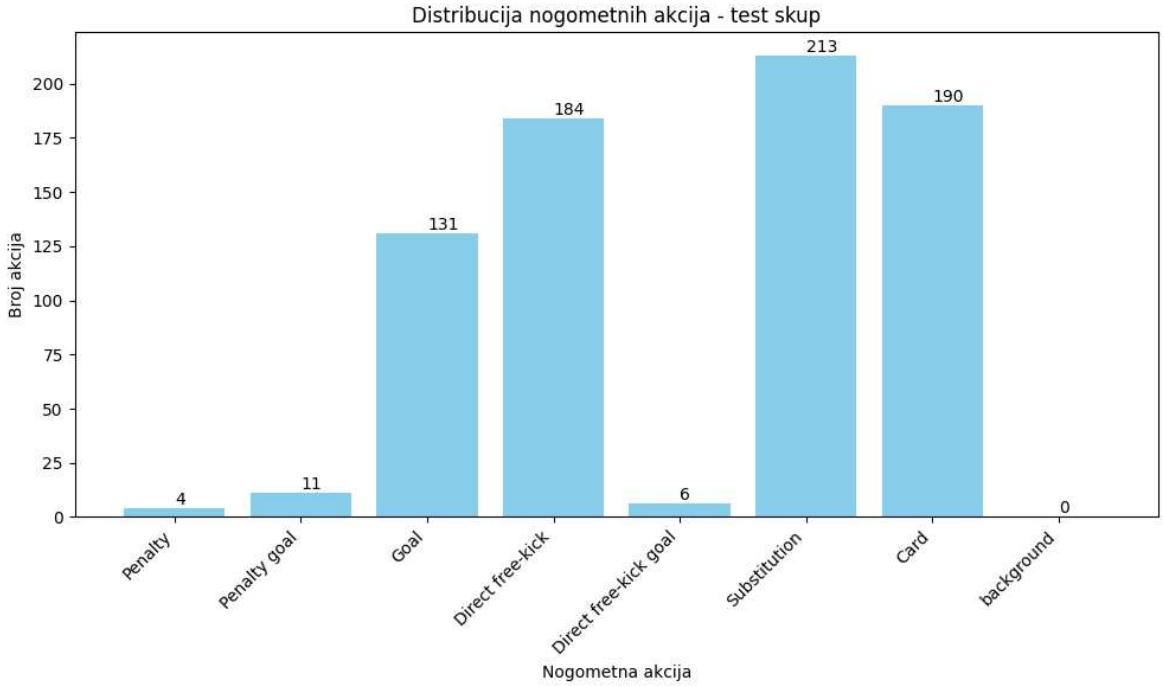
Na sljedećim slikama prikazana je distribucija oznaka po svakom skupu pojedinačno. Na slici (Slika 2.7) nalazi se distribucija oznaka skupa za učenje, na slici (Slika 2.8) skupa za validaciju te na slici (Slika 2.9) skupa za testiranje. S obzirom na distribuciju oznaka skupa za učenje, vidimo da postoje klase oznaka s malo primjera pa će biti potrebno primijeniti tehnike augmentacije podataka kako bi model uspio naučiti upravo te akcije, no detaljnije o tome opisano je u poglavlju (3).



Slika 2.7 Distribucija oznaka skupa za učenje



Slika 2.8 Distribucija oznaka skupa za validaciju



Slika 2.9 Distribucija oznaka skupa za testiranje

3. Metodologija detekcije nogometnih akcija

U prethodnom poglavlju (2) je opisan skup podataka koji se koristi u ovom radu te procesiranje cijelog skupa kako bi korištena metoda ostvarila željeni cilj generiranja sažetaka nogometne utakmice i postigla što bolje rezultate. U ovom poglavlju detaljno je opisana korištena metoda detekcije nogometnih akcija koja se globalno sastoji od dva dijela. Prvi dio je ekstrakcija značajki iz samog videa pomoću konvolucijske neuronske mreže ResNet-152 koja na ulazu prima originalni video rezolucije 224x224, dok se kao izdvojene značajke uzimaju dobiveni vektori nakon sloja globalnog sažimanje dimenzije (T, 2048), gdje te T predstavlja broj slikovnih okvira (engl. *frames*) unutar videa na ulazu. Dakle, zadatak prve neuronske mreže je izdvajanje značajki tj. izdvajanje smislenih reprezentacija ulaznih podataka te takvu neuronsku mrežu nazivamo engl. *backbone*. Izdvojene značajke tada se šalju na ulaz druge neuronske mreže, koju prema autorima članka [2], nazivamo *„RMS-Net“*. Zadatak te neuronske mreže je klasifikacija video sekvence u jednu od klasa koja predstavlja jednu nogometnu akciju te vremenski odmak kada se akcija dogodila. Ukoliko u video sekvenci ne postoji akcija, *„RMS-Net“* treba tu sekvencu klasificirati u klasu *„background“*. Najprije ćemo u poglavlju (3.1) opisati izdvajanje značajki, zatim će u poglavlju (3.2) biti detaljno opisana arhitektura i način rada *„RMS-Net“* mreže te u poglavlju (3.3) način evaluacije uspješnosti kreiranja sažetaka nogometne utakmice.

3.1. Ekstrakcija značajki pomoću konvolucijske neuronske mreže ResNet-152

Zadatak ekstrakcije značajki ima jako veliku ulogu u području računalnog vida jer gotovo svaki sustav ili zadatak ovisi o reprezentaciji ulaznih podataka. Prije uvođenja strojnog/dubokog učenja koristile su se metode temeljene na statistici koje su imale poteškoća u pronalaženju uzoraka zbog različitih scenarija podataka, sadržaja i mogućnosti položaja objekata na slici te općenito problem procesiranja veće količine podataka, a već prvi napredak donijelo je uvođenje klasičnih algoritama strojnog učenja [4]. Međutim,

prekretnica su bile konvolucijske neuronske mreže s kojima je postalo ostvarivo procesiranje velikih skupova podataka te učenje značajki neovisno o položaju, veličini, rotaciji ili osvijetljenosti objekata na slici [4]. Danas postoje brojne poznate arhitekture koje se koriste u svrhu ekstrakcije značajki (engl. *backbones*) poput:

- AlexNet
- GoogleNet
- VGG
- MobileNet
- EfficientNet
- ResNet i druge

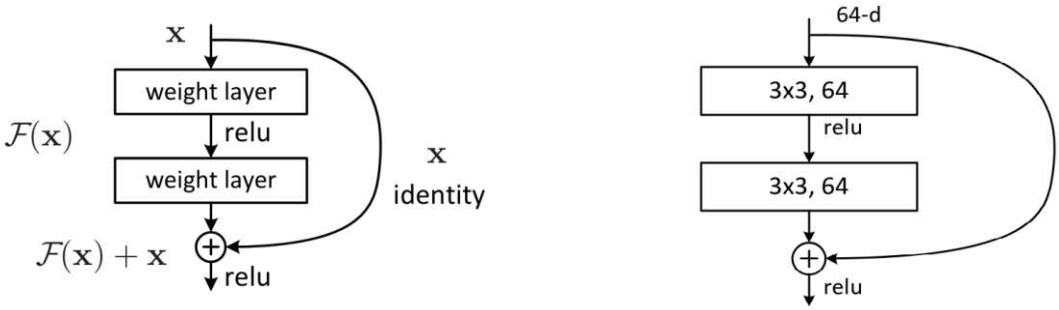
Za različite zadatke koji se mogu rješavati u području računalnog vida, odabir „prave“ arhitekture za ekstrakciju značajki može biti vrlo skup i zahtjevan proces [4]. U zadatku poput ovog, gdje na temelju značajki želimo prepoznati nogometnu akciju, jako je bitno prepoznati dobre i kvalitetne značajke iz sirovih visokodimenzionalnih podataka poput rubova, tekstura, boja ili oblika.

S obzirom na rezultate koje je postigla kroz brojne zadatke i svoju specifičnu arhitekturu, kao ekstraktor značajki (engl. *backbone*) u ovom radu je korištena je arhitektura konvolucijske neuronske mreže ResNet-152.

ResNet arhitektura je tip konvolucijske neuronske mreže koja uključuje „rezidualno učenje“ koje je osmišljeno od strane autora članka [5] koji su i predstavili samu arhitekturu ResNet. Ova arhitektura omogućila je lakše treniranje mnogo dubljih neuronskih mreža uvođenjem rezidualnih blokova koji pomažu kod suočavanja s problemom nestajanja ili eksplodiranja gradijenta, koji su vrlo česti kod dubljih modela, a doprinose nestabilnosti procesa učenja što čini konvergenciju otežanom. Na slici (Slika 3.1) prikazan je jedan primjer rezidualnog bloka od kojih se sastoji ResNet konvolucijska neuronska mreža. Temeljna ideja rezidualnog bloka su takozvane „skip konekcije“ koje omogućuju preskakanje jednog ili nekoliko slojeva te direktno povezuju ulaz s dubljim slojem. Na slici (Slika 3.1) možemo vidjeti kako je ulaz prvog sloja direktno povezan na izlaz sljedećeg sloja.

Kada želimo riješiti neki problem poput ekstrakcije značajki, intuicija govori da dodavanjem više slojeva postižemo učenje od jednostavnijih prema kompleksnijim

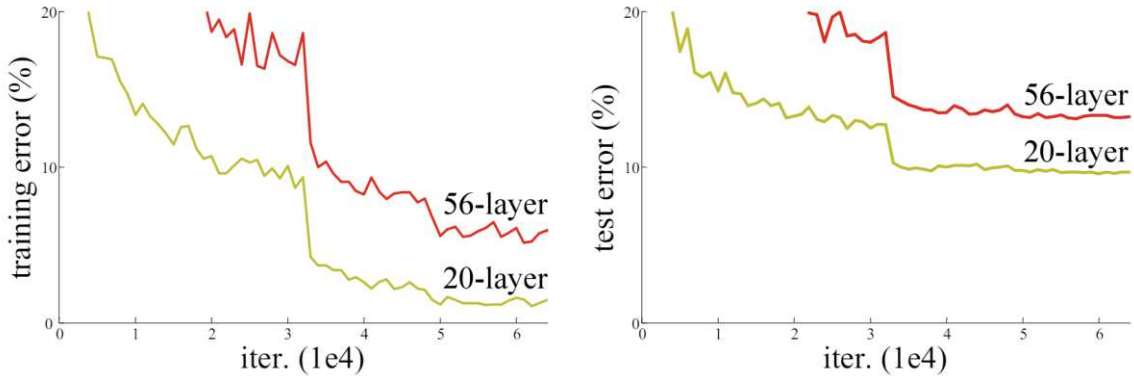
značajkama što bi pomoglo u učenju korisnik značajki. Međutim povećavajući dubinu mreže, točnost postaje zasićena i manja kao što vidimo na slici (Slika 3.2) [5].



Slika 3.1 Rezidualni blok [5]

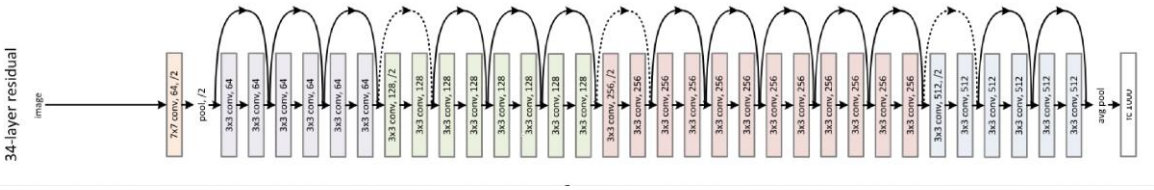
Dodavanjem „skip konekcije“, umjesto da model uči izravno mapiranje, model uči rezidual tj. razliku između ulaza i izlaza sloja. Dakle, umjesto da model aproksimira **H(x)**, što predstavlja funkciju koju bi mreža idealno nastojala naučiti kako bi mapirala ulaz **x** na željeni izlaz, eksplicitno smo dopustili modelu da aproksimira **F(x) + x = H(x)** [5]. Upravo zbog toga dolazi do smanjenja problema degradacije performansi i omogućuje se da značajno dublji modeli bolje uče i hvataju složenije uzorke iz podataka.

Gledajući unatražnji prolaz neuronske mreže, gradijenti se često smanjuju ili povećavaju uzastopnim množenjem pravilom ulančavanja što dovodi do nestabilnog i otežanog učenja. „Skip konekcije“ omogućavaju da se gradijenti izravno prenesu kroz mrežu bez prolaska kroz sve slojeve što pomaže u sprječavanju nastajanja eksplodirajućeg ili nestajućeg gradijenta što uvelike pomaže u stabilnosti procesa učenja.



Slika 3.2 Pogreška učenja neuronske mreže koja ne koristi „skip konekcije“ [5]

Postoji više mogućih arhitektura ResNet konvolucijske neuronskre mreže poput: ResNet-18, ResNet-34, ResNet-152. Broj koji se nalazi u imenu arhitekture označava ukupan broj slojeva te mreže. Kao primjer, na slici (Slika 3.3) nalazi se arhitektura mreže ResNet-34 koja ima ukupno 34 sloja.



Slika 3.3 Arhitektura ResNet-34

U ovom radu kako bi ekstrakcija značajki bila što kvalitetnija, korištena je arhitektura ResNet-152 koja se sastoji od ukupno 152 sloja. Korišten je model čije su težine prethodno naučene na poznatom skupu podataka ImageNet [7]. Na ulaz modela dolaze slike (okviri video sekvence) u rezoluciji od 224x224x3, a kao izlaz uzimaju se značajke koje se dobiju propuštanjem ulaza kroz neuronsku mrežu sve do sloja srednjeg globalnog sažimanja. Taj sloj uzima srednju vrijednost svih mapi značajki što rezultira jednom vrijednosti po mapi značajki čime za jednu sliku na ulazu, dobijemo 1D vektor značajki na izlazu, koji je dimenzije 2048 zbog tolikog broja mapi značajki. Ukoliko se radi o videozapisu od 45 minuta, koliko je trajanje jednog poluvremena utakmice, pretvarajući to u sekunde, dobije se 2700 sekundi. S obzirom da okvire u ovom radu uzorkujemo frekvencijom 2 okvira po sekundi, za jedno takvo poluvrijeme dobit ćemo 5400 okvira koji se moraju propustiti kroz mrežu ResNet-152. Shodno tome, video od 45 minuta nakon prolaska kroz mrežu, na izlazu daje značajke dimenzije (5400, 2048). Također, autori skupa podataka SoccerNet [1] pružaju značajke dobivene istim modelom uz korištenje metode glavnih komponenti (engl. *Principal component analysis*, PCA), čime je dimenzija značajki od 2048 reducirana na 512. U tom slučaju, značajke za video od 45 minuta imaju dimenziju od (5400, 512).

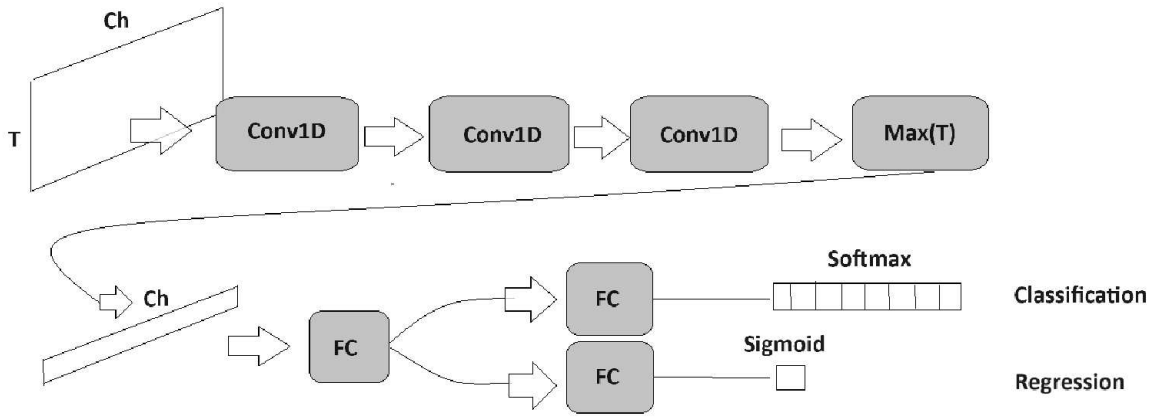
Nakon što se dobiju značajke kao što je opisano, te značajke se koriste kao ulaz u drugi model ovog rada. Model i način na koji se vrši klasifikacija svake video sekvence i predviđanje vremena nogometne akcije, opisan je u sljedećem poglavlju (3.2).

3.2. RMS-Net

Nakon što se odradi proces ekstrakcije značajki kao što je opisano u prethodnom poglavlju (3.1), vrijeme je za ključni model i metodologiju ovog rada, a to je model koji vrši klasifikaciju video sekvenci ovisno o nogometnoj akciji te predviđa relativni vremenski odmak u kojem se je akcija dogodila. Kao što je spomenuto ranije, na ulaz ovog modela dolaze video sekvence (sekvence izdvojenih značajku) u trajanju od 20 sekundi tj. 41 okvira.

Formalno možemo reći da model na ulazu prima sekvencu $\mathbf{X} = (x_1, x_{2,\ldots,}x_T)$, gdje je T = 41 i predviđa vremenski pomak moguće akcije te vrši klasifikaciju sekvence u jednu od C + 1 klasa, gdje je C = 7, a dodatna klasa je klasa *„background“*.

Na slici (Slika 3.4) prikazana je arhitektura RMS-net modela, dok se u tablici (Tablica 1.) nalazi detaljni opis cijele arhitekture. Dakle, model na ulazu prima sekvencu okvira, koja je po vremenskoj dimenziji duljine **T** = 41, dok u prostornoj dimenziji ulaz ima dimenziju 512, nakon što se odradi ekstrakcija značajki te analiza glavnih komponenti. Nakon ulaza slijede tri uzastopna 1D konvolucijska sloja koja vrše konvoluciju preko vremenske dimenzije kako bi se kombinirale značajke različitih okvira unutar ulazne sekvence [2].



Slika 3.4 Arhitektura RMS-Net

Nakon tri konvolucijska sloja slijedi operacija maksimuma kako bi uklonili vremensku dimenziju, nakon čega slijedi jedan potpuno povezani sloj te na samom kraju dva paralelna potpuno povezana sloja gdje jedan vrši klasifikaciju sekvence u jednu od C+1 klasa, primjenjujući aktivacijsku funckiju softmax na logite prethodnog sloja, a drugi predviđa regresijski vremenski odmak primjenjujući sigmoidalnu aktivacijsku funkciju na logite prethodnog sloja. U nastavku će biti objašnjena ideja sigmoidalne aktivacijske funkcije za predviđanje relativnog vremenskog odmaka.

Tablica 1. Arhitektura RMS-Net modela

Sloj	Aktivacijska funkcija	Veličina konvolucijske jezgre	Ulazni kanali	Izlazni kanali
Conv1D	ReLu	3	512	256
Conv1D	ReLu	9	256	256
Conv1D	ReLu	9	256	128
Max preko vremena	-	-	-	-
FC	ReLu	-	128	64
$FC_{klasifikacija}$	Softmax	-	64	C + 1
$FC_{regresija}$	Sigmoida	-	64	1

Pomalo je zbunjujuće koristiti aktivacijsku funkciju sigmoide prilikom predviđanja vremena kada se neka akcija dogodila u video sekvenci od 20 sekundi jer sigmoida na svom izlazu daje vrijednosti u intervalu [0, 1]. Međutim, prilikom kreiranja video sekvenci potrebno je označiti u kojem dijelu video sekvence se neka akcija dogodila. S obzirom na analizu koja je prikazana na slici (Slika 2.5), ideja je najprije uzeti sve moguće akcije iz skupa za učenje te oko svake akcije uzeti interval od 20 okvira (10 sekundi) s obzirom da su upravo okviri (engl. *frames*) ti koji čine vremensku dimenziju izdvojenih značajki. Dakle, nakon što uzmemo 20 okvira prije i poslije neke akcije kreirali smo video sekvencu

koja sadrži akciju koju je potrebno detektirati, a s obzirom da se ona nalazi točno na sredini te sekvence, vrijeme kada se akcija dogodila pretvaramo u spomenuti relativni vremenski odmak i označavamo ga s 0.5 u tom slučaju. Video sekvencu označavamo kao klasu ovisno o tome koja nogometna akcija se nalazi unutar te video sekvence. Za preostale dijelove početnog videa potrebno je kreirati video sekvence koje ne sadrže željenu nogometnu akciju te označiti takvu video sekvencu kao klasu *„background“*. Što se tiče vremenske oznake, takvim sekvencama je dodijeljena nasumična oznaka iz intervala [0,1]. Ono što je bitno spomenuti je da kada se kreira video sekvenca koja ne sadrži akciju, kako sigurno ne bi došlo do preklapanja video sekvenci, početak *„background“* video sekvence počinje 41 okvira (engl. *frames*) nakon neke od označenih akcija. Također sekvenca koja je označena kao *„background“* završava 41 okvir (engl. *frame*) prije neke od označenih akcija.

Ovdje vrijedi istaknuti kako sve video sekvence koje sadrže neku od željenih akcija tu akciju sadrže točno na pola sekvence s vremenskom oznakom 0.5 što izgleda da bi model mogao trivijalno naučiti na način da prilagodi predviđanje akcije točno na sredini video sekvence. Kako bi to izbjegli, uvedena je tehnika augmentiranja podataka i video sekvenci koja će pomoći u izbjegavanju tog problema, a s druge strane povećati skup podataka za klase koje nemaju puno oznaka poput *„Penalty“* i *„Direct free-kick goal“* kao što je prikazano na slici (Slika 2.7). Način na koji su podaci augmentirani objašnjen je u poglavlju (3.2.1).

S obzirom na specifičnost izlaza, potrebno je iskoristiti specifičnu funkciju gubitka koja se sastoji od dva dijela:

1. Klasifikacijski gubitak
2. Regresijski gubitak

Klasifikacijski gubitak je standardni gubitak unakrsne entropije koji je prikazan formulom (1), gdje je C+1 broj klasa, a p_c vjerojatnost izlaza mreže da video sekvenca pripada klasi *c*. Indikatorska funkcija $\mathbf{1}_{c=ej}$ poprima vrijednost 1 ako je klasa *c* jednaka stvarnoj klasi e_j , a inače je 0.

$$\mathcal{L}_{cls} = -\sum_{c=0}^{C} \mathbb{1}_{c=e_j} \log(p_c) \tag{1}$$

S druge strane, kao regresijski gubitak koristi se standardni kvadratni gubitak koji je prikazan formulom (2), gdje je *o* izlaz regresijskog dijela modela koji se propušta kroz sigmoidu, a r_j predstavlja relativni vremenski odmak unutar video sekvence.

$$\mathcal{L}_{regr} = (\sigma(o) - r_j)^2 \tag{2}$$

Ukupni gubitak nad kojim je RMS-Net model učen je težinska suma dva spomenuta gubitka te je prikazan formulom (3), gdje je λ koeficijent koji otežuje regresijski dio gubitka .

$$\mathcal{L} = \mathcal{L}_{cls} + \lambda \mathcal{L}_{regr} \tag{3}$$

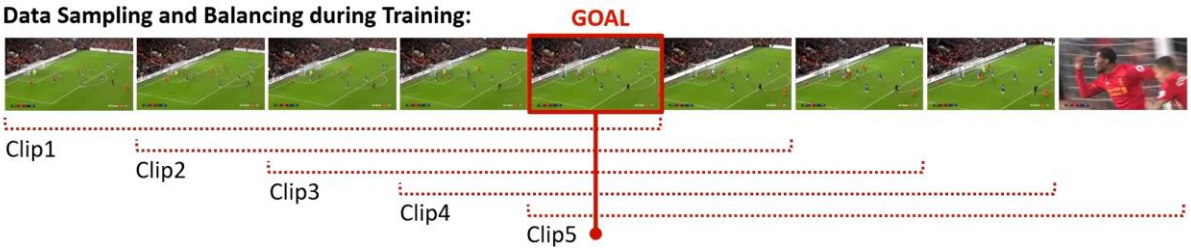
Ono što je bitno spomenuti, video sekvence koje ne sadrže nogometnu akciju i označene su kao *„background“*, za njih je regresijski dio gubitka nula te kod takvih primjera na ukupni gubitak utječe samo klasifikacijski gubitak.

S obzirom da smo ranije spomenuli kako postoje klase koje imaju znatno manje primjera (nogomentih akcija) poput *„Penalty“, „Penalty goal“* i *„Direct free-kick goal“*, potrebno provesti augmentaciju podataka kako bi model bio u stanju naučiti te klase, razlikovati ih od drugih ili ne svrstavati u većinsku klasu. Način na koji su podaci augmentirati opisan je u sljedećem poglavlju (3.2.1)

3.2.1. Augmentacija video sekvenci skupa za učenje

Na distribuciji klasa prikazanoj na slici (Slika 2.7), možemo vidjeti kako neke od klasa imaju znatno manje primjera. Također, spomenuli smo da način na koji su kreirane video sekvence, sadržavajući svaku nogometnu akciju točno na sredini, nije najidealniji zbog toga što bi model u procesu testiranja s velikom vjerojatnošću naišao na video sekvencu koja nogometnu akciju nema točno na sredini, već ovisno o tome kako bi slijedno dolazile video sekvence, akcija bi unutar te sekvence mogla biti na bilo kojem mjestu te model ne bi bio dovoljno robustan da prepozna takvu situaciju s obzirom da to nikad nije vidio u skupu za učenje. Upravo iz toga proizlazi motivacija za augmentaciju video sekvenci koja se slikovito može prikazati kao što je prikazano na slici (Slika 3.5). Ideja je uzorkovanje video sekvenci iz nogometnih utakmica na način da se originalna video sekvenca koja ima označenu nogometnu akciju točno na srediti s oznakom 0.5, pomiče za određeni broj okvira (engl. *frames*) unaprijed ili unatrag kako bi se relativna pozicija akcije unutar video

sekvence mijenjala. Kao što vidimo na slici (Slika 3.5), u prvoj video sekvenci akcija *„Goal“* se nalazi na samom kraju, u trećoj sekvenci se nalazi točno na sredini, a u petoj sekvenci se nalazi na početku. Time se povećava robusnost modela te mogućnost učenja da može predvidjeti vremenski pomak akcije bez obzira što se ona nalazi na različitim mjestima unutar video sekvence.



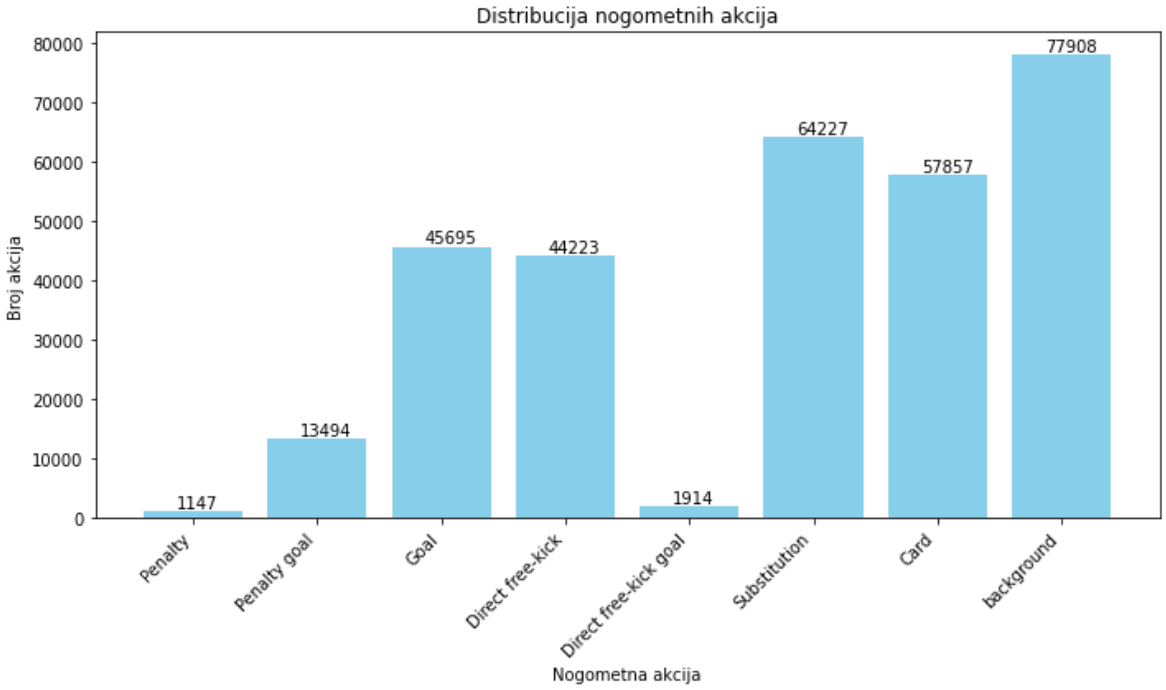
Slika 3.5 Primjer augmentacije video sekvence

S obzirom na to da se u originalnom uzorkovanju svaka nogometna akcija nalazi točno u sredini video sekvence duljine 41 okvira, sa svake strane akcije nalazi se 20 okvira za koje je moguće pomaknuti video sekvencu. Takvim pomicanjem postoji mogućnost preklapanja s nekom drugom nogometnom akcijom što može biti problematično. Na slici (Slika 2.5) prikazano je koliko se akcija nalazi u okolini akcije od 20 okvira tj. 10 sekundi sa svake strane. Ako video sekvencu recimo pomičemo za 20 okvira u lijevo, u tom slučaju prije same akcije koju promatramo sada se više ne nalazi 20 okvira već 40 što dovodi u pitanje postoji li preklapanje. Taj problem je riješen na način da je napravljena provjera koliko se akcija nalazi u krugu od 40 okvira oko svake akcije te za sva preklapanja koja su pronađena naprosto je maksimalni pomak u oba smjer prilagođen tak da ne dolazi do preklapanja s drugom akcijom. Takvih slučajeva u skupu podataka nije pronađeno mnogo te je augmentacija provedena bez posljedica.

Što se tiče označavanja pomaknutih video sekvenci, nužno je promijeniti relativni vremenski pomak u skladu s pomakom video sekvence kako bi model imao točnu informaciju gdje se akcija nalazi u pomaknutoj video sekvenci. Za svaki pomak originalne video sekvence, novi vremenski relativni pomak akcije računa se prema formuli (4).

$$\text{relativni\_pomak} = 0.5 - \left(\frac{\text{pomak\_sekvence}}{\text{broj\_okvira\_po\_sekvenci}} \right) \tag{4}$$

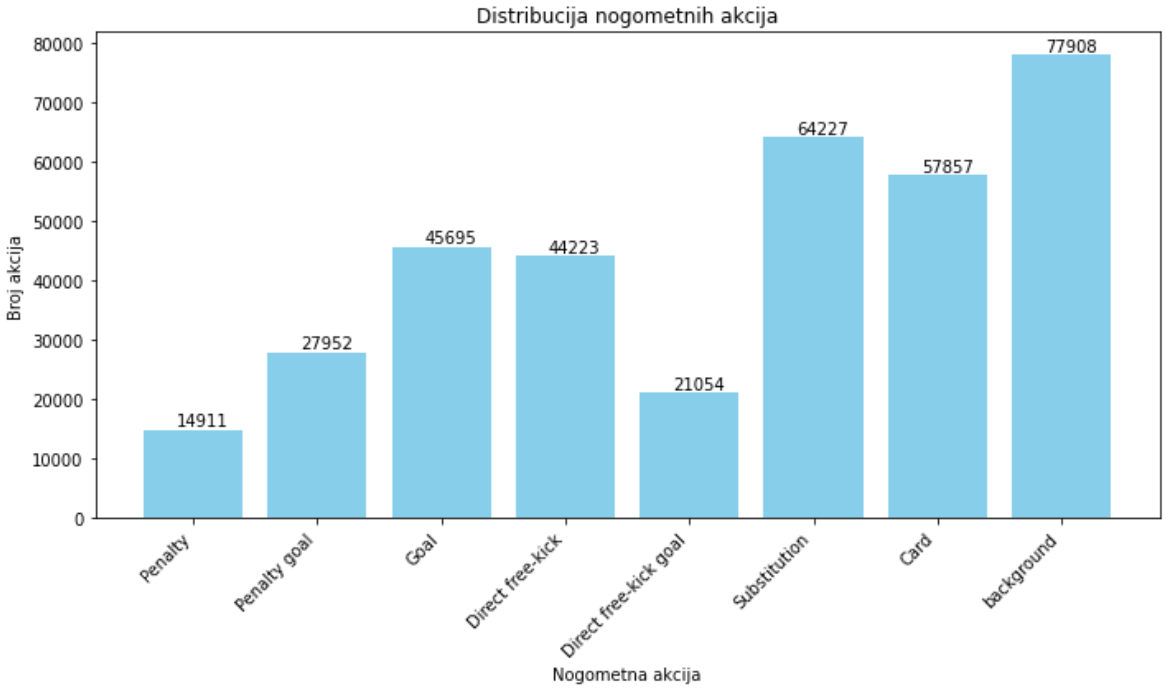
Nakon što je provedena augmentacija video sekvenci pomicanjem svake video sekvence za maksimalni broj od 20 okvira u lijevo ili desno (maksimalni broj je manji od 20 ako dolazi do preklapanja akcija), distribucija novih oznaka skupa za učenje prikazana je na slici (Slika 3.6).



Slika 3.6 Distribucija oznaka skupa za učenje nakon augmentacije pomicanjem video sekvenci

Kao što je prikazano na slici (Slika 3.6), broj oznaka pa tako i ukupno broj video sekvenci je znatno povećan. Jasno, video sekvence koje ne sadrže nogometnu akciju nisu augmentirane, već broj tih sekvenci proizlazi sam po sebi s obzirom da postoji mnogo perioda unutar utakmica kada se ne događa niti jedna od ključnih nogometna akcija. Međutim, problem malog broja video sekvenci s akcijama *„Penalty“, „Penalty goal“* i *„Direct free-kick goal“* i dalje postoji. Kako bi povećali broj primjera tih klasa i donekle uravnotežili skup podataka te kako bi model mogao naučiti razlikovati video sekvence tih klasa, napravljena je dodatna augmentacija. Te tri klase augmentirane su na način da se za svaku video sekvencu slučajnim odabirom maskira određeni okvir te sekvence. Za svaku

video sekvencu koja sadrži nogometnu akciju *„Penalty“* maskirano je ukupno 12 slučajno uzorkovanih okvira od ukupno 40, za video sekvencu koja sadrži nogometnu akciju *„Penalty goal“* maskirano je ukupno 7 okvira od ukupno 40 te za video sekvencu koja sadrži nogometnu akciju *„Direct free-kick goal“* maskirano je ukupno 10 okvira od ukupno 40. Na taj način dobije se dodatno proširenje skupa podataka za učenje te je konačna distribucija oznaka prikazana na slici (Slika 3.7).



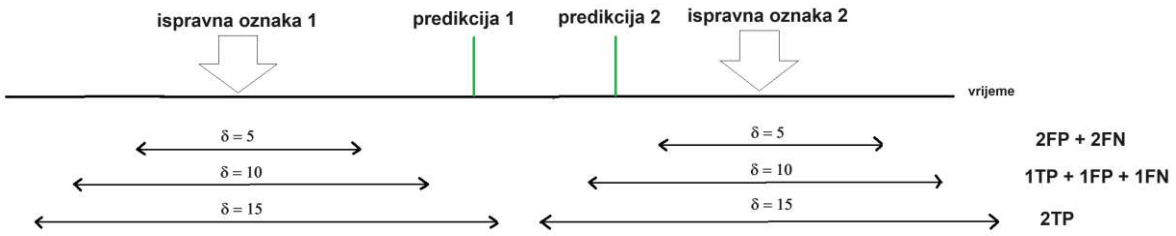
Slika 3.7 Konačna distribucija oznaka skupa za učenje

Prije provođenja eksperimenata, potrebno je objasniti i opisati mjeru kojom će se mjeriti uspješnost modela u zadatku prepoznavanja nogometnih akcija, a ona je detaljno objašnjena u poglavlju (3.3).

3.3. Evalucijska mjera srednje prosječne preciznosti (engl. *mean Average Precision*)

S obzirom da zadatak detektiranja nogometnih akcija ne pokriva samo klasifikaciju već i predikciju vremenskog odmaka, potrebna je evaluacijska mjera koja može mjeriti obje stavke. Evaluacijska mjera koja će se koristiti zove se mjera srednje prosječne preciznosti (engl. *mean Average Precision*). To je mjera koja za svaku klasu pojedinačno provjerava jesu li predikcije unutar fiksnog intervala δ oko ispravnih oznaka te na temelju toga definiramo *„True Positives“*(TP), *„False Positives“*(FP) i *„False Negatives“*(FN) [6]. Fiksni interval δ poprima vrijednosti od 5 do 60 sekundi. Ako je poprimio vrijednost od 10 sekundi, to znači da se promatra je li se predikcija našla u intervalu od 10 sekundi oko ispravne oznake tj. 5 sekundi prije i 5 sekundi nakon.

Na slici (Slika 3.8) možemo vidjeti primjer na kojem se nalaze dvije oznake iste klase te njihove predikcije. Na temelju te slike, pogledat ćemo način na temelju kojeg se dodjeljuju oznake TP, FP ili FN.

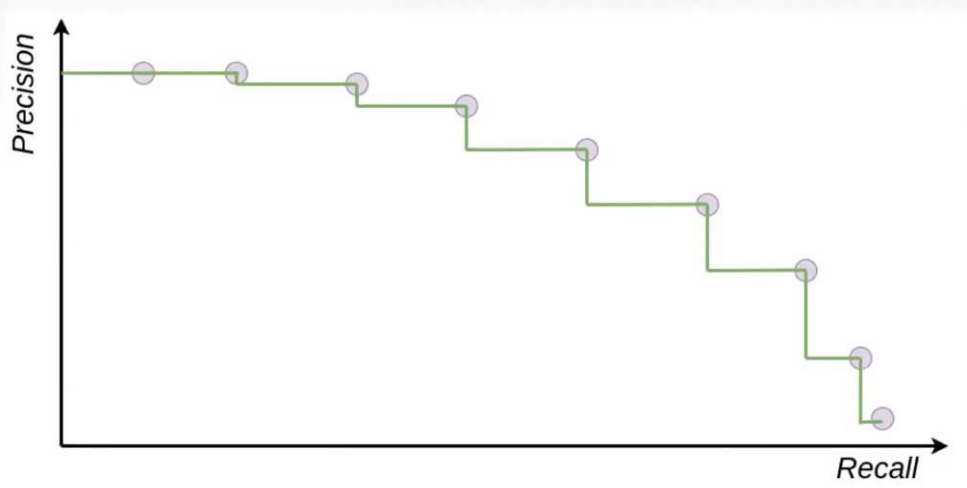


Slika 3.8 Primjer evaluacije pomoću mjere prosječne preciznosti [3]

Najprije ćemo razmotriti interval δ = 5 sa slike (Slika 3.8) gdje se može vidjeti da niti jedna od predikcija ne upada u interval oko svoje ispravne oznake te to računamo kao 2 FP primjera te 2 FN jer za obje ispravne oznake nema predikcije koja upada u interval. Nadalje, gledajući interval δ = 10, vidimo da „predikcija 2“ upada u interval oko „ispravne oznake 2“ čime dobivamo 1 TP, dok „predikcija 1“ ne upada u interval oko „ispravne oznake 1“ te čini 1 FP te „ispravna oznaka 1“ nema predikcija unutar svog intervala pa tako čini 1 FN.

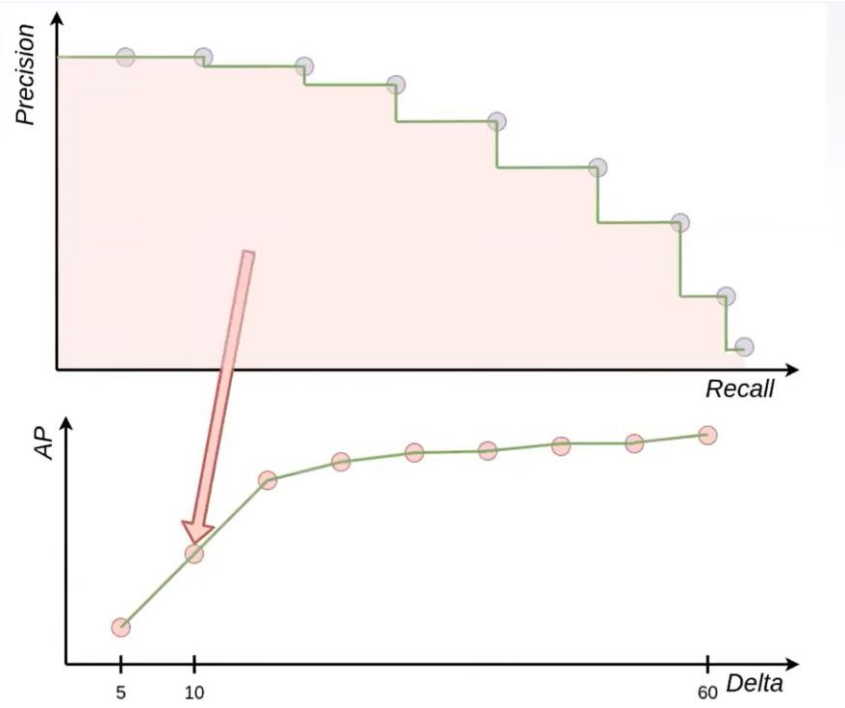
Gledajući posljednji interval sa slike (Slika 3.8), vidimo da obje predikcije upadaju u interval oko svoje ispravne oznake te tako čine 2 TP.

Nakon što se dobiju svi TP, FP i FN za sve intervale δ za svaku klasu, izračunaju se preciznost i odziv. Dobivene metrike potrebno je postaviti u prostor koji je prikazan na slici (Slika 3.9), aproksimirati graf te izračunati područje ispod grafa.



Slika 3.9 Prostor preciznosti i odziva [3]

Nakon što se dobije područje ispod prikazanog grafa, to područje ispod grafa zovemo prosječna preciznost (engl. *Average Precision*). To područje možemo izračunati za svaki od intervala δ (5, 10, 15,…,60) čime možemo prikazati vrijednost prosječne preciznosti ovisno o intervalu δ kao što je prikazano na slici (Slika 3.10).



Slika 3.10 Područje prosječne preciznosti [3]

Nakon što se dobije graf ovisnosti prosječne preciznosti od intervalu δ, potrebno je izračunati područje ispod te krivulje. Nakon toga, proces je potrebno ponoviti za svaku klasu i uzeti njihov prosjek što nam konačno daje mjeru srednje prosječne preciznosti.

4. Eksperimentalna izvedba i rezultati

Nakon što su u prethodnim poglavljima opisani koraci procesiranja podataka, evaluacijskih metoda te metoda koja se koristi, vrijeme je prikazati potpunu izvedbu cijele metode te rezultata koje je ta metoda postigla. U ovom poglavlju će najprije biti prikazani detalji izvedbe sustava, a zatim i rezultati.

4.1. Eksperimentalna izvedba

Kao što je opisano u poglavlju (3), prvi korak cijele metode je ekstrakcija značajki pomoću modela ResNet-152 koji je predtreniran na skupu podataka ImageNet [7]. Nakon ekstrakcije značajki, dolazi model RMS-Net čija je arhitektura opisana tablicom (Tablica 1.). Kako bi model RMS-Net postigao najbolje rezultate, odabrani su hiperparametri i njihove vrijednosti prikazane tablicom (Tablica 2).

Tablica 2. Hiperparametri RMS-Net modela

HIPERPARAMETAR	VRIJEDNOST
Broj epoha učenja	20
Veličina mini-grupe (engl. *batch size*)	16
Moment optimizacijskog postupka	0.85
Weight decay optimizacijskog postupka	0.0001
Stopa učenja (engl. *learning rate*)	0.001
Koeficijent lambda regresijskog gubitka (3)	10
Korišteni optimizator	SGD [8]
Težine klasa kod gubitka unakrsne entropije	[70, 10, 1, 2, 30, 1, 1, 50]

Uz hiperparametre koji su navedeni u tablici (Tablica 2.), korištena je i tehnika dinamičke promjene stope učenja tijekom procesa učenja kosinusnom funkcijom koja doprinosi stabilnosti učenja i poboljšava konvergenciju [9].

Nadalje, s obzirom na dostupnost primjera kao što je prikazano distribucijom na slici (Slika 3.7), neposredno prije svake epohe radi se slučajno uzorkovanje označenih video sekvenci na način:

1. Uzorkuje se 11 000 video sekvenci klasa: *„Penalty“, „Penalty goal“, „Direct free-kick goal“*
2. Uzorkuje se 40 000 video sekvenci klase *„background“*
3. Uzorkuje se 9 000 video sekvenci preostalih klasa: *„Goal“, „Direct free-kick“, „Substitution“* i *„Card“*.

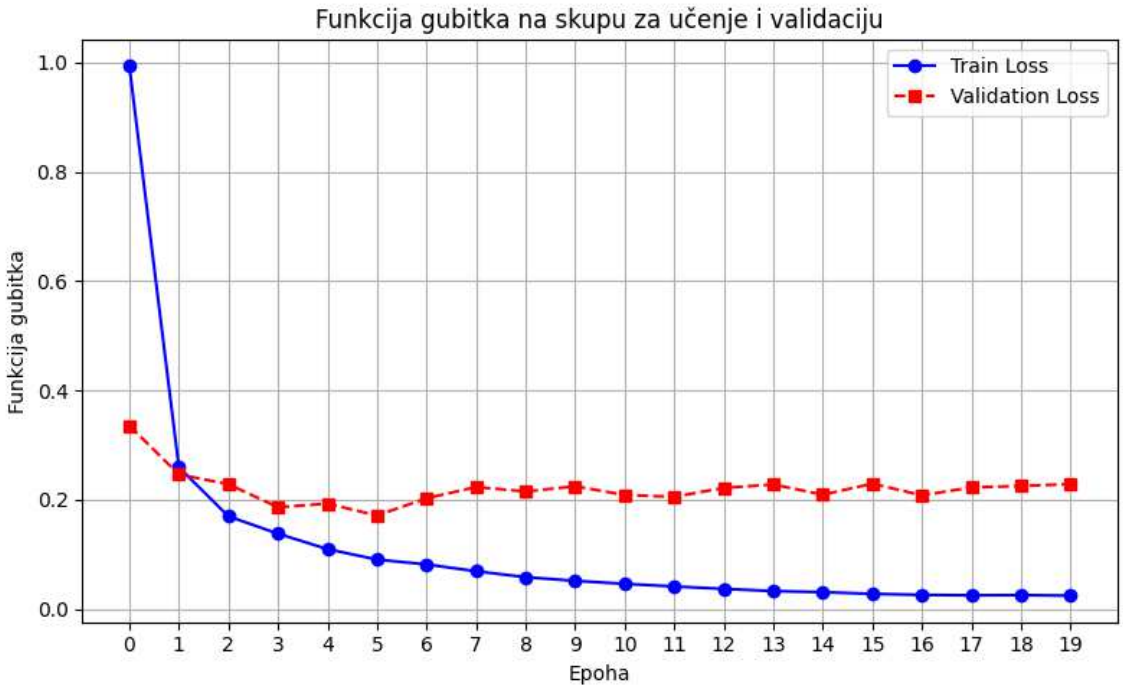
Slučajno uzorkovanje na ovaj način doprinosi tome da model kroz svaku epohu vidi djelomično drugačije primjere iz svake klase što doprinosi boljoj generalizaciji modela u konačnici. Pokazalo se ključnim u svakoj epohi uzorkovati znatno više video sekvenci klase *„background“* kako bi model mogao naučiti razlikovati kada određena video sekvenca ne sadrži niti jednu akciju.

Što se tiče validacijskog i testnog skupa podataka, video sekvence su kreirane na način da se slijedno uzimaju sekvence od 20 sekundi početnog video zapisa te se označe u skladu s oznakama koje su dostupne kako bi mogli evaluirati točnost modela RMS-Net. Na kraju, relativni vremenski pomak se u testnom okruženju pretvara u apsolutni kako bismo dobili točna vremena detektiranih akcija unutar cijele utakmice.

4.2. Rezultati

U ovom poglavlju detaljno ćemo razmotriti evaluacijske metrike te rad samog modela na validacijskom i testnom skupu podataka. Hiperparametri koji su predstavljeni u prethodnom poglavlju (4.1) pokazali su se najboljim nakon brojnih eksperimenata. Eksperimenti su obuhvaćali i promjene u arhitekturi modela RMS-Net što je na kraju i dovelo do drugačije arhitekture u odnosu na arhitekturu koju su predstavili autori članka [2]. Nakon svih eksperimenata, odabrani su optimalni hiperparemetri kao i optimalna arhitektura RMS-Net te se svi rezultati u nastavku odnose upravo na to.

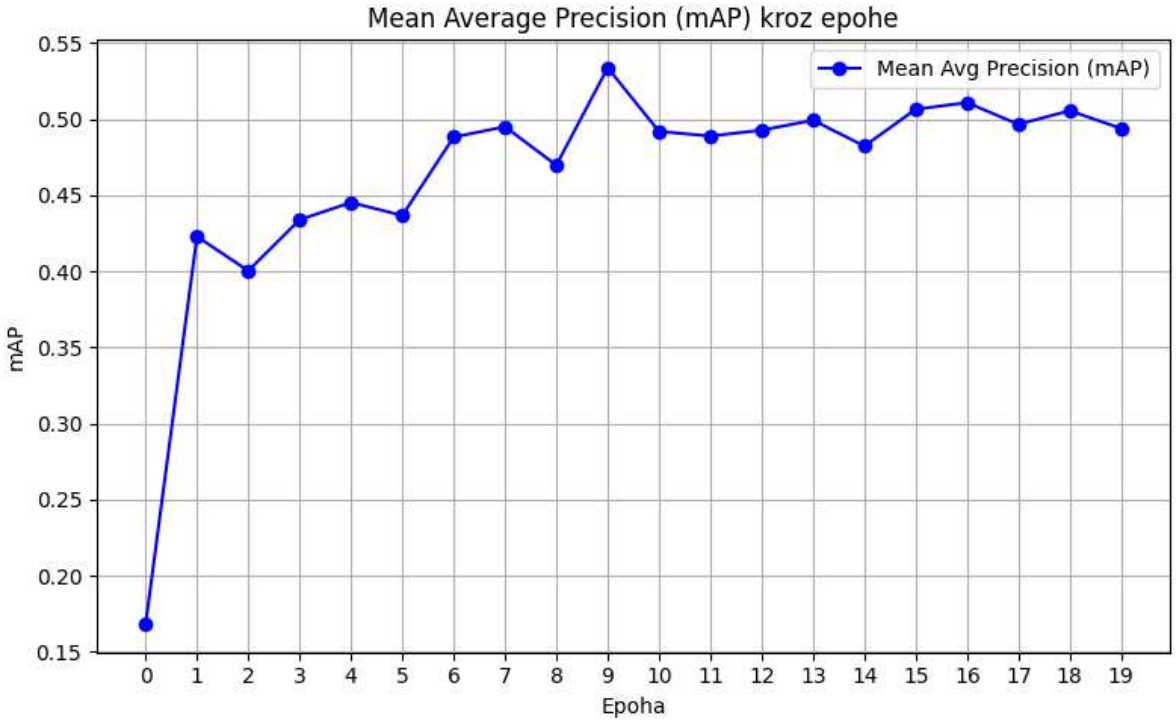
Najprije, na slici (Slika 4.1) je prikazano je kretanje konačne funkcije gubitka RMS-Net modela na skupu za učenje i skupu za validaciju kroz 20 epoha . Funkcija gubitka se prema formuli (3) sastoji od klasifikacijskog gubitka i otežanog regresijskog gubitka za $\lambda = 10$. Na temelju grafova prikazanih na slici (Slika 4.1) može se vidjeti kontinuirani pad funkcije gubitka na skupu za učenje što sugerira da model uči sve bolje i bolje, međutim to nas vrlo lagano može odvući u prenaučenost (engl. *overfitt*). Što se tiče validacijskog gubitka, on u početku opada do 5. epohe nakon čega blago poraste i stagnira. Ovime bi se moglo zaključiti da bi se proces učenja mogao prekinuti već nakon 5 epoha, međutim model je postigao najbolji rezultat u 9. epohi kao što ćemo vidjeti u nastavku.



Slika 4.1 Funkcija gubitka nad skupom za učenje i validaciju

Tijekom provođenja eksperimenata, isprobane su regularizacijske tehnike poput slučajnog odbacivanja neurona (engl. *dropout*), ranijeg zaustavljanja i slično. Međutim, rezultati na validacijskom i testnom skupu u tom slučaju su bili lošiji te je odlučeno da se neće koristiti.

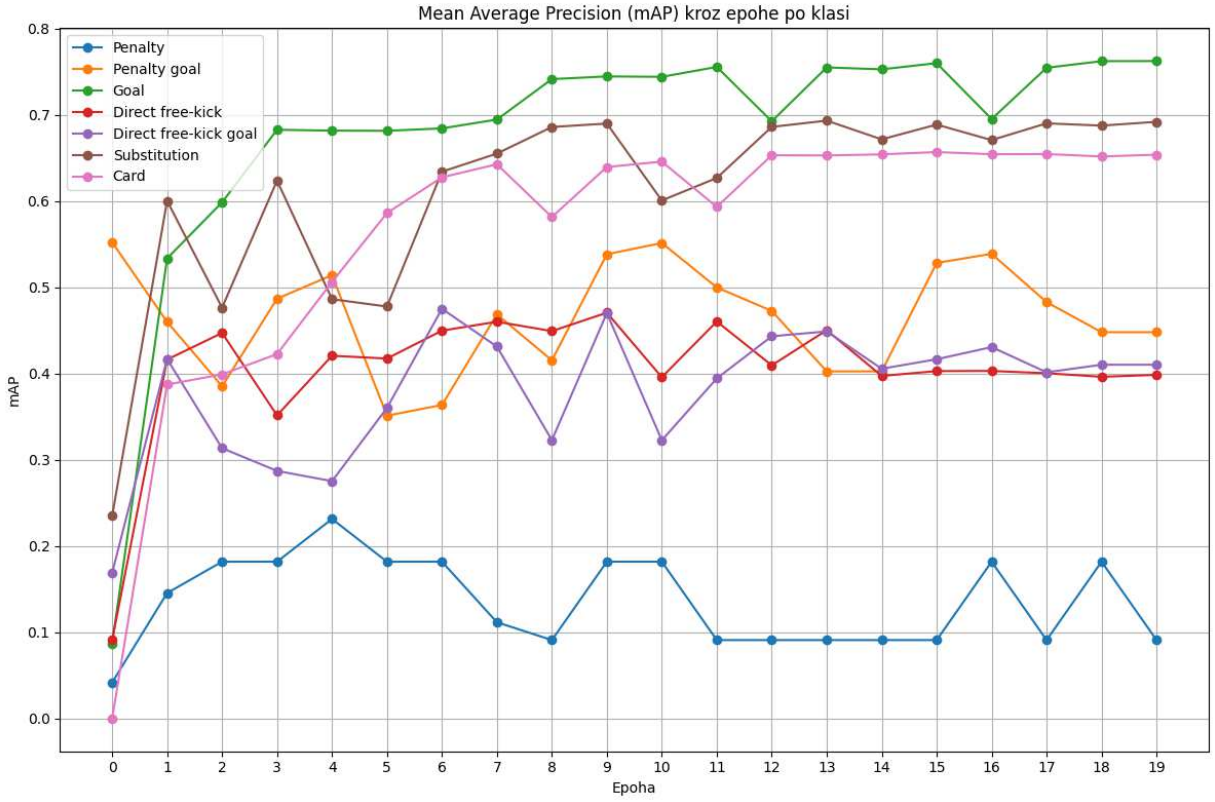
Sljedeće ćemo razmotriti kretanje najbitnije metrike u ovom radu na validacijskom skupu, a to je srednja prosječna preciznost. Graf kretanja srednje prosječne preciznosti je prikazan na slici (Slika 4.2). Možemo vidjeti kontinuirani rast te metrike kroz epohe, no isto tako i da je najveća vrijednost dosegnuta u epohi 9, a ona je iznosila **mAP = 0.5337**. S obzirom da se vidi uzlazni trend, isproban je proces i na 30 epoha, međutim srednja prosječna preciznost je stagnirala nakon 20. epohe, a i na temelju grafa funkcija gubitka na slici (Slika 4.1) može se zaključiti da nema smisla dalje učiti model. U tablici (Tablica 4.) prikazani su rezultati modela u ovom radu te modela autora članka [2]. Bitna je razlika što su autori članka model učili i testirali na 3 klase (*„Goal“, „Substitution“, „Card“*) dok je model u ovom treniran na spomenutih 7 klasa. Naravno, u oba slučaja nalazi se i klasa *„background“,* ali s obzirom da te video sekvence ne sadrže nogometnu akciju, ta klasa ne ulazi u računanje srednje prosječne preciznosti.



Slika 4.2 Mjera srednje prosječne preciznosti na skupu za validaciju

S obzirom na rečeno, za očekivati je da će model koji mora razlikovati 7 klasa imati nešto lošije metričke rezultate.

Kako bi provjerili kako se model ponaša na pojedinim klasama, na slici (Slika 4.3) je prikazano kretanje prosječne preciznosti na validacijskom skupu kroz epohe za svaku klasu pojedinačno. Na temelju te slike mogu se donijeti zaključci o tome kako je model uspješno naučio detektirati pojedinu nogometnu akciju. Model je daleko najlošiji na izvedenoj klasi *„Penalty“,* gdje unutar video sekvence ne uspijeva najbolje detektirati akciju da je penal promašen. Model se također muči s razlikovanjem klasa *„Direct free-kick“* i *„Direct free-kick goal“,* što je u jednu ruku i razumljivo jer je nužno znati ishod akcije slobodnog udarca kako bi model mogao zaključiti o čemu se radi. U takvim situacijama zasigurno bi pomogao mehanizam pažnje unutar transformer modela. Po grafu na slici (Slika 4.3) se može vidjeti da je model najbolje rezultate dao upravo nad klasama *„Goal“, „Substitution“ i „Card“*, što su klase nad kojima su autori članka [2] učili svoj model.



Slika 4.3 Mjera prosječne preciznosti na skupu za validaciju po klasama

Tablica (Tablica 3.) prikazuje vrijednosti metrike kada bi pogledali prosječne preciznosti po klasama za vrijeme 9. epohe, kada je dobiven najbolji ukupni rezultat. Kada bismo pogledali rezultate ovog modela nad 3 klase nad kojima je testiran model autora članka [2], srednja prosječna preciznost za samo te 3 klase iznosila bi **mAP = 0.6915,** iako je bitno spomenuti da bi neke oznake bile označene kao *„Goal“* koje su ovdje označene kao *„Penalty goal“* ili *„Direct free-kick goal“*, međutim rezultat je ipak dovoljno konkurentan za usporedbu s rezultatom modela autora članka [2] koja se može vidjeti u tablici (Tablica 4).

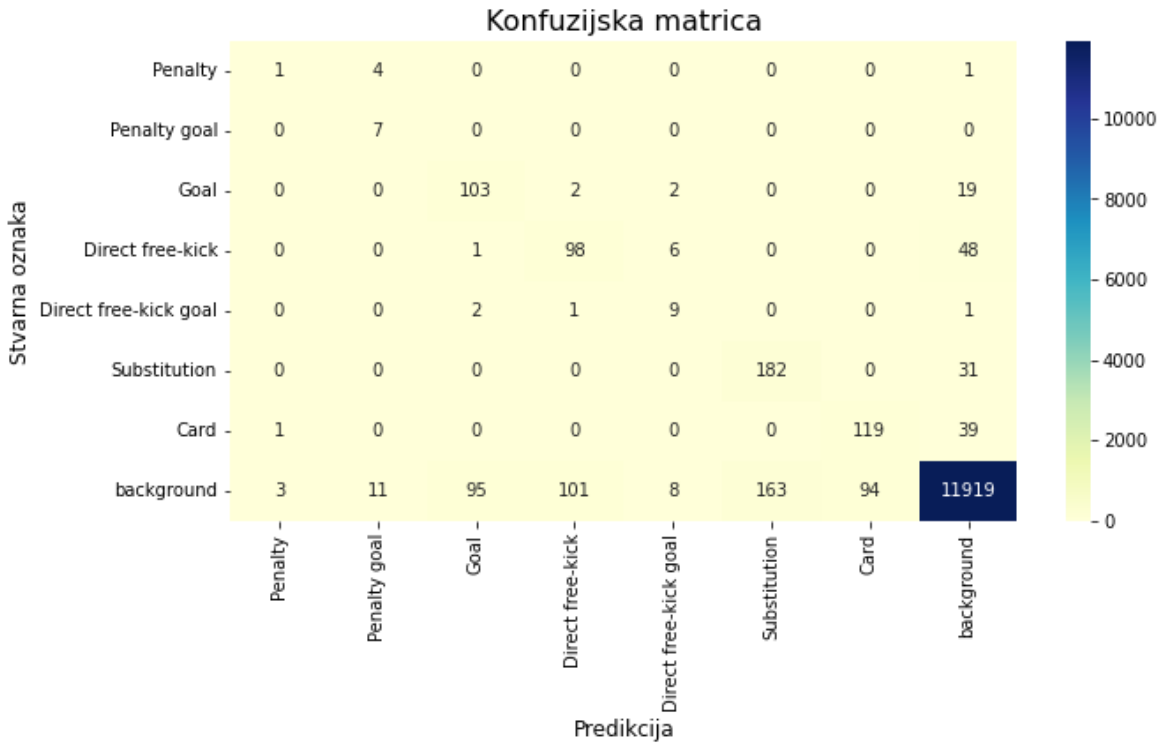
Tablica 3. Prosječna preciznost po klasama prilikom najbolje srednje prosječne preciznosti

Penalty	**Penalty goal**	**Goal**	**Direct free-kick**	**Direct free-kick goal**	**Substitution**	**Card**
0.1818	**0.5385**	**0.7448**	**0.4707**	**0.4702**	**0.6901**	**0.6397**

Kao kratak zaključak ovih rezultata možemo zaključiti da postoje klase koje je model naučio bolje detektirati, a postoje i one koje je naučio lošije klasificirati. Kod svih video sekvenci okolina pomaže u prepoznavanju akcija, no u klasama poput *„Penalty“* i *„Penalty goal“* detalj koji presuđuje je taj je li gol postignut ili nije. Isto vrijedi i za klase *„Direct free-kick“* i *„Direct free-kick goal“*. Model bi se u tim situacijama trebao fokusirati na ishod te akcije kako bi se moglo razlučiti o kojoj akciji se radi. Konvolucijski modeli tu imaju određeni problem te bi mehanizam pažnje tu vjerojatno pomogao. Međutim, bez obzira na to, ovaj pristup je znatno jednostavniji i resursno manje zahtjevan od transformer modela, a ipak donosi respektabilne rezultate.

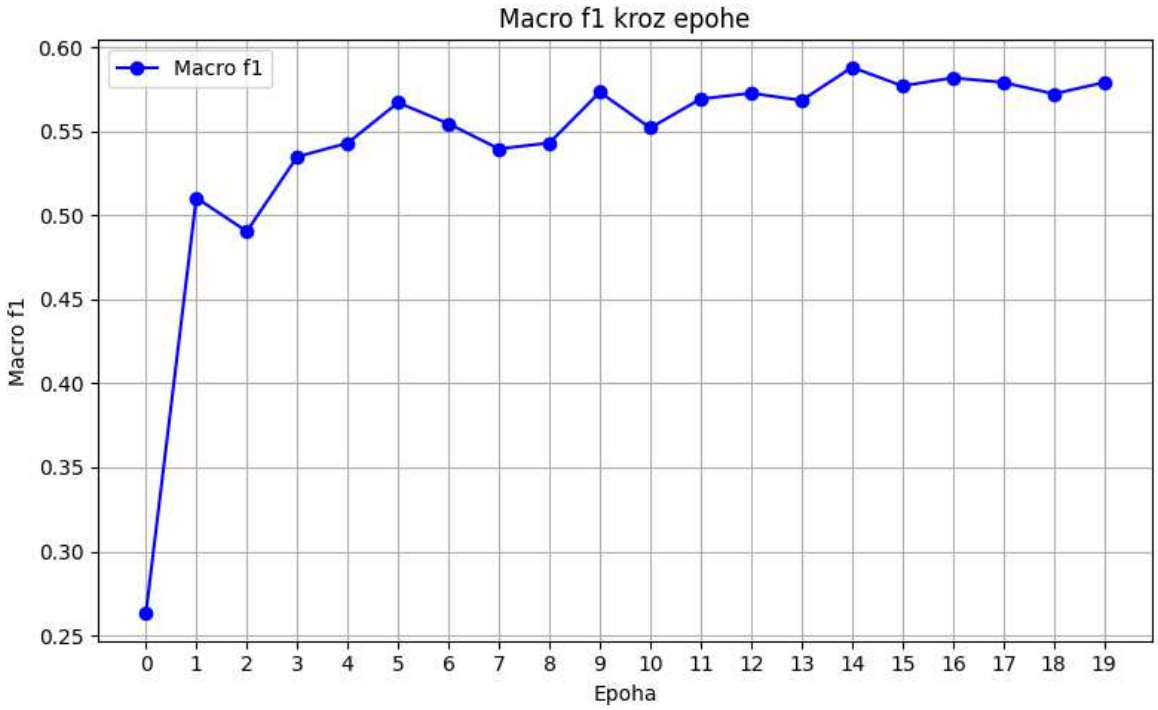
S obzirom da je dio ovog modela i klasifikacijski zadatak video sekvenci, koji je svakako bitan jer prije određivanja relativnog vremenskog pomaka, bitno je uopće prepoznati akciju te video sekvenci dodijeliti ispravnu klasu. Kako bi vidjeli kako se model nosi u klasifikacijskom dijelu, na temelju čega možemo donijeti i nove zaključke, na slici (Slika 4.4) prikazana je konfuzijska matrica klasifikacije video sekvenci na validacijskom skupu u jednu od 7 klasa akcija te dodatnu klasu *„background“*. Kao što možemo vidjeti, matrica je uglavnom dijagonalna što je željeni ishod. Ranije smo spomenuli problem razlučivanja klase *„Penalty“* od *„Penalty goal“* što ova konfuzijska matrica potvrđuje. Naime,

možemo vidjeti da je model 4 video sekvence koje bi trebale imati oznaku *„Penalty“*, klasificirao u klasu *„Penalty goal“*, dok je jednu video sekvencu označio kao *„background“*. S druge strane, svih 7 očekivanih oznaka *„Penalty goal“* klasificirao je upravo u klasu *„Penalty goal“*. Vidimo da je model solidno klasificirao video sekvence oznake *„Direct free-kick goal“*. Također, model je veliku većinu video sekvenci koje sadrže oznaku *„Goal“, „Card“* ili *„Substitution“* klasificirao točno.

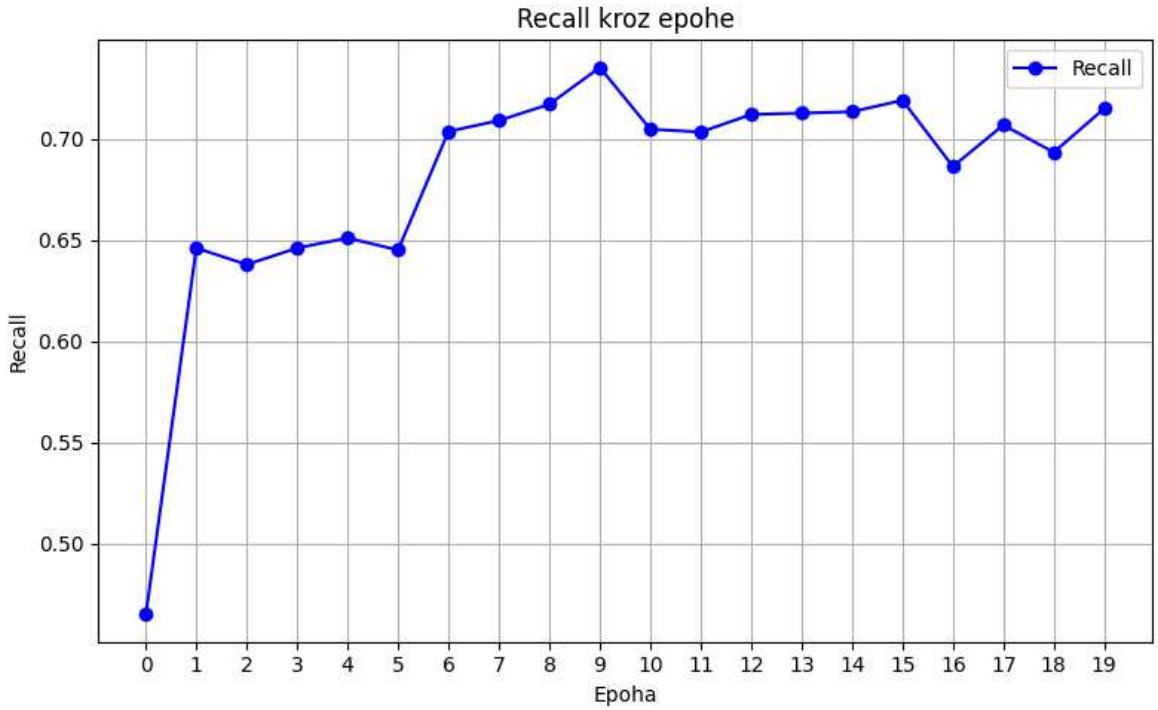


Slika 4.4 Konfuzijska matrica klasifikacije video sekvenci na validacijskom skupu podataka

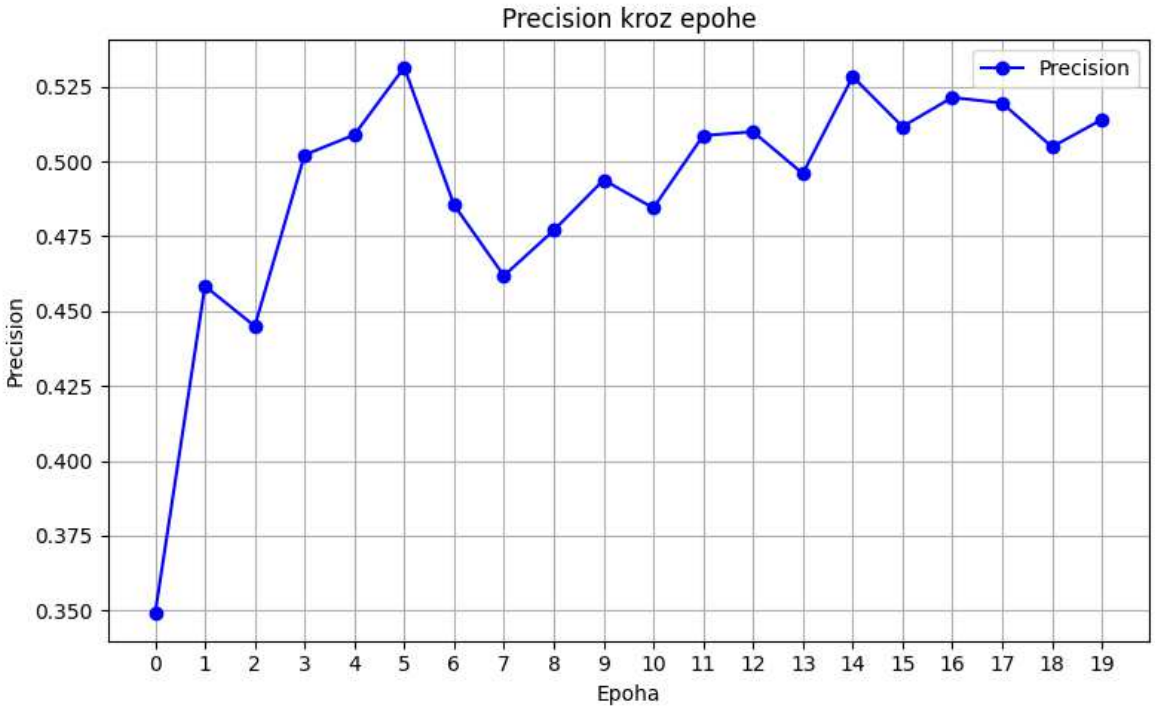
U sklopu učenja, praćene su i klasifikacijske mjere poput f1, preciznost i odziva [10]. S obzirom da se radi o većem broju klasa, kao f1 mjera odabrana je makro f1 mjera koja uzima f1 mjeru svake klase pojedinačno te radi prosjek tretirajući svaku klasu jednako. Također, odziv i preciznost su prikazani kao prosječne mjere po svim klasama. Kretanje makro f1 mjere kroz epohe učenja na validacijskom skupu prikazano je na slici (Slika 4.5), odziva na slici (Slika 4.6), dok je kretanje preciznosti kroz epohe učenja na validacijskom skupu prikazano je na slici (Slika 4.7).



Slika 4.5 Makro f1 mjera kroz epohe na validacijskom skupu



Slika 4.6 Odziv kroz epohe na validacijskom skupu



Slika 4.7 Preciznost kroz epohe na validacijskom skupu

Tablica 4. Usporedba rezultata RMS-Net modela u ovom radu i RMS-Net modela autora članka [2]

Model	Valid mAP	Test mAP
RMS-Net model iz ovog rada, testiran nad 7 klasa	**0.534**	**0.516**
RMS-Net model iz ovog rada, testiran nad 3 iste klase kao autora članka [2]	**0.692**	**0.654**
RMS-Net autora članka [2], testiran nad 3 klase	**0.678**	**0.655**

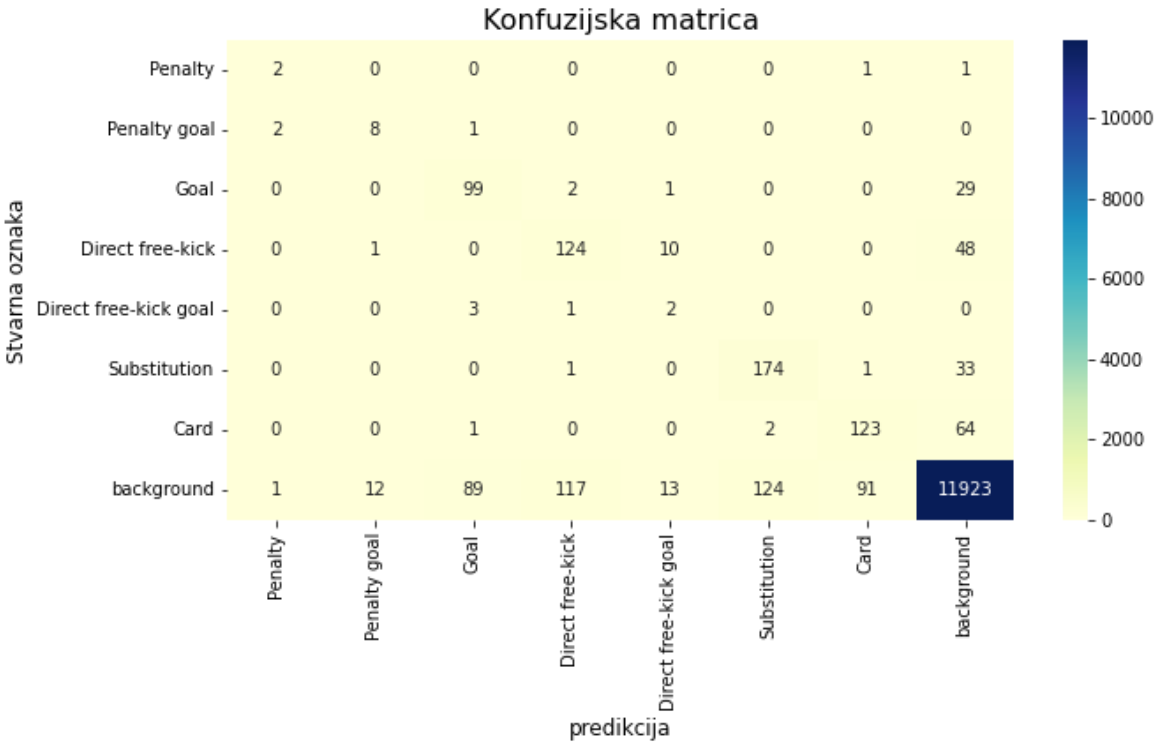
Na samom kraju rada, nužno je ispitati rad najboljeg modela na testnom skupu podataka, koji je odabran na temelju validacijskog skupa. Ideja kreiranja video sekvenci i ispitivanja ostaje ista kao što je to napravljeno i na validacijskom skupu podataka. U tablici (Tablica 4.) nalaze se rezultati modela na testnom skupu. Kao što vidimo, RMS-Net model razvijen u ovom radu, na svih 7 klasa daje solidan rezultat od **mAP = 0.516,** a ako bismo pogledali rezultat modela na 3 klase koje odgovaraju klasama nad kojima su autori članka [2] testirali model, vidimo da su rezultati gotovo jednaki. Taj rezultat u slučaju modela ovog rada izračunat je na način da su uzete prosječne preciznosti nad tri spomenute klase (*„Goal", „Substitution", „Card"*) te je napravljen njihov prosjek. U tablici (Tablica 5.) prikazane su prosječne preciznosti po klasama na testnom skupu podataka.

Tablica 5. Prosječna preciznost po klasama na testnom skupu podataka

Penalty	**Penalty goal**	**Goal**	**Direct free-kick**	**Direct free-kick goal**	**Substitution**	**Card**
0.4066	**0.6783**	**0.6586**	**0.5288**	**0.0364**	**0.7323**	**0.5718**

Na temelju prikazanih rezultata u tablici (Tablica 5.), vidimo da se model ponaša poprilično loše kod detektiranja akcije *„Direct free-kick goal"*, dok je na svim ostalim klasama na testnom skupu dao poprilično dobre rezultate. Kako bi provjerili način na koji model vrši klasifikaciju, na slici (Slika 4.8) je prikazana konfuzijska matrica klasifikacije video sekvenci na testnom skupu. U usporedbi s konfuzijskom matricom na validacijskom skupu, matrica izgleda poprilično slično. Vidimo da je model tek dvije video sekvence od šest, koje su označene kao *„Direct free-kick goal",* klasificirao ispravno. Klasifikacija preostalih video sekvenci izgleda poprilično zadovoljavajuće.

Na temelju konfuzijske matrice, moguće je prikazati jednake metrike koje su prikazane i na validacijskom skupu. Te metrike prikazane su u tablici (Tablica 6.).



Slika 4.8 Konfuzijska matrica klasifikacije video sekvenci na testnom skupu podataka

Tablica 6. Klasifikacijske mjere na testnom skupu podataka

Makro f1 mjera	Odziv	Preciznost
0.5655	**0.6797**	**0.5015**

Zaključak

Računalni vid i umjetna inteligencija svakodnevno prolaze nove uspone i primjene u raznim područjima znanosti, sporta i svakodnevnog života. Sport je danas neizostavan dio života velikog broja ljudi diljem svijeta te se stvaraju prilike za unos tehnologije u to područje kako bi se stvari olakšale i ubrzale. Jedan takav primjer je kreiranje video sažetaka iz dugačkih video zapisa cijelih nogometnih utakmica.

Ovaj rad predstavlja metodu koja može detektirati ukupno sedam ključnih nogometnih akcija unutar dugačkog videa te na temelju detektiranih akcija stvoriti sažetak i tako olakšati posao prolaska kroz brojne videozapise nogometnih utakmica i ručnog kreiranja sažetaka. Metoda se sastoji od dvije konvolucijske neuronske mreže, gdje prva pretvara ulazne slike u korisne značajke, dok druga prima sekvence tih korisnih značajki tražeći nogometnu akciju unutar primljene sekvence.

Ovakva metoda nudi brojne mogućnosti ponajviše zbog svoje resursne jednostavnosti. Naime, model RMS-Net, koji klasificira video sekvence i predviđa relativni vremenski pomak, zapravo je veoma mali model koji u ovom slučaju obavlja zadatak na solidan način i pokazuje zadovoljavajuće rezultate. Međutim, model ima manu što može detektirati samo jednu nogometnu akciju unutar video sekvence, iako se u kontekstu kreiranja video sažetaka nogometnih utakmica to ne pokazuje kao veliki problem zbog toga što su ključne akcije u nogometu poprilično rijetke gledajući cijeli videozapis.

Također, postoje modeli i metode koje bi mogle pomoći u poboljšavanju rezultata ovog zadatka kao što su transformer modeli koji primjenjuju mehanizam pažnje. S obzirom da se u ovom zadatku radi s video sekvencama koje imaju vremensku komponentu, mehanizam pažnje mogao bi pomoći da se model fokusira na različite dijelove ulazne sekvence s različitom jačinom kako bi se pronašli dijelovi koji su značajniji za obavljanje zadataka klasifikacije. U kontekstu ovog rada, to bi moglo pomoći kod klasificiranja video sekvenci u klase *„Penalty“, „Penalty goal“, „Direct free-kick“* i *„Direct free-kick goal“* kod kojih detalji čine razliku. Primjer upravo transformer arhitekure modela na ovom problemu izdan je u članku [11].

Literatura

[1] Adrien Deliège and Anthony Cioppa and Silvio Giancola and Meisam J. Seikavandi and Jacob V. Dueholm and Kamal Nasrollahi and Bernard Ghanem and Thomas B. Moeslund and Marc Van Droogenbroeck *SoccerNet-v2 : A Dataset and Benchmarks for Holistic Understanding of Broadcast Soccer Videos*, The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2021.

[2] Tomei, Matteo and Baraldi, Lorenzo and Calderara, Simone and Bronzin, Simone and Cucchiara, Rita *RMS-Net: Regression and Masking for Soccer Event Spotting*, 2020 25th International Conference on Pattern Recognition (ICPR), 2021., str. 7699-7706

[3] *SoccerNet Action Spotting,* SoccerNet, Poveznica: https://www.soccer-net.org/tasks/action-spotting; pristupljeno 20. lipnja 2024.

[4] Elharrouss, Omar and Akbari, Younes and Almaadeed, Noor and Al-Maadeed, Somaya, *Backbones-Review: Feature Extraction Networks for Deep Learning and Deep Reinforcement Learning Approaches*, arXiv:2206.08016, 2022. Poveznica: https://arxiv.org/abs/2206.08016

[5] He, Kaiming and Zhang, Xiangyu and Ren, Shaoqing and Sun, Jian, *Deep Residual Learning for Image Recognition*, arXiv:1512.03385, 2015. Poveznica: https://arxiv.org/abs/1512.03385

[6] Giancola, Silvio and Amine, Mohieddine and Dghaily, Tarek and Ghanem, Bernard. *SoccerNet: A Scalable Dataset for Action Spotting in Soccer Videos.* 1792-179210. 10.1109/CVPRW.2018.00223, 2018.

[7] Russakovsky, Olga and Deng, Jia and Su, Hao and Krause, Jonathan and Satheeshm, Sanjeev and Ma, Sean and Huang, Zhiheng and Karpathy, Andrej and Khosla, Aditya and Bernstein, Michael and C. Berg, Alexander and Fei-Fei, Li, *ImageNet Large Scale Visual Recognition Challenge* , arXiv:1512.03385, 2015. Poveznica: https://arxiv.org/pdf/1409.0575

[8] *SGD optimizer,* Pytorch, Poveznica: https://pytorch.org/docs/stable/generated/torch.optim.SGD.html; pristupljeno 20. lipnja 2024.

[9] Loshchilov, Ilya and Hutter, Frank, *SGDR: Stochastic Gradient Descent with Warm Restarts*, arXiv:1608.03983v5, 2016. Poveznica: https://arxiv.org/pdf/1608.03983v5

[10] Šnajder J., Vrednovanje modela, Zagreb: Nastavni materijali iz kolegija Strojno učenje 1, Fakultet elektrotehnike i računarstva, 2023.

[11] Denize, Julien and Liashuha, Mykola and Rabarisoa, Jaonary and Orcesi, Astrid and Hérault, Romain, *COMEDIAN: Self-Supervised Learning and Knowledge Distillation for Action Spotting using Transformers* arXiv:2309.01270, 2023. Poveznica: https://arxiv.org/pdf/2309.01270

[12] Logunova, Inna., *Deep Learning Applications for Computer Vision*, 2022. Poveznica: https://serokell.io/blog/deep-learning-for-computer-vision; pristupljeno: 24. lipnja 2024.

Sažetak

Cilj ovog rada bio je razviti i ispitati metodu koja može unutar videozapisa cijele nogometne utakmice detektirati ključne nogometne akcije s ciljem kreiranja video sažetaka nogometne utakmice. Metoda se sastoji od dvije konvolucijske neuronske mreže. Prvi dio čini arhitektura ResNet-152 koja iz ulaznog slijeda okvira (engl. *frames*) na izlazu daje slijed izdvojenih značajki. Sekvence tih značajki ulaze u drugi model imena RMS-Net koji za svaku sekvencu mora odrediti nalazi li se neka od ključnih nogometnih akcija unutar te sekvence radeći klasifikaciju sekvence u jednu od sedam klasa te za njih i predvidjeti relativni vremenski pomak kako bi znali trenutak u kojem se akcija dogodila. Ako model ne pronađe akciju unutar sekvence, tada tu sekvencu klasificira kao klasu pozadine koja ne ulazi u konačni video sažetak.

Ključne riječi: videozapis, konvolucijske neuronske mreže, ResNet-152, detektiranje nogometnih akcija

Summary

The aim of this thesis was to develop and test a method that can detect key soccer actions within a full match video to create video summaries of the soccer match. The method consists of two convolutional neural networks. The first part is a ResNet-152 architecture, which extracts a sequence of features from the input sequence of frames. The sequences of these features are fed into a second model named RMS-Net, which for each sequence must determine whether any of the key soccer actions are present within that sequence by classifying the sequence into one of seven classes and predicting the relative temporal offset to know the exact moment the action occurred. If the model does not find an action within the sequence, it classifies that sequence as a background class, which is not included in the final video summary.

Keywords: video, convolutional neural networks, ResNet-152, action spotting in football