Procjena smjera pogleda vozača temeljena na analizi slike

Dugonjevac, Dario

Master's thesis / Diplomski rad

2024

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Electrical Engineering and Computing / Sveučilište u Zagrebu, Fakultet elektrotehnike i računarstva**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:168:017426

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2025-03-23**

Repository / Repozitorij:

FER Repository - University of Zagreb Faculty of Electrical Engineering and Computing repozitory

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 413

PROCJENA SMJERA POGLEDA VOZAČA TEMELJENA NA ANALIZI SLIKE

Dario Dugonjevac

Zagreb, lipanj 2024.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 413

PROCJENA SMJERA POGLEDA VOZAČA TEMELJENA NA ANALIZI SLIKE

Dario Dugonjevac

Zagreb, lipanj 2024.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

Zagreb, 4. ožujka 2024.

DIPLOMSKI ZADATAK br. 413

Pristupnik:	**Dario Dugonjevac (0036523186)**
Studij:	Računarstvo
Profil:	Znanost o podacima
Mentor:	akademik prof. dr. sc. Sven Lončarić

Zadatak: **Procjena smjera pogleda vozača temeljena na analizi slike**

Opis zadatka:

Procjena smjera pogleda vozača je važna tema u području računalnog vida. Smjer pogleda vozača izrazito je bitan za sigurnost putnika u vozilu jer bilo kakva distrakcija vozača može dovesti do ozbiljnih prometnih nezgoda. U sklopu diplomskog rada potrebno je proučiti metode za procjenu smjera pogleda iz slika osoba koje se temelje na primjeni dubokog učenja. Potrebno je odabrati neku od metoda za procjenu smjera pogleda koja postiže rezultate bliske stanju tehnike i primjeniti ju na skup podataka koji čine slike vozača te pripadne oznake trodimenzionalnog vektora smjera pogleda. Odabranu metodu je potrebno ispitati na testnom skupu slika koristeći prikladnu metriku.

Rok za predaju rada: 28. lipnja 2024.

Ovaj diplomski rad posvećujem svojoj obitelji i prijateljima.
Hvala vam na svemu što ste mi pružili tijekom mog studija.

Zahvaljujem svima koji su mi pomogli pri izradi ovog rada svojim savjetima, preporukama i ostalim ne tako beznačajnim sitnicama, a posebno mom mentoru akademiku prof. dr. sc. Svenu Lončariću.

Sadržaj

1. Uvod **2**

2. Povezani radovi **4**

3. Skup podataka **7**

4. Modeli dubokog učenja **10**

4.1. Neuronska mreža 11

4.2. Konvolucijska neuronska mreža 12

4.3. Transformatori 14

4.4. Transformatori za obradu vizualnih podataka 17

5. Model **21**

5.1. Konvolucijski slojevi i ResNet-18 21

5.2. Transformator blokovi 23

5.3. Parametri modela 24

6. Hiperparametri **27**

7. Rezultati **31**

8. Zaključak **39**

Literatura **41**

Sažetak **45**

Abstract **46**

1. Uvod

Procjena smjera pogleda vozača bitno je područje istraživanja u domeni računalnog vida. Zbog velike količine informacija koje svake sekunde dolaze do nas tijekom vožnje, vrlo je lako izgubiti pozornost i ne obraćati pažnju na cestu. Ako postoji senzor u automobilu koji je u stanju detektirati pogled osobe, može dati obavijest vozaču koji će zbog toga vjerojatnije obratiti pozornost na cestu i smanjiti mogućnost prometne nesreće. Vozačeva pažnja i fokusiranost dok se nalazi za volanom u prometu veoma je bitna. Članak [1] iz 2018. godine tvrdi da je čak 10% smrti u američkoj saveznoj državi West Virginia u prometnim nesrećama uzrokovano rastresenom vožnjom. Ako osoba ne reagira pravovremeno, može doći do visoke materijalne štete, ali i do gubitka života. U cilju razvoja naprednih sustava potpore vozaču, u ovom diplomskom radu istraživat će se metoda za procjenu smjera vozačevog pogleda temeljena na analizi slika putem dubokog učenja.

Duboko učenje, kao grana strojnog učenja, svakodnevno pokazuje ogromne napretke i potencijale u razumijevanju i obradi velike količine podataka. Slike, kao skupina podataka koje će se koristiti u ovom diplomskom radu, svojom složenošću dodatno otežavaju rješavanje problema procjene smjera pogleda. Kombinirajući nove i napredne tehnike dubokog učenja s problemom procjene pogleda vozača na slikama, cilj ovog istraživanja je razvoj preciznijih i pouzdanijih metoda od prethodno postojećih koje će omogućiti automatsko praćenje vozačevog pogleda u stvarnom vremenu.

Osnovni koraci ovog istraživanja uključuju proučavanje postojećih metoda za procjenu smjera pogleda, odabir najprimjerenije metode te njenu primjenu na skupu podataka koji sadrži slike vozača i pripadne oznake trodimenzionalnog vektora smjera pogleda. Nakon toga, metoda će biti testirana na testnom skupu slika. Pritom će biti korištena odabrana metrika za procjenu točnosti modela. Rezultati će biti uspoređeni s postojećim modelima i njihovim rezultatima.

Ovaj rad ima za cilj pružiti doprinos razumijevanju i razvoju tehnologija koje mogu poboljšati sigurnost u prometu kroz sustave za praćenje vozačeva pogleda. Kroz primjenu naprednih tehnika dubokog učenja, očekuje se da će rezultati ovog istraživanja biti korisni za daljnji razvoj sustava za pomoć vozaču te unapređenje sigurnosti u cestovnom prometu.

2. Povezani radovi

Detekcija smjera pogleda problem je dubokog učenja koji se već duže vrijeme pokušava riješiti. Rad [2] jedan je od radova koji objašnjava postojeće metode i prethodne radove napisane u svrhu rješavanja problema detekcije pogleda. U tome radu moguće je vidjeti napredak metoda i modela koji su se koristili i stvarali kroz vrijeme.

Kao što je rečeno u [2]:"Rane metode procjene pogleda otkrivaju obrasce pokreta očiju kao što su fiksacija, trzaji i glatka potraga." [3]

Nakon toga, razvojem tehnologija i računalnog vida koriste se kamere koji prate pogled osoba. Metode koje koriste takav pristup mogu se podijeliti u 3 kategorije:

- Metode regresije 2D značajki oka: uči funkciju preslikavanja od geometrijske značajke do točke pogleda. [4, 5, 6]
- Metode oporavka 3D modela oka: gradi geometrijske modele oka specifične za subjekt za procjenu smjerova ljudskog pogleda [7, 8, 9]
- Metode temeljene na izgledu: izravno uče funkciju mapiranja od slika do ljudskog pogleda. [10, 11]

Prvi korak prema korištenju dubokih modela [10] koristio je jednostavan model konvolucijske neuronske mreže te su uspjeli ostvariti bolje rezultate od većine konvencionalnih pristupa u to vrijeme.

Današnji suvremeni (eng. state-of-the-art) modeli koji ostvaruju najbolje rezultate u detekciji pogleda osoba su:

- Gaze360: predstavlja inovativan skup podataka i metodologiju za procjenu pogleda izvan osi (off-axis gaze estimation) koristeći panoramske slike snimljene s više ka-

mera, omogućujući preciznije praćenje pogleda u prirodnim okruženjima. [12]

- Multi-Zoom Gaze: uvodi novu metodu koja koristi višerazinske (multi-scale) konvolucijske neuronske mreže za precizno praćenje pogleda, omogućujući poboljšanu točnost i robusnost u različitim uvjetima promatranja. [13]

- L2CS: predstavlja napredni pristup za praćenje pogleda koji kombinira gubitak od centra do sfere (L2 loss) i sferičnu regresiju za postizanje visoke točnosti u određivanju smjera pogleda. [14]

Detekcija pogleda vozača problem je koji postoji već duže vrijeme. [15, 16, 17] Iako rad [15] možda ne pruža najbolje rezultate, dostavio je skup podataka koji svojim velikim obujmom može biti prekretnica u problemu detekcije pogleda vozača. Taj skup podataka je ujedno i skup podataka korišten u ovome diplomskome radu.

Rad [17] koristio je dvije kamere te kombinirajući slike lica i očiju s obje kamere radio klasifikaciju na jedan od 15 prethodno definiranih točaka gledanja vozača. Također, rad [16] koristio je sličan pristup, ali koristeći jednu NIR kameru (eng. Near InfraRed). Oba rada koristila su konvolucijsku neuronsku mrežu za dobivanje rezultata.

Prvi rad koji je stvorio današnje modele transformatora i koji je započeo novo doba dubokog učenja je "Attention is all you need". [18] Današnji modeli transformatora baziraju se na metodama i arhitekturom koji su proizašli iz ovoga rada.

Osim toga, značajan je i rad [19] koji je prvi objavio korištenje transformatora za obradu vizualnih podataka. Ovim radom konvolucijski modeli dobili su ozbiljnog protivnika u današnjim problemima obrade slika.

Rad na kojem se bazira ovaj diplomski rad zove se "Gaze Estimation using Transformer", ili procjena pogleda koristeći transformator. [20]. Taj rad je inovativni znanstveni članak koji koristi transformacijske modele za praćenje pogleda, značajno unapređujući točnost i robusnost u odnosu na tradicionalne metode. Model se sastoji od vizualnog enkodera i dekodera pogleda, koristeći mehanizme pažnje za fokusiranje na relevantne dijelove slike. Evaluacija na skupovima podataka poput GazeCapture i MPIIGaze pokazala je superiorne performanse modela korištenog u [20] u različitim uvjetima osvjetljenja i pozadinskog šuma. Transformatori omogućuju bolju integraciju konteksta unutar slike,

poboljšavajući generalizaciju na neviđene podatke. Ovaj rad otvara nove smjerove istraživanja u praćenju pogleda, ukazujući na potencijal naprednih modela dubokog učenja u računalnom vidu.

3. Skup podataka

Skup podataka korišten u članku "Look Both Ways" [15] sadrži detaljne podatke o pogledima vozača i vizualnim značajkama cestovnih scena. Podaci su prikupljeni pomoću kamera montiranih unutar vozila, koje snimaju lice vozača i cestu ispred vozila.

Skup podataka sadrži informacije o 28 različitih vozača čiji se pogled i istaknutost ceste (eng. saliency) periodično spremaju kao slike i informacije u tekstualnim datotekama. Svaka osoba sadrži više tisuća slika i podatkovnih informacija o pogledu i istaknutosti.

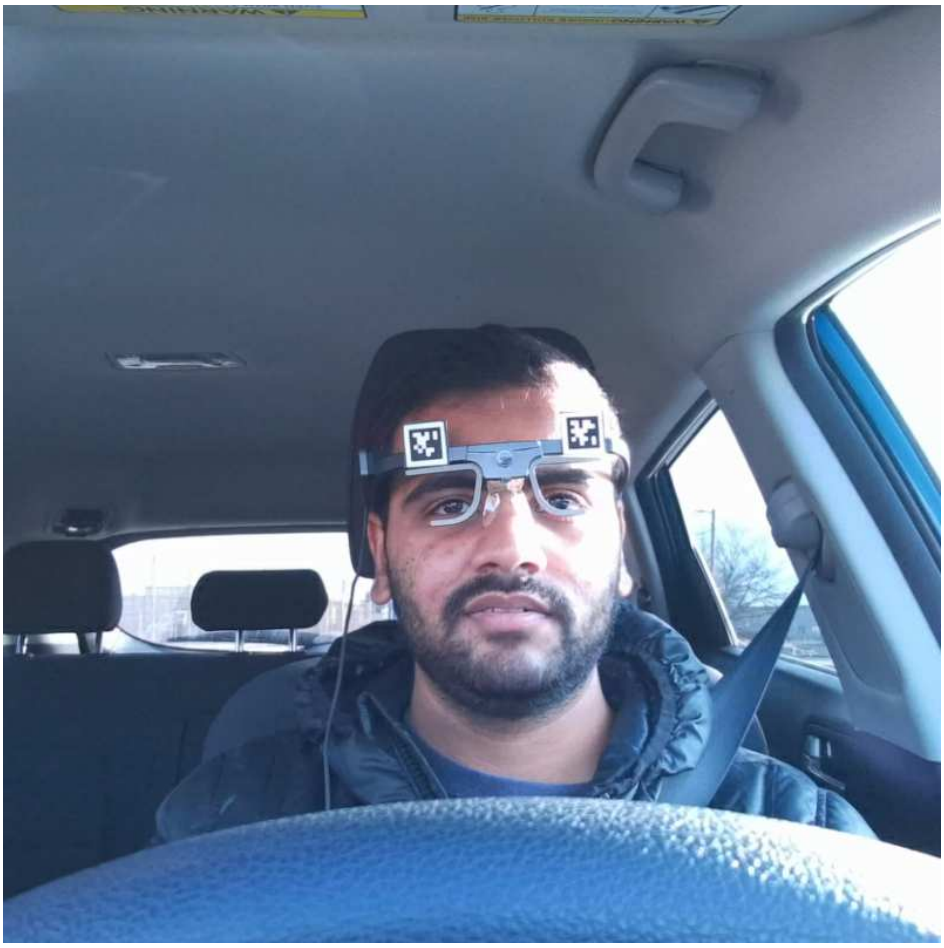
Atribut	**Vrijednosti**
Gaze_Loc_2D	[294, 385]
Gaze_Loc_3D	[1.08163943, -0.7400649, -15.73993926]
Right_Gaze_Dir	[0.06651897, -0.04568472, -0.99673875]
Right_2D_Eye_Loc	[382.59, 427.752]
Right_3D_Eye_Loc	[-0.01171911, 0.01084548, 0.64325006]
Left_Gaze_Dir	[0.06261185, -0.04553521, -0.99699865]
Left_2D_Eye_Loc	[475.76, 422.633]
Left_3D_Eye_Loc	[0.05297367, 0.00804434, 0.64000006]

Tablica 3.1. Tablica s podacima o položaju očiju i pogleda

Slike 3.1. i 3.2. predstavljaju primjer para slike lice i slike scene koji se nalaze u danom skupu podataka. Na slici lica osobe vidimo da vozač ima naočale uz pomoć kojih se precizno mjeri vektor njegovog pogleda.

Tablica 3.1. daje primjer izgleda tekstualne datoteke sa informacijama o položaju očiju i pogleda. Atributi objašnjavaju sljedeće stvari:

- **Gaze_Loc_2D:** Koordinata točke u koju se gleda u slici scene.
- **Gaze_Loc_3D:** Koordinata točke u koju se gleda u 3D sceni.



Slika 3.1. Primjer slike lica

- **Right_Gaze_Dir:** Normaliziran vektor smjera pogleda iz desnog oka.
- **Right_2D_Eye_Loc:** Koordinata desnog oka u slici lica.
- **Right_3D_Eye_Loc:** Koordinata desnog oka u 3D sceni snimke lica.
- **Left_Gaze_Dir:** Normaliziran vektor smjera pogleda iz lijevog oka.
- **Left_2D_Eye_Loc:** Koordinata lijevog oka u slici lica.
- **Left_3D_Eye_Loc:** Koordinata lijevog oka u 3D sceni snimke lica.

Za potrebe ovog diplomskog rada neće se koristiti slike scene, pa samim time se neće koristiti ni neki od atributa koji se koriste za detekciju istaknutosti scene.

Slika 3.2. Primjer slike scene

4. Modeli dubokog učenja

Prije nego što pređemo na razgovor o modelu dubokog učenja koji se koristi u ovom radu, moramo prvo da objasniti što je strojno učenje, svojevrsna osnova dubokog učenja. Naime, znanost o strojnom učenju je grana istraživanja iz dijela umjetne inteligencije, a konkretno se bavi razvojem algoritama i tehnika koji omogućavaju računalu da uči iz prethodno datih podataka bez eksplicitnog programiranja.

Strojno učenje omogućuje računalima prilagođavanje vlastitog ponašanja na temelju iskustva, a ne samo na temelju unaprijed definiranih pravila. To se postiže analizom velikih količina podataka kako bi se identificirali uzorci i donijele zaključci, čime se omogućuje računalima da donose odluke, predviđaju buduće događaje ili obavljaju zadatke bez eksplicitne ljudske intervencije.

Duboko učenje je grana strojnog učenja koja se koristi za rješavanje problema unutar područja umjetne inteligencije. Koristi niz nelinearnih transformacija kako bi naučio reprezentaciju podataka. Najčešće se primjenjuje u područjima gdje je dimenzionalnost podataka izuzetno visoka.

Primjena strojnog učenja je široka i obuhvaća mnoga područja kao što su medicina, financije, marketing, transport, industrija, sigurnost, i mnoga druga. Na primjer, u medicini se koristi za dijagnostiku bolesti na temelju medicinskih slika ili analize pacijentovih podataka, dok se u financijama koristi za analizu tržišta i predviđanje trendova. U transportu se koristi za optimizaciju rute i upravljanje prometom, dok se u industriji primjenjuje za poboljšanje procesa proizvodnje i održavanja.

Nakon definicije dubokog učenja, potrebno je objasniti pojmove vezane uz duboko učenje i sami model koji je korišten. Ti pojmovi su: neuronska mreža, konvolucijska neuronska mreža, transformator i transformator za obradu vizualnih podataka.

4.1. Neuronska mreža

Započnimo s neuronskom mrežom. Neuronska mreža predstavlja osnovni primjer modela dubokog učenja koji, kroz upotrebu nelinearnih transformacija, mijenja podatke na način koji ih čini lakše interpretiranim.

U ovom kontekstu, interpretacija podataka podrazumijeva treniranje modela, odnosno postizanje optimalnog modela s odgovarajućim težinama za zadani skup podataka.

U potpuno povezanim neuronskim mrežama, operacija koja se koristi za izračunavanje vrijednosti jednog neurona izgleda ovako:

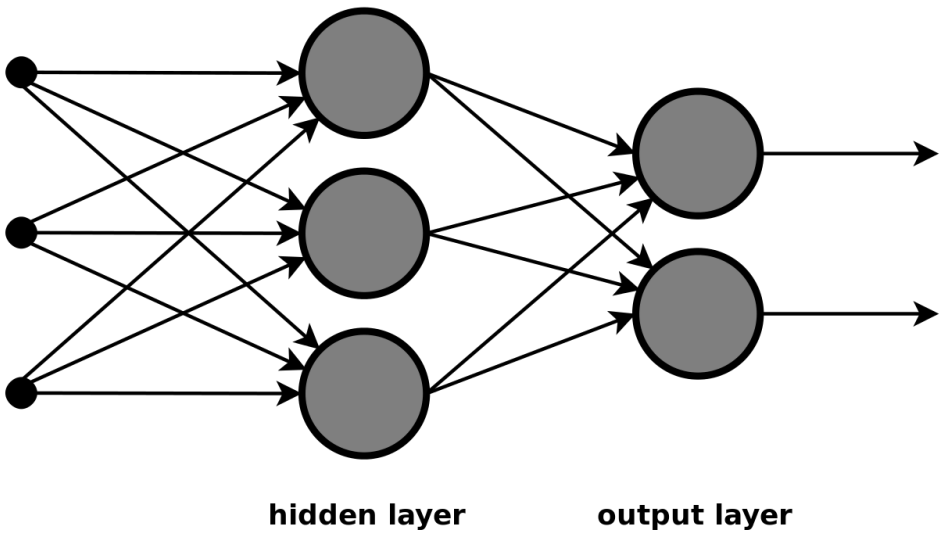
$$y_j = \sum_{i=1}^{D} w_{ji} \cdot x_i + w_{j0}, \qquad j = 1, ..., M \tag{4.1}$$

U ovoj jednadžbi, j predstavlja sloj modela, x predstavlja ulaze prethodnih slojeva, a w predstavlja težine koje se uče tijekom treniranja modela. Uzmemo li za primjer problem klasifikacije na dvije klase, cilj neuronske mreže je pronaći težine s pomoću kojih će moći savršeno razlučiti te dvije klase u prostoru podataka.

Zbog nepotpunosti modela i podataka, često je nemoguće postići takvu razinu točnosti modela. Zbog toga se obično traži model koji zadovoljava ravnotežu između složenosti i točnosti. Model ne bi trebao biti prekompleksan jer se tada previše prilagođava podacima, što u većini slučajeva dovodi do manje točnosti kada mu se daju novi, prethodno neviđeni podaci.

Na slici 4.1. prikazana su 3 različita sloja: ulazni sloj, skriveni sloj te izlazni sloj. Ulazni sloj, predstavljen crnim točkicama, je ulazni podatak neuronske mreže. Skriveni sloj (eng. hidden layer) sastoji se od više slojeva čija je uloga transformacija podataka. Izlazni sloj (eng. output layer) je posljednji sloj neuronske mreže s pomoću kojeg se donosi odluka o rezultatu klasifikacije ili regresije.

Neuronske mreže imaju izuzetno široku primjenu i neizostavan su dio dubokog učenja. Velika većina drugih modela dubokog učenja na neki način koristi potpuno povezanu unaprijednu neuronsku mrežu. Na primjer, konvolucijske neuronske mreže ko-



Slika 4.1. Jednostavna neuronska mreža [21]

riste neuronsku mrežu nakon konvolucijskih slojeva kako bi dobile rezultate iz prostora značajki koji nastaje konvolucijskim slojevima.

4.2. Konvolucijska neuronska mreža

Konvolucijska neuronska mreža (skraćeno CNN) predstavlja primjer modela dubokog učenja koji koristi operaciju konvolucije. Konvolucija kao operacija izražena je sljedećom formulom:

$$f(t) * g(t) = \int_{-\infty}^{\infty} f(\tau) \cdot g(t - \tau), d\tau \tag{4.2}$$

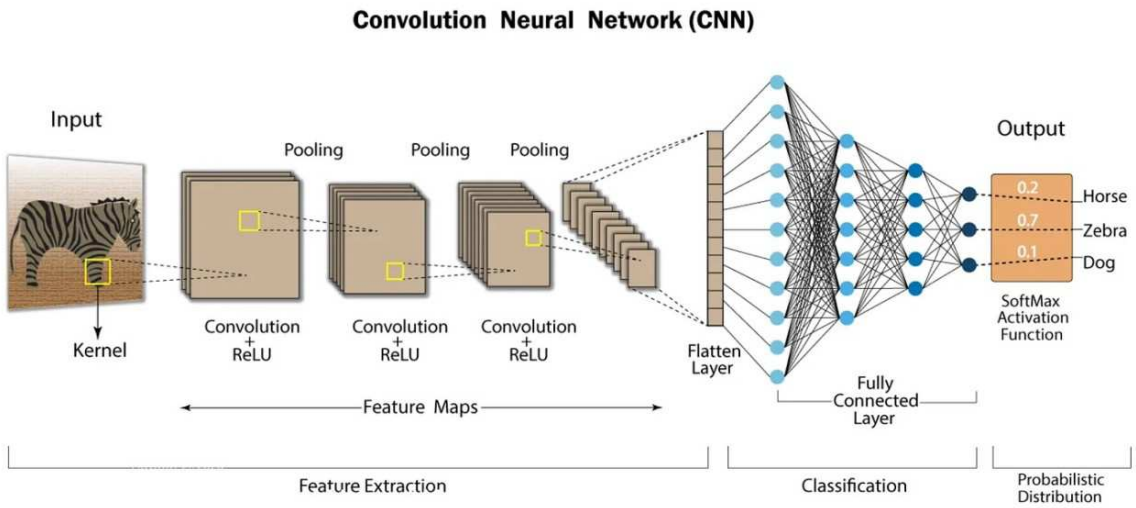
U formuli, f i g predstavljaju dvije funkcije nad kojima se računa konvolucija. S obzirom na dimenziju, postoji više vrsta konvolucija, a za slike se najčešće koriste 2D konvolucije. Formula za 2D konvoluciju izgleda ovako:

$$(f * g)(x, y) = \iint_{-\infty}^{\infty} f(u, v) \cdot g(x - u, y - v), du, dv \tag{4.3}$$

Kao što se može vidjeti, pomak se događa u dvije dimenzije (inducirano dvama varijablama), a ne samo u jednoj.

Konvolucijska neuronska mreža sastoji se od dva dijela: konvolucijsko sloja i pot-

puno povezanog sloja. Konvolucijski sloj označava dio modela koji koristi kombinaciju konvolucije, aktivacijske funkcije te funkcije sažimanja. Taj dio modela zadužen je za prethodno spomenute nelinearne transformacije podataka u svrhu boljeg razumijevanja. Potpuno povezani sloj zadužen je za stvaranje rezultata modela. Ukoliko se radi o višeklasnoj klasifikaciji, model će nakon određenog broja potpuno povezanih slojeva imati izlazni sloj koji ima jednak broj izlaznih neurona koliko ima i klasa. Primjer konvolucijske neuronske mreže prikazan je na slici ispod.



Slika 4.2. Konvolucijska neuronska mreža [22]

Na slici 4.2. postoje 3 glavna dijela: ekstrakcija značajki, klasifikacija te distribucija vjerojatnosti. Ekstrakcija značajki je postupak koji se odvija u konvolucijskom sloju kojim se dobivaju značajke potrebne za klasifikaciju objekta (ili regresiju). Rezultat konvolucijskog sloja naziva se polje značajki. Ono predstavlja višedimenzionalno polje (najčešće trodimenzionalno u slučajevima obrade slika) u kojem se nalaze karakteristike slike dobivene nelinearnim transformacijama.

Klasifikacija je postupak koji se odvija u potpuno povezanom sloju konvolucijske neuronske mreže kojim se dolazi do vjerojatnosti za pojedini objekt. Ako, na primjer, imamo 10 različitih objekata koje se želi klasificirati, rezultat tog sloja bit će 10 brojeva čiji je međusobni zbroj jednak 1 te pojedini broj predstavlja vjerojatnost pojedinog objekta.

Distribucija vjerojatnosti predstavlja rezultat klasifikacijskog sloja kojim donosimo odluku o objektu na jednostavan način: objekt s najvećom vjerojatnošću se predstavlja kao rezultat klasifikacije. Na slici 4.2. je to objekt *zebra* s vjerojatnošću od *0.7*.

Konvolucijske neuronske mreže imaju veliko područje primjene u računalnom vidu, gdje se vrlo često na ulaz dubokog modela daje slika. Osim mogućnosti klasifikacije, vrlo često CNN modeli imaju mogućnost i detekcije objekata na slikama, zbog čega su vrlo dominantni u današnjem svijetu računalnog vida.

4.3. Transformatori

Novija skupina modela dubokog učenja, popularizirana kroz potrebu za boljim sekvencijskim modelima dubokog učenja, su transformatori.

Mehanizam pozornosti prvi put se pojavljuje 2017. godine u članku pod nazivom "Attention is all you need" [18]. U tome članku, kako je već prethodno rečeno, pokazano je kako je moguće koristiti mehanizam pozornosti u transformatorima u svrhu prevođenja teksta, čime su poboljšani današnji prevoditelji.

Transformatori su arhitektura dubokog učenja koja je postala izuzetno popularna, posebno u obradi prirodnog jezika. Ova arhitektura se pokazala veoma uspješnom u različitim zadacima kao što su strojno prevođenje, generiranje teksta, odgovaranje na pitanja i još mnogo toga. Ključni elementi transformatora su slojevi mehanizma pažnje.

Osnovne karakteristike transformatora su:

- **Mehanizam pažnje**: Ovo je srce transformatora. Omogućava modelu da se fokusira na različite dijelove ulaznih podataka tijekom obrade. Umjesto da obradi cijelu sekvencu podataka u istom koraku, mehanizam pažnje omogućava modelu da pridaje veću težinu određenim dijelovima sekvence. To omogućava bolje razumijevanje odnosa između različitih dijelova sekvence.

- **Višeslojni modeli**: Transformatori su obično sastavljeni od više slojeva. Svaki sloj obično sadrži mehanizam pažnje, praćen slojevima potpuno povezanih mreža. Više slojeva omogućava modelu da nauči složenije obrasce u podacima.

- **Ulazna i izlazna reprezentacija ugrađivanja**: Ulazni podaci (npr. riječi u rečenici) i izlazni podaci (npr. kategorije) obično se predstavljaju kao vektorski ugrađeni prostori. Ovi vektori omogućavaju modelu da "razume" semantičke veze između različitih riječi ili tokena.

- **Maskiranje**: Maskiranje se koristi u transformatorima kako bi se određeni dijelovi sekvence sakrili od modela tijekom treniranja. Na primjer, tijekom treniranja modela za generiranje teksta, model ne smije vidjeti buduće tokene kako bi mogao generirati sljedeću riječ.

- **Samopouzdanje**: Transformatori koriste mehanizam samopouzdanja da bi odredili značajnost različitih dijelova ulazne sekvence. Svaki dio sekvence (npr. riječ ili token) može međusobno komunicirati i odlučiti koliko pažnje treba posvetiti drugim dijelovima sekvence.

Transformatori su značajno unaprijedili performanse u mnogim NLP zadacima, a dodatno su modificirani i prilagođeni za druge vrste podataka i zadatke, poput slike i audio obrade. Njihova modularna arhitektura, koja se oslanja na mehanizam pažnje, omogućava veću paralelizaciju i efikasnije učenje, što ih čini veoma efikasnim za obradu velikih skupova podataka.

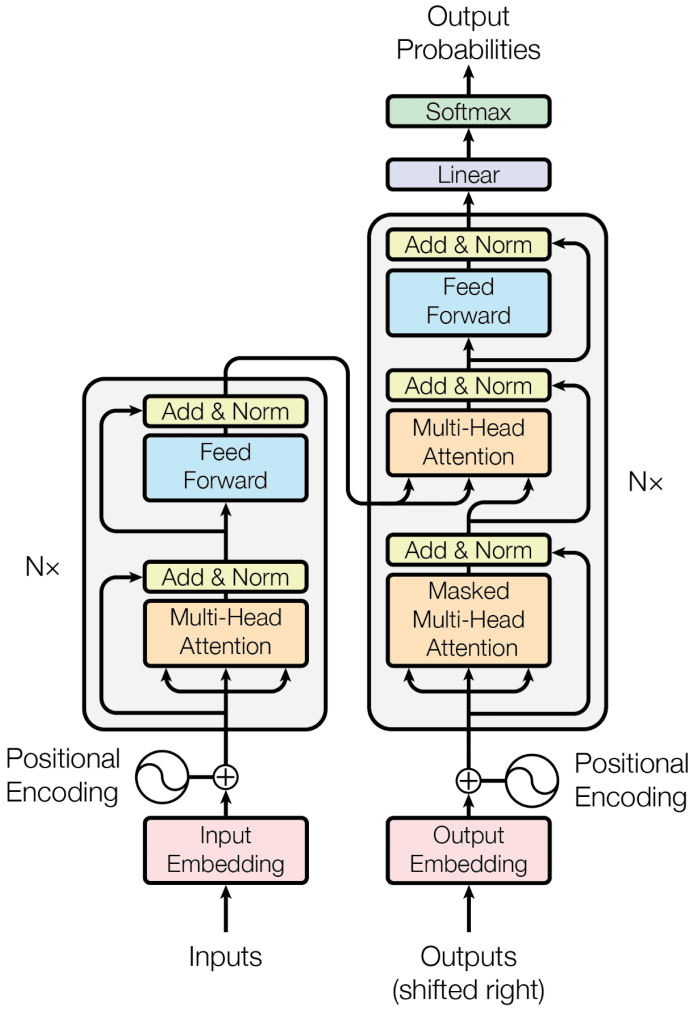
Slika 4.3. prikazuje primjer transformatora. Ulaz transformatora prvo prolazi kroz ugrađivanje reprezentacije ulaza. Razlog tome je što ulaz transformatora mogu biti i riječi, ali za sami model potrebna je reprezentacija u brojevima. Zbog veličine rječnika i drugih mogućih ulaza najčešće se radi o vektorima velikih dimenzija.

Nakon reprezentacije ulaza dodaje se reprezentacija pozicije koja je, kao i reprezentacija ulaza, unaprijed određena te ona služi za dodavanje informacije o poziciji ulaznih tokena unutar sekvence.

Nakon obrade ulaza slijedi sam model transformatora. Model transformatora podijeljen je na enkodere i dekodere.

Enkoder je komponenta transformatora koja uzima ulaznu sekvencu i proizvodi skriveni niz reprezentacija. Sastoji se od jednog ili više slojeva, od kojih svaki sloj uključuje: mehanizam samopozornosti, sloj potpuno povezanih mreža te normalizacija i preskakanje.

Dekoder je komponenta transformatora koja generira izlaznu sekvencu iz niza skrivenih reprezentacija stvorene od strane enkodera i prethodnih tokena izlazne sekvence. Dekoder također sadrži nekoliko slojeva, s tim da svaki sloj uključuje: mehanizam sa-



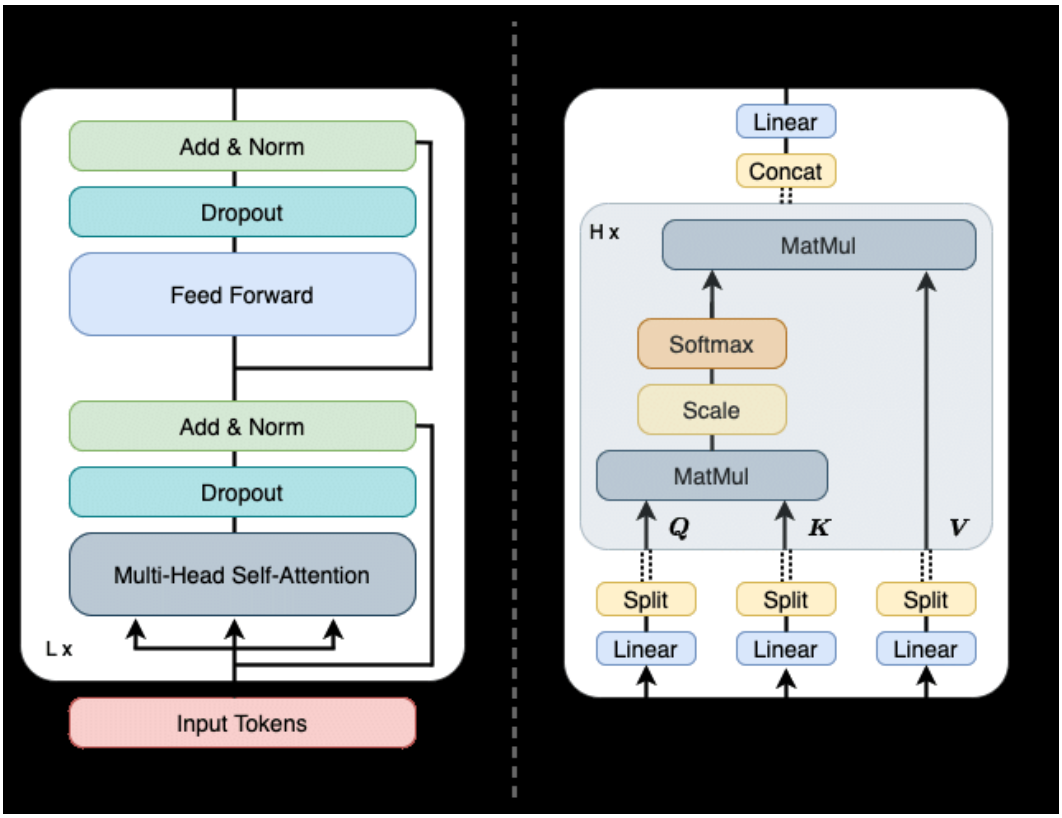
Slika 4.3. Transformator [23]

mopozornosti, mehanizam pozornosti prema enkoderu, sloj potpuno povezanih mreža te normalizacija i preskakanje.

Slika 4.4. predstavlja arhitekturu jednog enkodera. S lijeve strane je prikaz klasične arhitekture modela koja se sastoji od: preskakanja, mehanizma višeglave samopozornosti, odbacivanja (eng. dropout) te normalizacije.

Desna strana slike prikazuje arhitekturu višeglave samopozornosti u transformatorima. Proces počinje s ulaznim vektorima koji se dijele u tri različita vektora: Q, K i V.

Nakon toga, vektori se dijele na H različitih "glava", što omogućava modelu da istovremeno obrađuje različite dijelove informacija. Svaka glava samostalno izračunava pozornost koristeći matrično množenje, skaliranje, i *Softmax* funkciju.



Slika 4.4. Enkoder [24]

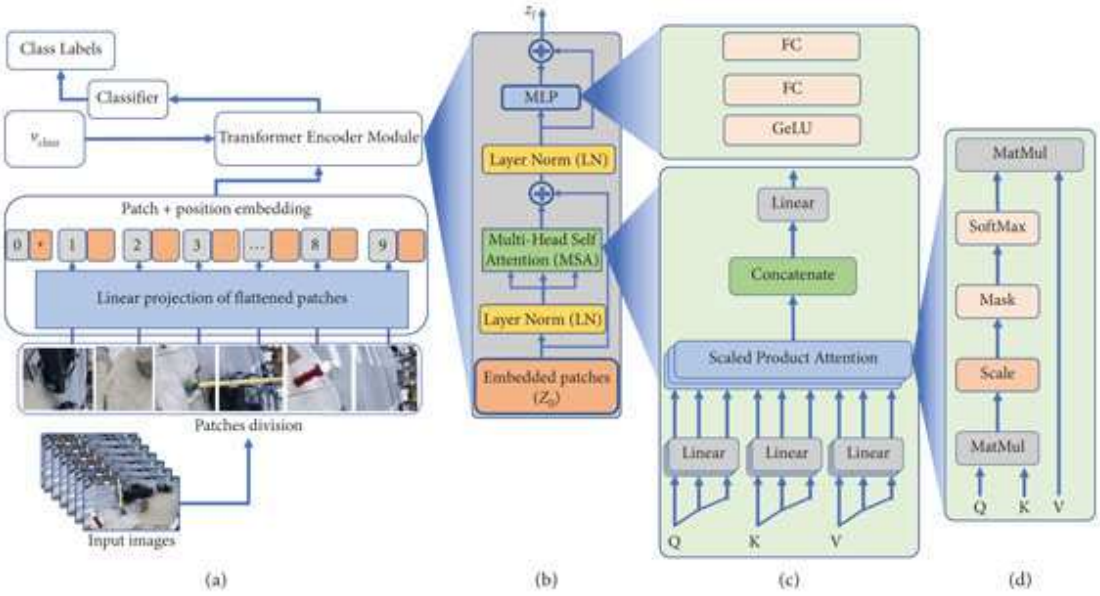
Rezultati svih glava se zatim spajaju i prolaze kroz potpuno povezani sloj. Ovaj mehanizam omogućava modelu da se istovremeno fokusira na različite dijelove sekvence, poboljšavajući sposobnost učenja složenih obrazaca u podacima.

Dekoderi imaju vrlo sličnu arhitekturu kao i enkoder uz jedan dodatak: poveznica s rezultatom enkodera kroz mehanizam pozornosti prema enkoderu.

4.4. Transformatori za obradu vizualnih podataka

Transformatori su postali ključni alat u području obrade prirodnog jezika, no njihova primjena se širi i na obradu vizualnih podataka. Vizualni podaci, poput slika i videa, predstavljaju bogat izvor informacija, no njihova kompleksnost zahtijeva sofisticirane tehnike obrade. Upravo u tome transformatori pokazuju svoju snagu.

Prva upotreba transformatora za obradu slika dogodila se 2021. godine u radu [19]. U tome radu pokazano je kako nije nužno uvijek se okrenuti konvolucijskim neuronskim mrežama kada se kao ulazni podatak koriste slike.



Slika 4.5. Transformator za obradu vizualnih podataka [25]

Slika 4.5. prikazuje primjer transformatora za obradu slika. Prva ključna stvar za uočiti je podjelu slika na dijelove (eng. patch). Svaki dio slike predstavlja jedan ulaz koji se predaje transformatoru.

Svaki dio slike prolazi kroz jednake korake kao i riječi kod klasičnih transformatora. Prvi korak je reprezentacija dijelova slike, zatim normalizacija, nakon toga višeglava samopozornost pa linearni sloj.

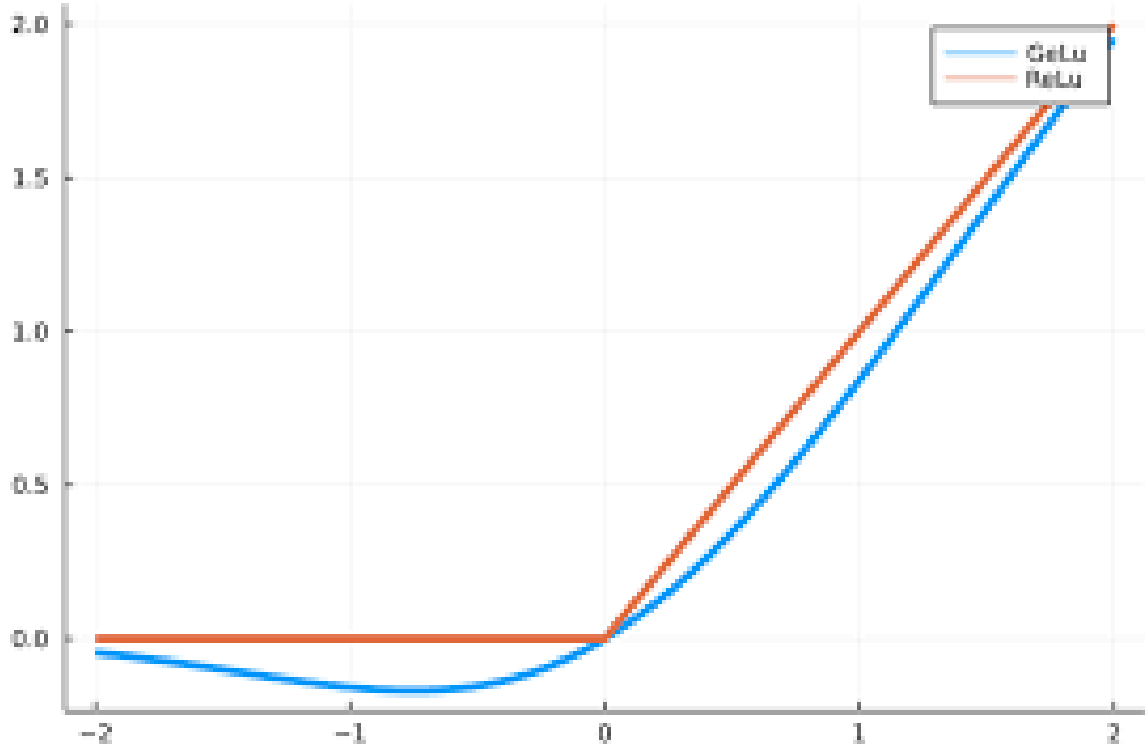
Na slici se vidi korištenje GeLu aktivacijske funkcije popraćena s dva potpuno povezana sloja u posljednjem linearnom dijelu nakon povezivanja elemenata. GeLu (Gaussian Error Linear Unit) je jedna od aktivacijskih funkcija koje se koriste u neuronskim mrežama. Ova funkcija ima prednosti u smislu glatkoće i diferencijabilnosti, što ju čini pogodnom za optimizaciju pomoću algoritama gradijentnog spusta.

GeLu funkcija je kombinacija ulaznog podatka s funkcijom distribucije Gaussa i funkcijom greške (eng. error function). Matematički, GELU funkciju možemo zapisati kao:

$$\mathrm{GELU}(x) = x \cdot \Phi(x) = 0.5x \cdot \left(1 + \mathrm{erf}\left(\frac{x}{\sqrt{2}}\right)\right) \tag{4.4}$$

Kao što je prikazano na slici 4.6. GeLu funkcija predstavlja derivabilnu verziju funk-

cije zglobnice (ili ReLu). Derivabilnost je definirana kao neprekidnost u svakoj točki funkcije, što u funkciji zglobnice nije ostvareno za vrijednost nula. Funkcija GeLu rješava ovaj problem te se zbog toga često koristi u situacijama kada je derivabilnost bitna.



Slika 4.6. Usporedba GeLu i ReLu funkcija [26]

Model transformatora za obradu vizualnih podataka sastoji se isključivo od slojeva enkodera, bez prisutnosti dekodera. Razlog tome je potreba samo za kodiranjem ulaza nelinearnim transformacijama bez stvaranja izlaza. Izlazi se ostvaruju potpuno povezanim slojevima koji imaju identičnu ulogu kao i potpuno povezani sloj konvolucijske neuronske mreže. Na slici 4.2. to bi predstavljao klasifikacijski dio koji pretežno čini, kako je već prethodno spomenuto, potpuno povezani sloj.

Jedna od glavnih primjena transformatora u vizualnoj obradi je detekcija objekata. Primjena transformatora omogućuje detekciju objekata u različitim okruženjima, uključujući složene scene s više objekata.

Osim detekcije objekata, često se koristi i klasifikacija objekata. Tada klasifikatori imaju dodatan zadatak kojim moraju, često uz detekciju, klasificirati objekt iz niza prethodno definiranih klasa.

Međutim, u slučaju ovoga diplomskoga rada transformator za obradu vizualnih podataka koristit će se u svrhu dobivanja koordinata pogleda vozača automobila, što je regresijski problem.

5. Model

Ovaj diplomski rad fokusira se na kombinaciju više modela. [20] Koristi se model hibridnog transformatora za obradu vizualnih podataka s ciljem procjene pogleda vozača. Ovaj model kombinira karakteristike konvolucijskih neuronskih mreža i transformatora kako bi postigao visoku preciznost u zadatku procjene pogleda. Model uključuje sljedeće dvije glavne komponente:

- Konvolucijski slojevi: koriste se za ekstrakciju značajki iz ulaznih slika
- Transformator blokovi: koriste se za modeliranje dugoročnih zavisnosti između značajki i za integraciju kontekstualnih informacija

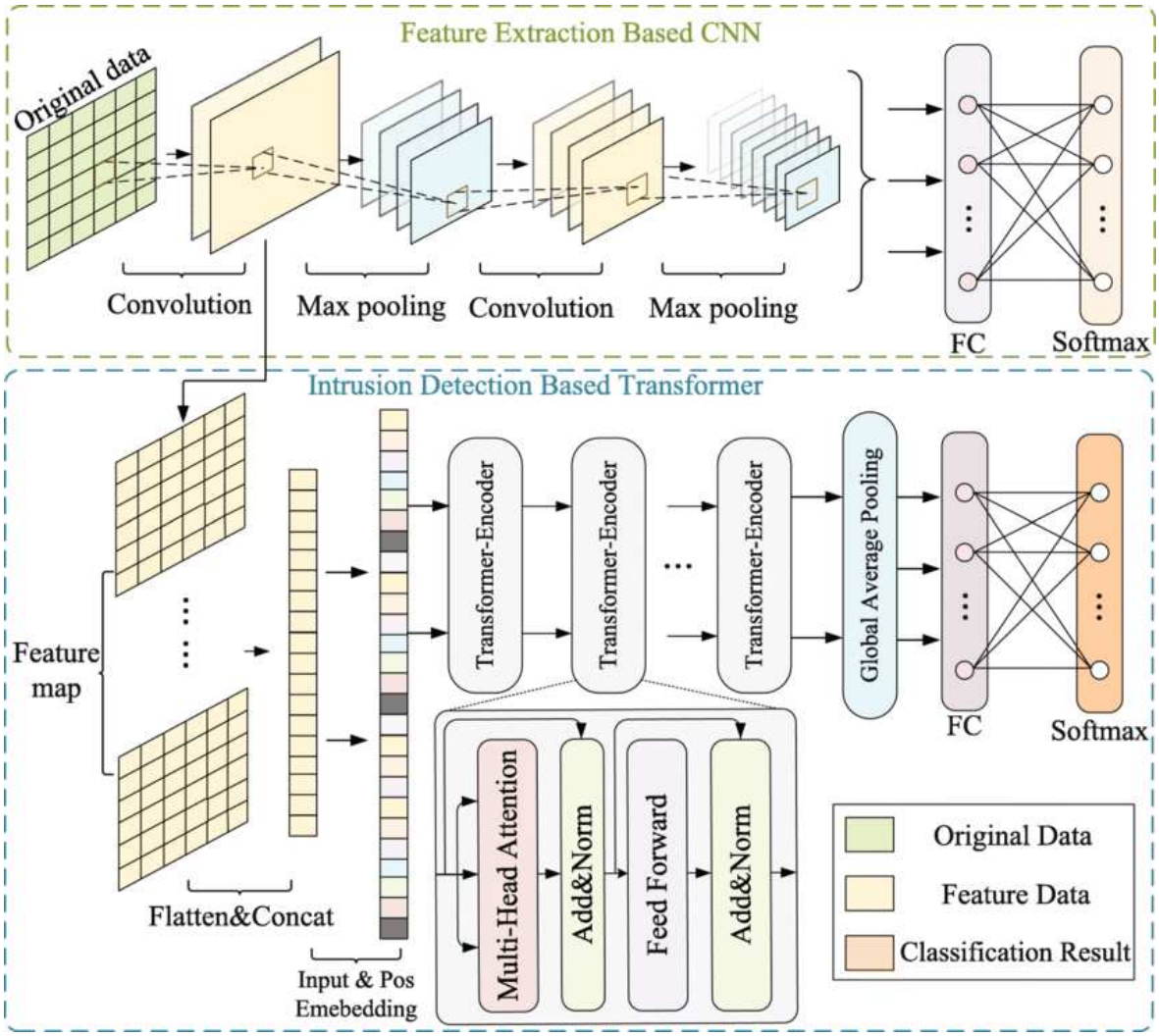
Na slici 5.1. vidimo primjer arhitekture modela korištenog u ovome diplomskome radu. Model počinje od konvolucijskog modela, ali završava s njime nakon konvolucije i ekstrakcije značajki. S tim rezultatima dolazi na transformator za obradu vizualnih podataka te ih koristi kao što inače koristi i slike.

5.1. Konvolucijski slojevi i ResNet-18

Konvolucijski slojevi prvi su korak u obradi vizualnih podataka ovoga modela. Njihova uloga je ekstrakcija lokalnih značajki iz ulaznih slika, kao što su rubovi, teksture i osnovni oblici. Ovi slojevi koriste konvolucijske filtre koji se kreću preko slike i identificiraju lokalne obrasce.

U ovome modelu, konvolucijska neuronska mreža koristi se za dobivanje niza značajki koje predstavljaju ulazne slike na višoj razini apstrakcije. Te značajke zatim služe kao ulaz za model transformatora za obradu vizualnih podataka.

Arhitektura korištena u ovome modelu je ona od modela pod imenom ResNet-18.

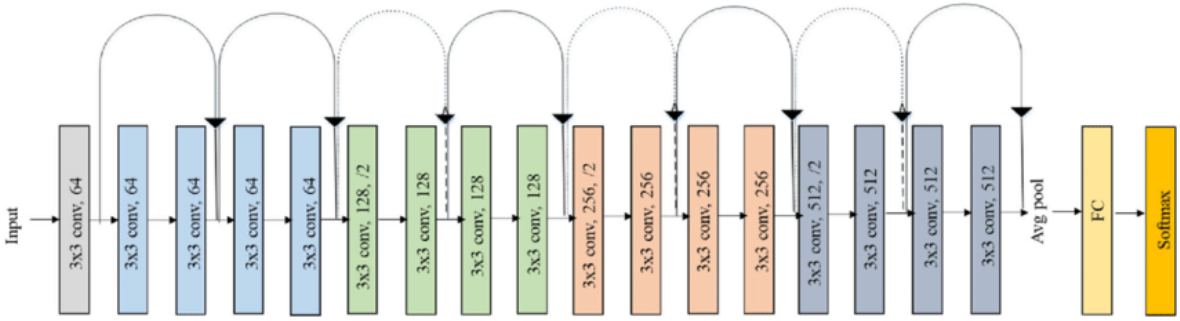


Slika 5.1. Arhitektura hibridnog transformatora [27]

ResNet-18 je dobro poznat duboki konvolucijski neuronski model koji koristi rezidualne blokove za efikasnu ekstrakciju značajki, omogućujući izgradnju vrlo dubokih mreža bez problema s nestajanjem gradijenata.

ResNet-18 sastoji se od 18 slojeva koji se mogu trenirati, organiziranih u 4 bloka rezidualnih slojeva. Rezidualni slojevi su na slici 5.2. označeni sa različitim bojama. Rezidualni blok ključni je element ResNet arhitekture. On uključuje izravnu vezu između ulaza i izlaza. Ovaj dizajn omogućuje mreži da uči identitetska mapiranja, što pomaže kod problema nestajućeg gradijenta.

Značajke koje generira ResNet-18 su u obliku 2D tenzora. Prije nego što ove značajke mogu biti proslijeđene transformatoru, potrebno je transformirati ih u odgovarajući oblik sekvencijalnih podataka. Ovaj korak uključuje "izravnavanje" (eng. flattening) 2D ten-



Slika 5.2. Arhitektura ResNet-18 modela [28]

zora u 1D vektore. Svaki vektor predstavlja jednu lokaciju u izvornoj slici, a svi vektori zajedno čine sekvencu koja se koristi kao ulaz za transformator.

Sljedeći korak je dodavanje pozicijskih informacija. S obzirom da transformatori ne mogu inherentno razumjeti redoslijed ulaza, pozicijske informacije se dodaju svakom vektoru.

Posljednji korak je prosljeđivanje sekvence vektora kojoj je pridodana pozicijska informacija na ulaz u transformator za daljnju obradu.

5.2. Transformator blokovi

Transformator blokovi ključni su dio ovog hibridnog modela. Oni se koriste za obradu sekvencijskih podataka i modeliranje zavisnosti između udaljenih elemenata u sekvenci.

Transformator se sastoji od dva glavna dijela:

- Mehanizma samopozornosti: omogućava modelu obraćanje pažnje na različite sekvence kako bi razumio kontekstualne zavisnosti.
- Potpuno povezani sloj neuronske mreže: sastoji se od dva potpuno povezana sloja neuronske mreže s aktivacijskom funkcijom između njih (slika 4.5.).

Mehanizam samopozornosti izračunava važnost svakog elementa u sekvenci u odnosu na ostale elemente. To se postiže s pomoću tri matrice: Q (eng. Query), K (eng. Key) i V (eng. Value).

Rezultat mehanizma samopozornosti dobije se tako što se izračuna težinska suma elemenata sekvence, gdje su težine određene pažnjom između elemenata. Formula, po

uzoru na sliku 4.4., izgleda ovako:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{5.1}$$

Potpuno povezani sloj primjenjuje se nezavisno i identično na svakoj poziciji u sekvenci. Sastoji se od dva linearna sloja s aktivacijskom funkcijom zglobnice između njih.

5.3. Parametri modela

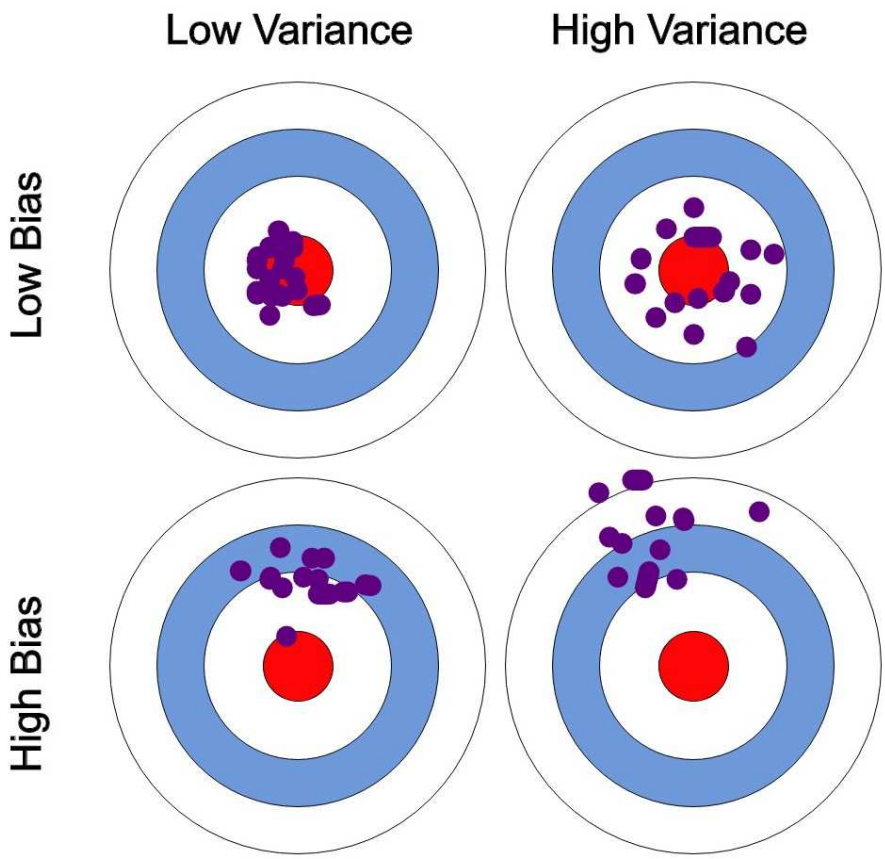
Parametri modela dubokog učenja su unutarnje varijable koje model uči tijekom procesa treniranja. To su vrijednosti koje se optimiziraju kako bi model mogao izvršavati zadatak na najbolji mogući način, poput klasifikacije slika, prepoznavanja govora, ili predviđanja vremenskih serija. Najvažniji parametri uključuju težine (weights) i pristranosti (biases) koji definiraju kako su različiti neuroni povezani i kako se informacije obrađuju unutar mreže.

Težine su varijable koje određuju snagu veze između neurona u različitim slojevima mreže. Pristranosti su dodatni parametri koji omogućuju modelu da bolje prilagodi funkciju aktivacije i osigura veću fleksibilnost u učenju složenih obrazaca.

U konvolucijskim neuronskim mrežama, kao što je ResNet-18, težine su prisutne u konvolucijskim filtrima koji se primjenjuju na ulazne slike kako bi se izdvojile značajke poput rubova i tekstura.

Rezultati treniranja daju razli;it omjer varijance i pristranosti. Pristranost označava pogrešku zbog pretpostavki modela koje pojednostavljuju stvarnost, dok varijanca mjeri koliko modelove predikcije variraju s različitim treniranim skupovima podataka. Visoka pristranost vodi do prejednostavnih modela, gdje model ne uspijeva uhvatiti složenost podataka. Visoka varijanca vodi do prenaučenosti, gdje model previše dobro prati šum u podacima. Cilj je postići balans, minimizirajući obje ove komponente za optimalnu izvedbu.

Slika 5.3. prikazuje odnos pristranosti i varijance. Mala pristranost se očitava kao rezultat blizu centra, dok je velika pristranost odstupanje od centra. Mala varijanca je



Slika 5.3. Odnos pristranosti i varijance [29]

prikazana kao više međusobno bliskih rezultata, dok je rezultat visoke varijance raspršenost rezultata modela.

Proces treniranja modela uključuje iterativno prilagođavanje težina i pristranosti korištenjem algoritama optimizacije kao što su Stohastički gradijentni spust ili Adam. Ovi algoritmi koriste informacije iz funkcije gubitka (eng. loss function), koja mjeri koliko dobro model predviđa željene izlaze, kako bi ažurirali parametre u smjeru koji smanjuje gubitak.

Parametri ResNet-18 modela su:

- Težine konvolucijskih slojeva: Matrice koje se koriste za ekstrakciju značajki iz ulaznih slika. Svaki konvolucijski sloj ima svoje težine koje se optimiziraju tijekom treniranja.

- Pristranosti konvolucijskih slojeva: Dodane vrijednosti koje pomažu u pomicanju funkcije aktivacije.

Parametri transformatora za obradu vizualnih podataka su:

- Težine mehanizma samopozornosti: Parametri koji određuju kako se različiti dijelovi ulaza međusobno pozivaju i kombiniraju. To uključuje težine za Q (eng. Query), K (eng. Key) i V (eng. Value).
- Pristranosti mehanizma samopozornosti: Dodane vrijednosti koje pomažu u pomicanju funkcije aktivacije.
- Težine potpuno povezanih slojeva: Težine koje povezuju sve neurone iz jednog sloja s neuronima u sljedećem sloju.
- Pristranosti potpuno povezanih slojeva: Pristranosti za neurone u potpuno povezanim slojevima.
- Težine pozicijskih kodiranja: Parametri koji se koriste za dodavanje informacija o poziciji ulaznih zakrpa.

6. Hiperparametri

Hiperparametri su ključni elementi u procesu treniranja modela dubokog učenja koji se postavljaju prije početka treniranja i ostaju konstantni tijekom cijelog procesa. Njihova pravilna konfiguracija može bitno poboljšati performanse modela. Također je moguć i utjecaj na sposobnost generalizacije modela na nove, prethodno neviđene podatke. Hiperparametri kontroliraju proces učenja i arhitekturu modela, za razliku od parametara modela koji se uče tijekom treniranja i u početku se postavljaju na proizvoljne (ili češće nasumične) vrijednosti.

Ključne uloge hiperparametara u dubokom učenju su:

- Kontrola procesa učenja: Hiperparametri određuju kako model uči iz podataka. Oni definiraju brzinu i način na koji se težine modela ažuriraju tijekom treniranja. Na primjer, oni mogu odrediti koliko snažno će model reagirati na pogreške tijekom svakog koraka treniranja ili koliko puta će cijeli skup podataka biti procesiran tijekom treniranja.

- Arhitektura modela: Hiperparametri također određuju strukturu modela. Oni specificiraju broj slojeva u mreži, broj neurona u svakom sloju, veličinu filtera u konvolucijskim slojevima, i druge arhitekturalne karakteristike. Ove postavke izravno utječu na kapacitet modela da uči složene obrasce i odnose u podacima.

- Regularizacija i generalizacija: Hiperparametri pomažu u regulaciji složenosti modela kako bi se spriječila prenaučenost (eng. overfitting) i poboljšala sposobnost modela da generalizira na nove podatke. Regularizacijske tehnike kao što su dropout, L1 i L2 regularizacija su kontrolirane putem hiperparametara. Ovi mehanizmi dodaju dodatne restrikcije na model kako bi se osiguralo da ne nauči samo šum u treniranim podacima.

- Optimizacija performansi: Pravilno postavljanje hiperparametara može značajno poboljšati učinkovitost treniranja, smanjujući vrijeme potrebno za postizanje zadovoljavajućih performansi i poboljšavajući rezultate. Kroz tehnike poput grid search, random search i bayesovske optimizacije, istražuju se različite kombinacije hiperparametara kako bi se pronašli najbolji skup za dati problem.

- Utjecaj na resurse: Hiperparametri utječu na korištenje računalnih resursa, uključujući memoriju i procesorsko vrijeme. Na primjer, veličina grupe (eng. batch) direktno utječe na količinu memorije potrebne za treniranje, dok kompleksnost modela (broj slojeva i neurona) utječe na ukupno vrijeme treniranja.

- Fleksibilnost modela: Hiperparametri omogućuju prilagodbu modela specifičnim zadacima i skupovima podataka. Fleksibilnost u postavljanju hiperparametara znači da se modeli mogu optimizirati za širok raspon problema, od jednostavnih klasifikacija do složenih zadataka kao što su prepoznavanje objekata ili razumijevanje prirodnog jezika.

Najbitniji hiperparametri modela dubokog učenja su:

- Brzina učenja (eng. learning rate): Određuje korak ažuriranja tijekom treniranja. Prevelika brzina učenja teško će dovesti do konvergencije, dok premala brzina učenja dovodi do prespore konvergencije modela. (slika 6.1.)

- Veličina grupe (eng. batch size): Broj uzoraka koji se obrađuju prije ažuriranja težina. Premala veličina grupe dozvoljava modelu da trenira na individualnim podacima, ali vrijeme trajanja treniranja će biti duže. S druge strane, prevelika veličina grupe brže će završiti proces treniranja modela, ali model potencijalno neće uhvatiti detalje u podacima koji su potrebni za veću točnost.

- Broj epoha: Broj puta kada cijeli skup podataka prolazi kroz mrežu. Mali broj epoha možda neće biti dovoljan za adekvatno treniranje modela, a preveliki broj epoha može dovesti do pretreniranosti modela.

- Optimizator: Algoritam koji optimizira težine modela tijekom treniranja. Danas je najpopularniji optimizator Adam, ali se također koriste: stohastički gradijentni spust (SGD), RMSProp, srednja kvadratna pogreška (MSE) itd.

- Regularizacija: Tehnike koje sprječavaju prenaučenost modela, poput L2 regularizacije. Tipično, regularizacija mijenja marginalno smanjenje točnosti treniranja za povećanje mogućnosti generalizacije. [30]

- Stopa odustajanja (eng. dropout rate): Postotak neurona koji se isključuju kako bi se spriječila prenuačenost modela.

Dodatno, modeli transformatora za vizualnu obradu podataka i ResNet-18 modeli imaju svoje vlastite hiperparametre. Za ResNet-18 modele to su:

- Veličina jezgre (eng. kernel size): Dimenzije filtara korištenih u konvolucijskim slojevima.

- Korak (eng. stride): Razmak između primjene filtara na ulazne podatke.

- Broj slojeva: Ukupan broj slojeva u mreži, što utječe na dubinu modela i njegovu ekspresivnu snagu.

- Veličina ulazne slike

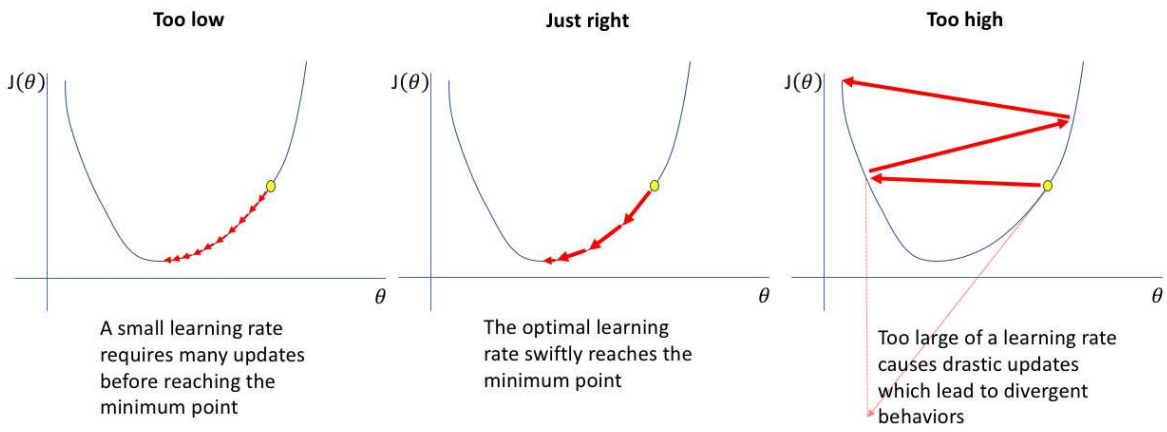
Hiperparametri modela transformatora su:

- Veličina zakrpe (eng. patch size): Dimenzije područja slike koja se obrađuju (veličina jedne sekvence).

- Broj transformator blokova: broj blokova (enkodera) koji se koristi u arhitekturi transformatora.

- Broj glava u mehanizmu samopozornosti: Broj paralelnih samopozornih mehanizama u svakom transformator bloku.

- Dimenzija modela: Ukupna dimenzija skrivenih slojeva i vektora tokena korištenih u transformatoru. Utječe na kapacitet modela i računalne resurse potrebne za treniranje.



Slika 6.1. Važnost brzine učenja [31]

7. Rezultati

Hibridni transformator za obradu vizualnih podataka podvrgnut je testiranju nad velikom količinom podataka. U skupu za treniranje nalazilo se 97608 slika s njima pripadnim oznakama, dok se u skupu za testiranje nalazilo 23737 podataka. To je omjer približno jednak 80:20.

Presjek skupa za treniranje i skupa za testiranje je prazan skup. Osim toga, osobe na kojima je model treniran nisu prisutne u skupu za testiranje. Sveukupno je sudjelovalo 28 osoba u stvaranju ovoga skupa podataka. Od tih 28 osoba, podaci od čak njih 23 ulazi u skup za treniranje, dok nad preostalih 5 osoba se vršilo testiranje. Zbog ravnopravnsti testiranja skupovi za treniranje i testiranje nisu bili mijenjani tijekom procesa testiranja modela.

Prilikom testiranja mijenjali su se razni hiperparametri transformatora, dok su hiperparametri ResNet-18 modela ostali jednakima kroz cijelo testiranje. Parametri koji su bili mijenjani i testirani su: veličina grupe, funkcija gubitka i broj glava unutar višeglave pozornosti (eng. multi-head attention)

Svako testiranje provedeno je tri puta te su kao rezultat uzeti prosječni rezultati testiranja. Model je također spremao rezultate svakih 10 epoha te je sukladno tome provodio testiranje nad svim spremljenim kontrolnim točkama.

Model tijekom treniranja koristi skretanje (eng. yaw) i nagib (eng. pitch) kao rezultate koje treba predvidjeti. Skretanje predstavlja horizontalnu rotaciju, dok kut nagiba odnosi se na vertikalnu rotaciju.

U skupu podataka dan je trodimenzionalni normalizirani vektor pogleda. Normalizirani vektor definiran je kao vektor čija je duljina jednaka jedan.

Formule za pretvaranje trodimenzionalnog vektora pogleda u skretanje i nagib su sljedeće:

$$yaw = \arctan\left(gaze3d[0], -gaze3d[2]\right) \tag{7.1}$$

$$pitch = \arcsin\left(gaze3d[1]\right) \tag{7.2}$$

Kao metrika pogreške odabrana je prosječna kutna pogreška (eng. angular error). Ta pogreška se računa između dva trodimenzionalna vektora pomoću sljedeće formule:

$$\text{angular} = \frac{1}{N}\sum\left[\arccos\left(\min\left(\frac{\text{total}}{||\text{gaze}|| \cdot ||\text{label}||}, 0.9999999\right)\right) \times \frac{180}{\pi}\right] \tag{7.3}$$

Gaze označava dobiveni vektor pogleda koji se nakon pretvorio iz kuta skretanja i nagiba u trodimenzionalni vektor, dok je *label* točna vrijednost trodimenzionalnog vektora.

Formule s pomoću kojih se dobiva trodimenzionalni vektor iz kuta skretanja i nagiba su:

$$gaze3d[0] = \cos\left(yaw\right) \times \sin\left(pitch\right) \tag{7.4}$$

$$gaze3d[0] = -\sin\left(yaw\right) \tag{7.5}$$

$$gaze3d[0] = \cos\left(yaw\right) \times \cos\left(pitch\right) \tag{7.6}$$

Tablica 7.1. prikazuje rezultate dobivene na model trenirane kroz 80 epoha s različitom veličinom grupe. Ono što je iz rezultata vidljivo je to da su najbolji rezultati vrlo često dobiveni u ranim fazama treniranja.

Tablica 7.2. prikazuje uprosječene rezultate tablice 7.1. Iz prosječnih rezultata vidimo da je optimalna veličina grupe jednaka 64, te se najbolji rezultat postiže nakon 10 epoha treniranja.

Sljedeća tablica 7.3. pokazuje vrijednosti nakon izmjene funkcije gubitka. U tablicama 7.1. i 7.2. dobiveni su rezultati korištenjem L1 funkcije gubitka, dok je u tablici 7.3.

Epoch	TEST 1 batch size			TEST 2 batch size			TEST 3 batch size		
	256	**128**	**64**	**256**	**128**	**64**	**256**	**128**	**64**
10	**6.69**	**6.31**	**5.66**	**5.84**	7.05	**5.76**	**5.92**	6.42	**5.79**
20	**6.69**	6.34	6.32	6.47	6.48	6.10	6.51	**6.32**	6.32
30	6.69	6.96	6.56	6.59	**6.45**	6.51	6.75	6.79	6.34
40	6.77	6.92	6.79	6.71	6.64	6.68	6.99	6.86	6.89
50	6.93	6.94	6.86	6.78	6.74	6.81	6.92	6.87	6.95
60	6.94	7.04	6.92	6.79	6.77	6.84	7.14	6.94	7.00
70	6.95	7.03	6.93	6.77	6.79	6.85	7.09	6.95	6.97
80	6.98	7.05	6.95	6.81	6.78	6.88	7.11	6.96	7.03

Tablica 7.1. Rezultati testiranja modela s različitom veličinom grupe

Epoch	Batch size 256	Batch size 128	Batch size 64
10	**6.15**	6.60	**5.74**
20	6.56	**6.38**	6.25
30	6.68	6.73	6.47
40	6.82	6.81	6.79
50	6.88	6.85	6.87
60	6.99	6.92	6.92
70	6.94	6.922	6.92
80	6.97	6.93	6.95

Tablica 7.2. Prosječne vrijednosti za rezultate testiranja

korištena funkcija gubitka srednje kvadratne pogreške. Iz prosječnih rezultata vidimo povećanje pogreške nakon promjene funkcije gubitka.

L1 funkcija gubitka, poznata pod drugim nazivom kao i apsolutna funkcija gubitka, mjeri razliku između predviđenih i stvarnih vrijednosti koristeći apsolutne vrijednosti tih razlika. Formula za L1 funkciju gubitka glasi:

$$L1 = \sum_{i=1}^{n} |y_i - \hat{y}_i| \tag{7.7}$$

S druge strane, funkcija srednje kvadratne pogreške (skraćeno MSE) mjeri prosječnu kvadratnu razliku između predviđenih i stvarnih vrijednosti. Formula za MSE glasi ovako:

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{7.8}$$

Epoch	TEST 1	TEST 2	TEST 3	AVG
10	6.05	6.20	6.77	**6.34**
20	6.89	7.19	6.78	6.95
30	7.71	7.42	7.33	7.49
40	7.59	7.17	7.21	7.32
50	7.61	7.15	7.24	7.33

Tablica 7.3. Rezultati testiranja sa srednjom kvadratnom pogreškom

Tablica 7.4. prikazuje rezultate s različitim brojem glava u višeglavoj pozornosti. I dalje vidimo prisutnost veće točnosti na manjem broju epoha. Model u tablici 7.1. treniran je na N jednako 8 glava.

Epoch	TEST 1 N-heads			TEST 2 N-heads			TEST 3 N-heads		
	1	**4**	**16**	**1**	**4**	**16**	**1**	**4**	**16**
10	6.33	**5.87**	**6.02**	**6.15**	**5.82**	6.88	**5.91**	**5.90**	**6.33**
20	**6.14**	6.31	6.68	6.35	6.38	**6.24**	6.07	6.29	6.50
30	6.55	6.89	6.67	6.78	6.64	6.59	6.62	6.62	6.55
40	6.76	6.87	6.96	6.83	6.90	6.77	6.78	6.84	6.86
50	6.82	7.01	7.02	6.91	7.00	6.89	7.00	6.90	6.99

Tablica 7.4. Rezultati testiranja modela s različitom brojem glava

Tablica 7.5. prikazuje uprosječene rezultate modela s različitim brojem glava. Ako se usporede najbolji rezultati ove tablice s najboljim rezultatom tablice 7.2. može se uočiti da i dalje rezultat tablice 7.2. je najbolji u ovome testiranju. Njegov rezultat je 5.74, dok je rezultat tablice 7.5. jednak 5.86 što je drugi najbolji rezultat.

Epoch	Average N=1	Average N=4	Average N=16
10	**6.13**	**5.86**	**6.41**
20	6.19	6.33	6.47
30	6.65	6.72	6.60
40	6.79	6.87	6.86
50	6.91	6.97	6.97

Tablica 7.5. Uprosječeni rezultati modela s različitim brojem glava

Na slici 7.1. vidimo prikaz rezultata modela članka [15] odakle dolazi skup podataka korišten u ovome diplomskome radu. Njihov najbolji rezultat je 6.2, dok je najbolji rezultat ovoga rada 5.74. S obzirom da je ovo jedini rad koji koristi ovaj skup podataka, teško je usporediti rezultate drugih radova s ovim. Međutim, iz ovoga vidimo da je pristup hibridnog transformatora bolji od njihovog pristupa kombinacije detekcije pogleda i pozornosti.

Method	{5/75/20}		{20/60/20}		{40/40/20}		{60/20/20}	
	Self	Test	Self	Test	Self	Test	Self	Test
Gaze360 [18]	18.7	20.3	21.4	20.3	23.0	20.3	17.2	20.3
ETH XGaze [49]	11.6	15.6	11.9	15.6	12.6	15.6	15.4	15.6
Mean	9.5	9.2	9.5	9.2	9.0	9.2	8.7	9.2
Supervised-only	9.7	**7.8**	**6.9**	6.8	8.1	7.4	7.0	**6.7**
Ours	**8.2**	**7.8**	**6.9**	**6.5**	**7.4**	**7.2**	**6.2**	**6.7**

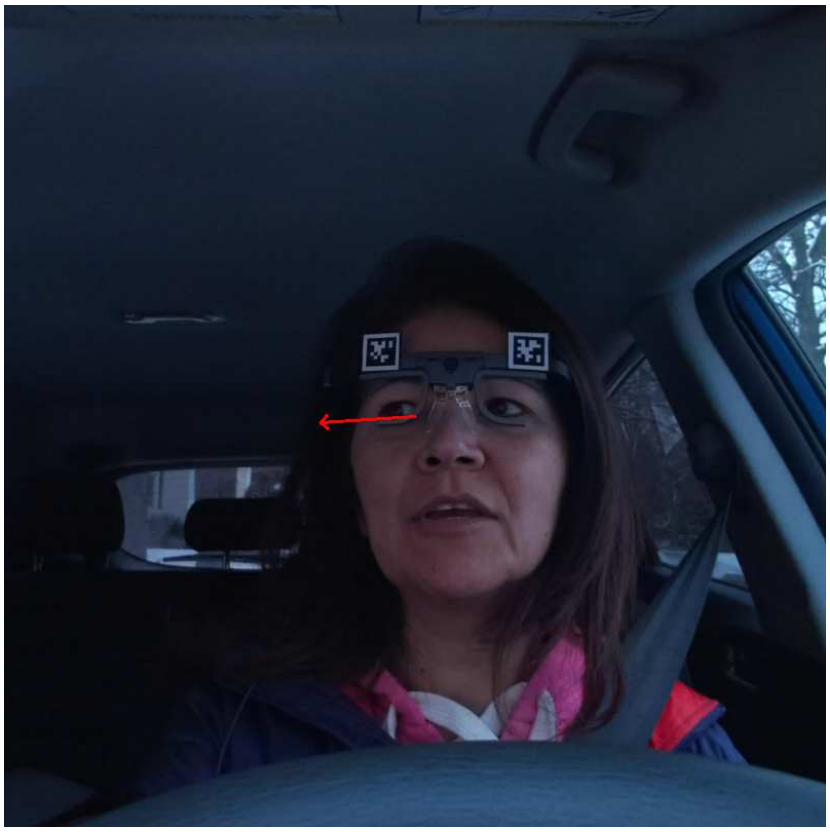
Slika 7.1. Rezultati članka [15]

Osim tabličnih rezultata, bitno je predočiti i rezultate u slikama. Sljedeći skup slika 7.2. - 7.4. prikazuje slike jedne osobe u vožnji te rezultate modela detekcije pogleda.

Slika 7.2. Rezultat Slika 1

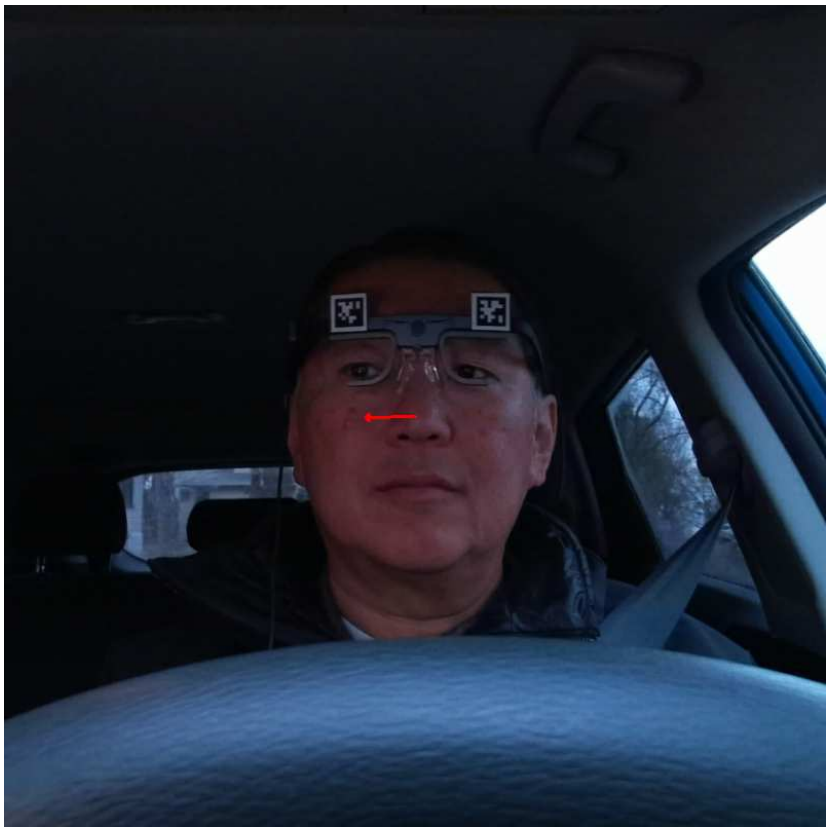
Sljedeći skup slika 7.5. - 7.7. prikazuje drugu osobu u vožnji te rezultat modela za detekciju pogleda.

Slika 7.3. Rezultat Slika 2

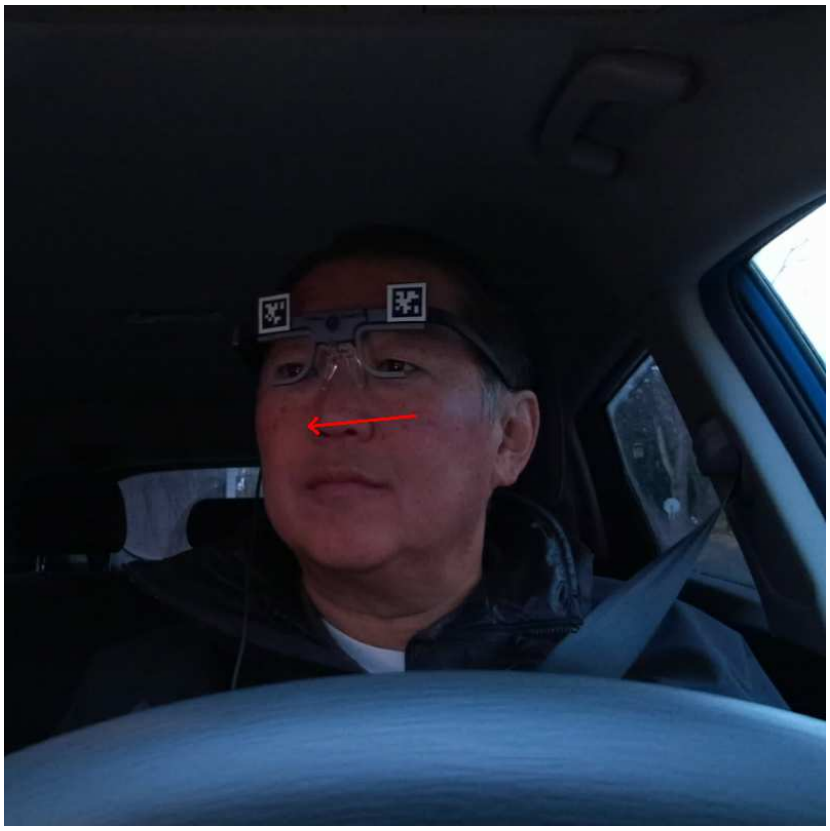


Slika 7.4. Rezultat Slika 3

Slika 7.5. Rezultat 2 Slika 1



Slika 7.6. Rezultat 2 Slika 2



Slika 7.7. Rezultat 2 Slika 3

8. Zaključak

Cilj ovog diplomskog rada bio je istražiti i razviti metodu za mjerenje smjera pogleda vozača pomoću analize slike, što je postignuto korištenjem naprednih metoda dubokog učenja, posebice modela transformatora. Kao što je vidljivo u odjeljku s rezultatima, predloženi hibridni model koji kombinira konvolucijske slojeve s transformatorskim blokovima učinkovit je u prepoznavanju smjera pogleda vozača. Ovo istraživanje je demonstriralo značajan potencijal primjene naprednih metoda dubokog učenja za poboljšanje sigurnosti u prometu kroz preciznu detekciju smjera pogleda vozača.

Rezultati eksperimentalnog dijela projekta pokazuju da je predloženi model bolji od prethodnih pristupa. Eksperimenti su provedeni na skupu podataka koji je predstavljao različite scenarije vožnje, uključujući različite položaje glave i očiju vozača. Na temelju testiranja najveći uspjeh imao je hibridni model s prosječnom kutnom pogreškom od 5,74, što je znatno više od dosadašnjeg najvećeg uspjeha od 6,2. Ova poboljšanja pripisuju se superiornom razumijevanju složenih uzoraka u slikovnim podacima modela transformatora u usporedbi s tradicionalnim konvolucijskim mrežama.

Implementacija ovog sustava u stvarnim uvjetima mogla bi uvelike povećati sigurnost u prometu. Sustavi pomoći u vožnji koji koriste ovu tehnologiju mogu rano označiti vozača ako ne obrati pozornost na cestu, čime se smanjuje vjerojatnost nesreća uzrokovanih nepažnjom. Mogućnosti uključuju integraciju u moderne automobile kao dio naprednih sustava pomoći vozaču, čime bi se povećala cjelokupna sigurnost putnika.

Daljnji razvoj ove tehnologije može uključivati dodatna poboljšanja u mogućnostima modela za prilagodbu u različitim uvjetima osvjetljenja i vožnje, kao i optimizaciju za bržu i učinkovitiju obradu u stvarnom vremenu. Dodatno, dodatna istraživanja bi se mogla posvetiti integraciji s drugim senzorima u vozilima kako bi se steklo potpuno ra-

zumijevanje stanja vozača i njegove okoline.

Ukratko, ovaj rad pokazuje da se procjena smjera pogleda vozača može značajno poboljšati korištenjem naprednih tehnika dubokog učenja. Rezultati ove studije daju čvrstu osnovu za daljnji razvoj automobilskih sigurnosnih sustava, čime se pridonosi općim poboljšanjima u cestovnoj sigurnosti.

Literatura

[1] On Behalf of Farmer, Cline and Campbell, PLLC, “LACK OF ATTENTION THE CAUSE OF MANY CAR ACCIDENTS”, https://www.farmerclinecampbell.com/blog/2018/04/lack-of-attention-the-cause-of-many-car-accidents/, 2018.

[2] Yihua Cheng, Haofei Wang, Yiwei Bao, Feng Lu, “Appearance-based Gaze Estimation with Deep Learning: A Review and Benchmark”, *JOURNAL OF LATEX CLASS FILES*, sv. VOL. 14, br. NO. 8, 2015.

[3] Laurence R. Young, David Sheena, “Survey of eye movement recording methods”, Behavior Research Methods and Instrumentation, 7(5), 397-429, 1975.

[4] C. Morimoto and M. Mimica, “Eye gaze tracking techniques for interactive applications”, Computer Vision and Image Understanding, vol. 98, no. 1, pp. 4-24, 2005.

[5] D. M. Stampe, “Heuristic filtering and reliable calibration methods for video-based pupil-tracking systems”, Behavior Research Methods, Instruments and Computers, vol. 25, no. 2, pp. 137-142, 1993.

[6] J. Qiang and X. Yang, “Real-time eye, gaze, and face pose tracking for monitoring driver vigilance”, Real-Time Imaging, vol. 8, no. 5, pp. 357-377, 2002.

[7] E. D. Guestrin and M. Eizenman, “General theory of remote gaze estimation using the pupil center and corneal reflections”, IEEE Transactions on Biomedical Engineering, vol. 53, no. 6, pp. 1124-1133, 2006.

[8] Z. Zhu and Q. Ji, “Novel eye gaze tracking techniques under natu- ral head movement”, IEEE Transactions on Biomedical Engineering, vol. 54, no. 12, pp. 2246-2260, 2007.

[9] R. Valenti, N. Sebe, and T. Gevers, “Combining head pose and eye location information for gaze estimation”, IEEE Transactions on Image Processing (TIP), vol. 21, no. 2, pp. p.802-815, 2012.

[10] X. Zhang, Y. Sugano, M. Fritz, and A. Bulling, “Appearance-based gaze estimation in the wild”, in The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.

[11] F. Lu, Y. Sugano, T. Okabe, and Y. Sato, “Adaptive linear regression for appearance-based gaze estimation”, IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), vol. 36, no. 10, pp. 2033-2046, 2014.

[12] Petr Kellnhofer, Adria Recasens, Simon Stent, Wojciech Matusik, Antonio Torralba, “Gaze360: Physically Unconstrained Gaze Estimation in the Wild”, International Conference in Computer Vision, 2019.

[13] Ashesh, Chu-Song Chen, Hsuan-Tien Lin, “360-Degree Gaze Estimation in the Wild Using Multiple Zoom Scales”, https://arxiv.org/abs/2009.06924, 2020.

[14] Ahmed A.Abdelrahman, Thorsten Hempel, Aly Khalifa, Ayoub Al-Hamadi, “L2CS-Net: Fine-Grained Gaze Estimation in Unconstrained Environments”, https://arxiv.org/abs/2203.03339, 2022.

[15] Isaac Kasahara, Simon Stent, and Hyun Soo Park, “Look Both Ways: Self-Supervising Driver Gaze Estimation and Road Scene Saliency”, https://www.ecva.net/papers/eccv_2022/papers_ECCV/papers/136730128.pdf, 2022.

[16] Rizwan Ali Naqvi, Muhammad Arsalan, Ganbayar Batchuluun, Hyo Sik Yoon, Kang Ryoung Park, “Deep Learning-Based Gaze Detection System for Automobile Drivers Using a NIR Camera Sensor”, https://www.mdpi.com/1424-8220/18/2/456.

[17] HYO SIK YOON, NA RAE BAEK, NOI QUANG TRUONG, AND KANG RYOUNG PARK, “Driver Gaze Detection Based on Deep Residual Networks Using the Combined Single Image of Dual Near-Infrared Cameras”, IEE Access, Open Access Journal vol 7, 2019.

[18] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, “Attention Is All You Need”, 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA., 2017.

[19] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, Neil Houlsby, “AN IMAGE IS WORTH 16X16 WORDS: TRANSFORMERS FOR IMAGE RECOGNITION AT SCALE”, Published as a conference paper at ICLR 2021, 2021.

[20] Yihua Cheng, Feng Lu, “Gaze Estimation using Transformer”, https://arxiv.org/pdf/2105.14424, 2021.

[21] wikipedia, “Neuronska mreza”, https://hr.wikipedia.org/wiki/Neuronska_mreÅ¿a#/media/Datoteka:Multi-Layer_Neural_Network-Vector.svg, 2021.

[22] Kh. Nafizul Haque, “What is Convolutional Neural Network — CNN (Deep Learning)”, https://www.linkedin.com/pulse/what-convolutional-neural-network-cnn-deep-learning-nafiz-shahriar/, 2023.

[23] Stefania Cristina, “The Transformer Model”, https://machinelearningmastery.com/the-transformer-model/, 2023.

[24] Vittorio Mazzia, “Action Transformer: A Self-Attention Model for Short-Time Human Action Recognition”, https://www.researchgate.net/figure/Transformer-encoder-layer-architecture-left-and-schematic-overview-of-a-multi-head_fig1_352992757, 2021.

[25] Altaf Hussain, Tanveer Hussain, Waseem Ullah, Sung Wook Baik, “Vision Transformer and Deep Sequence Learning for Human Activity Recognition in Surveillance Videos”, https://www.researchgate.net/figure/The-Vision-Transformer-architecture-a-the-main-architecture-of-the-model-b-the_fig2_359740029, 2022.

[26] Miguel Urena Pliego, Ruben Martinez, Beatriz González-Rodrigo, Miguel Marchamalo, “Automatic Building Height Estimation: Machine Learning Models for Urban Image Analysis”, https://www.researchgate.net/figure/Comparison-

of-the-ReLu-and-GeLu-activation-functions-ReLu-is-simpler-to-compute-but_fig3_370116538, 2023.

[27] Ruizhe Yao, Ning Wang, Peng Chen, "A CNN-transformer hybrid approach for an intrusion detection system in advanced metering infrastructure", https://www.researchgate.net/figure/The-architecture-of-CNN-Transformer-hybrid-network_fig1_365934687, 2022.

[28] Farheen Ramzan, Muhammad Usman Ghani Khan, Asim Rehmat, Zahid Mehmood, "A Deep Learning Approach for Automated Diagnosis and Multi-Class Classification of Alzheimer's Disease Stages Using Resting-State fMRI and Residual Neural Networks", https://www.researchgate.net/figure/Original-ResNet-18-Architecture_fig1_336642248, 2019.

[29] NVS Yashwanth, "5. Bias-Variance", https://nvsyashwanth.github.io/machinelearningmaster/bias-variance/, 2020.

[30] Jacob Murel Ph.D., Eda Kavlakoglu, "What is regularization?" https://www.ibm.com/topics/regularization, 2023.

[31] Jeremy Jordan, "Setting the learning rate of your neural network." https://www.jeremyjordan.me/nn-learning-rate/, 2018.

Sažetak

Procjena smjera pogleda vozača temeljena na analizi slike

Dario Dugonjevac

Ovaj rad istražuje metodu procjene smjera pogleda vozača temeljenu na analizi slike, koristeći duboko učenje i modele transformatora. Cilj je smanjiti broj prometnih nesreća uzrokovanih rastresenošću vozača, razvijajući sustav koji detektira smjer pogleda i upozorava vozača kada ne gleda cestu. Testiranje na dostupnim skupovima podataka pokazalo je da predloženi hibridni model pruža bolje rezultate u usporedbi s postojećim pristupima, smanjujući pogrešku detekcije. Ovi rezultati ukazuju na potencijalnu primjenu u stvarnim sustavima za povećanje sigurnosti na cestama.

Ključne riječi: duboko učenje; transformatori; računalni vid; sigurnost u prometu; analiza slike

Abstract

Estimation of driver's gaze direction based on image analysis

Dario Dugonjevac

This paper investigates a method for estimating the driver's gaze direction based on image analysis, utilizing deep learning and transformer models. The goal is to reduce the number of traffic accidents caused by driver distraction by developing a system that detects the gaze direction and alerts the driver when they are not looking at the road. Testing on available datasets has shown that the proposed hybrid model provides better results compared to existing approaches, reducing detection error. These results indicate potential applications in real-world systems to enhance road safety.

Keywords: deep learning; transformers; computer vision; road safety; image analysis