

Detekcija objekata na snimkama snimljenih dronom

Frlan, Rene

Master's thesis / Diplomski rad

2024

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Rijeka / Sveučilište u Rijeci**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:195:354974>

Rights / Prava: [In copyright](#)/[Zaštićeno autorskim pravom.](#)

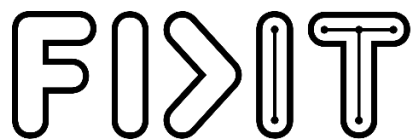
Download date / Datum preuzimanja: **2025-03-13**



Repository / Repozitorij:

[Repository of the University of Rijeka, Faculty of Informatics and Digital Technologies - INFORI Repository](#)





Sveučilište u Rijeci

**Fakultet informatike
i digitalnih tehnologija**

Sveučilišni diplomski studij Informatika

Rene Frlan

Detekcija objekata na snimkama snimljenih dronom

Diplomski rad

Mentor: prof. dr. sc. Marina Ivašić Kos

Rijeka, 25 srpnja 2024

Rijeka, datum

Zadatak za diplomski rad

Pristupnik/ica: Rene Frlan

Naziv diplomskog rada: **Detekcija objekata na snimkama snimljenih dronom**

Naziv diplomskog rada na eng. jeziku: **Object detection in drone footage**

Sadržaj zadatka:

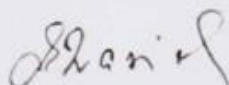
Proučiti i opisati osnovne koncepte dubokih neuronskih mreža s naglaskom na arhitekturu konvolucijske neuronske mreže, aktivacijske funkcije i hiperparametre, te popularnih metoda koje se koriste kod rješavanja zadataka računalnog vida kao što je detekcija objekata. Istražiti popularne arhitekture konvolucijskih mreža i dati njihov povijesni pregled.

Istražiti alate koji se koriste za učenje modela neuronske mreže i odabrati arhitekturu neuronske mreže koja će se koristiti za detekciju objekata u specifičnom zadatku detekcije osoba snimljene dronom.

Prikupiti slike osoba snimljenih dronom s javno dostupnih baza slika ili napraviti vlastiti skup slika. Pripremiti podatke za učenje ili fino podešavanja modela za detekciju osoba. Opisati postupak učenja modela za detekciju osoba zajedno s podešavanjem parametra, optimizacijom. Koristeći standardne metrike evaluirati rezultate i objasniti najbolji rezultat koji je postignut.

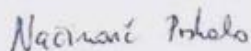
Mentor/ica:

Prof. dr. sc. Marina Ivašić-Kos



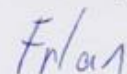
Voditeljica za diplomske radove:

Doc. dr. sc. Lucia Načinović Prskalo



Komentor/ica:

Zadatak preuzet: datum



(potpis pristupnika/ce)

Sažetak

Ovaj diplomski rad istražuje primjenu dronova u detekciji objekata, s posebnim fokusom na detekciju osoba. Primarni cilj bio je razviti i usporediti izvedbu pet modela: prilagođene konvolucijske neuronske mreže (CNN), prilagođenog modela YOLOv8n, YOLOv8s i standardnih verzija tih modela. Metodologija je uključivala prikupljanje i označavanje opsežnog skupa podataka snimljenog dronom, odabir odgovarajućih arhitektura dubokih neuronskih mreža za zadatak detekcije osoba, treniranje modela i ocjenjivanje njihove izvedbe na temelju standardnim metrika za točnost i preciznosti. Rezultati su pokazali da je prilagođeni model YOLOv8s nadmašio i prilagođeni CNN i sve ostale YOLOv8 modele u pogledu točnosti i učinkovitosti obrade. Rad zaključuje da je YOLOv8, kada je fino podešen, vrlo učinkovit za otkrivanje osoba u stvarnom vremenu pomoću dronova.

Ključne riječi: Dronovi; prepoznavanje objekata; prepoznavanje osoba; duboke konvolucijske neuronske mreže; YOLOv8; strojno i duboko učenje; računalni vid

SADRŽAJ

1. Uvod	1
2. Računalni vid i detekcija objekta	4
2.1. Povijest.....	4
3. Konvolucijske neuronske mreže	5
3.1. Povijest.....	8
3.2. YOLO familija modela	9
2.2.1. Vrste YOLOv8 modela	11
3.2.2. Povijest.....	12
4. Radna okolina za provedbu eksperimenta	13
4.1 Dron	14
4.2 Windows podsustav za Linux - WSL	15
3. Provedba eksperimenta za detekciju osoba pomoću dronova.....	17
3.1. Snimanje video datoteka	18
3.2. Izdvajanje kadrova	19
3.3. Anotacija slika i priprema podataka	20
3.4. Podjela podatka na skup za treniranje i testiranje	22
3.5. Treniranje modela	23
3.5.1. YOLOv8	23
3.5.2. CNN	25
3.6. Vizualizacija i validacija rada modela	27
4. Rezultati	28
4.1. YOLOv8 Nano.....	29
4.2 YOLOv8 Small	30
4.3. Personalizirani YOLOv8 Nano.....	31
4.4. Personalizirani YOLOv8 Small	32
4.5. Vlastiti CNN	34
4.6. Analiza rezultata	35
2.4. Usporedba YOLOv8 i vlastitog CNN-a.....	35
2.5. Prednosti	35
2.5.1. YOLOv8	36
2.5.2. CNN	36

2.6. Mane	36
2.6.1. YOLOv8	36
2.6.2. CNN	37
5. Zaključak.....	38
Literatura.....	39
Popis tablica	41
Popis slika	42
Popis priloga	43

1. Uvod

Pojava tehnologije bespilotnih letjelica (UAV) ili dronova značajno je transformirala različite sektore, od poljoprivrede i logistike do nadzora i odgovora u hitnim slučajevima. Dronovi nude jedinstvenu prednost i neusporedivu fleksibilnost, što ih čini nezamjenjivima u modernim tehnološkim primjenama. Opremljeni su raznovrsnim naprednim senzorima koji uključuju GPS, infracrvene senzore, senzore za prepoznavanje prepreka, te senzore za mjerenje temperature. Osim toga, posjeduju visokokvalitetne kamere koje mogu biti različitih tipova, poput kamera s visokim zumom, termalnih kamera i standardnih kamera. Ovi različiti tipovi kamera omogućuju precizno snimanje područja koje dron nadlijeće, pružajući sveobuhvatne informacije o promatranom terenu. Među raznovrsnim upotrebama bespilotnih letjelica jedno od najkritičnijih područja koje se brzo razvija je detekcija objekata, odnosno osoba. Detekcija osoba uključuje identifikaciju i lociranje osobe unutar slika ili video okvira snimljenih kamerom.

Ovaj diplomski rad bavi se kompleksnim procesom detekcije osoba snimljenih dronom, koristeći i podešavajući nove metode dubokog učenja koje omogućavaju u realnom vremenu s limitiranim računalnim resursima detekciju osoba snimljenih dronom tijekom leta.

Mogućnost što točnije detekcije osobe ili objekta snimljene s dronom, u realnom vremenu ključna je u svim primjenama u kojima se dronovi koriste kao pomoć jer ona izravno utječe na učinkovitost i odziv operacija u kojima se primjenjuju. u nastavku su navedene neke od najvažnijih:

- misije traganja i spašavanja, gdje dronovi mogu brzo pokriti velika, teško dostupna područja kako bi locirali nestale osobe;
- u sigurnosti i nadzoru u cilju povećanje javne sigurnosti na događajima s velikim brojem ljudi, u zračnim lukama i na granicama, gdje mogu kontinuirano nadzirati javne prostore, autonomno patrolirati područjem i pratiti i identificirati sumnjive aktivnosti u stvarnom vremenu pomažući u prevenciji i odgovoru na sigurnosne prijetnje,
- za upravljanjem velikim događajima, prosvjedima ili mjestima zločina i osiguranje javne sigurnosti,
- upravljanju katastrofama, gdje mogu pomoći u procjenjivanju situacije, te lociranje preživjelih,
- upravljanjem gradilištem i nadzor provedbe poslova i radova
- u očuvanju divljih životinja na način da pomažu u detekciji i praćenju ljudske prisutnosti u zaštićenim područjima kako bi spriječili krivolov i ilegalne aktivnosti
- za praćenje kretanja i ponašanja životinja u njihovim prirodnim staništima bez uznemiravanja
- u poljoprivredi, za praćenje stanja usjeva i vegetacije, kao pomoć u poljoprivrednim aktivnostima ili praćenje rada radnika i opreme.

Ukratko, svrha ovog istraživanja je iskoristiti napredne tehnike strojnog učenja i najsuvremenije modele poput YOLOv8 i CNN-a kako bi se poboljšala izvedba sustava za detekciju osoba pomoću dronova u različitim domenama primjene.

Integracija naprednih algoritama dubokog učenja i tehnika računalnog vida imala je značajan utjecaj na detekciju objekata. Tradicionalne metode, koje se oslanjaju na ručno izlučivanje značajki i heuristički pristupe, često podbacuju u pogledu točnosti i prilagodljivosti različitim uvjetima te nisu postizale veliku upotrebnu vrijednost kod detekcije objekata izvan laboratorijskih testiranja. Suprotno tome, moderni pristupi iskorištavaju snagu dubokog učenja, posebno dubokih konvolucijske neuronske mreže (CNN), za automatsko izlučivanje značajka i prepoznavanje složenih obrazaca iz golemih količina podataka postižu respektabilne rezultata u stvarnom okruženju i u realnim uvjetima. Među najsuvremenijim modelima detekcije objekata, ističu se modeli iz obitelji You Look Only Once (YOLO [1]) se svojim mogućnostima obrade u stvarnom vremenu i visokom preciznošću. YOLO tretira detekciju objekta kao jedan regresijski problem, predviđajući granične okvire i vjerojatnosti klasa izravno iz punih slika u jednoj evaluaciji. Ova promjena paradigme s regionalnih pristupa, kao što je Faster R-CNN [2], na modele detekcije u jednom prolazu značajno je smanjila računalne troškove i poboljšala brzinu detekcije. YOLOv8 [1], uključuje nekoliko poboljšanja arhitekture i u postupcima treniranja, čineći je još robusnijom i učinkovitijom.

Motivacija za odabir detekcije osoba pomoću bespilotnih letjelica kao teme diplomskog rada proizlazi iz njezinog potencijala za povećanje javne sigurnosti i operativne učinkovitosti. Sposobnost dronova da pruže zračni nadzor u stvarnom vremenu i prikupe kritične informacije iz nedostupnih ili opasnih područja neprocjenjiva je. Usredotočujući se na detekciju osoba, ovaj rad ima za cilj pridonijeti razvoju pouzdanijih i učinkovitijih sustava koji se mogu primijeniti u različitim scenarijima, od urbanih sredina do regija pogođenih katastrofama.

U ovom radu detaljno je specificirano detekcija osoba s ciljem primjene u situacijama poput potrage za izgubljenim osobama u šumi (Slika 1). U ovakvim scenarijima, od ključne je važnosti da model detekcije bude sposoban precizno prepoznati ljude u različitim uvjetima, uključujući varijabilne svjetlosne uvjete, različite položaje i potencijalne prepreke u okolini. Stoga, u našem slučaju, objekt koji se detektira je osoba, a svrha ovog modela je omogućiti brzu i pouzdanu identifikaciju osoba kako bi se unaprijedili naponi u spašavanju i traženju nestalih osoba u prirodnim okruženjima. Razvijena su i uspoređena pet modela: prilagođeni CNN, prilagođeni YOLOv8 nano, YOLOv8 small model i zadani YOLOv8 nano i YOLOv8 small modeli. Prilagođeni CNN dizajniran je od nule, sa slojevima i parametrima fino podešenim za optimizaciju performansi za detekciju osoba snimljenih dronom. Prilagođeni model YOLOv8 uključivao je postojeću arhitekture YOLOv8 koja je dodatno trenirana na pripremljenom skupu podataka slika osoba snimljenih dronom, kako bi se model fino podešio da se poveća točnost detekcije osoba na slikama snimljenih dronom. Zadani model YOLOv8 poslužio je kao mjerilo, pružajući standard za procjenu poboljšanja postignutih prilagodbom i finim podešavanjem. Metodologija je uključivala sveobuhvatan proces prikupljanja podataka,

označavanja, treniranje tj. podešavanja modela i evaluacije rezultata. Sastavljen je raznolik skup podataka koji obuhvaća različita okruženja, uvjete osvjetljenja i izgled osoba kako bi se osigurala robusnost modela. Na pripremljenom skupu podataka trenirani je svaki model pod istim uvjetima učenja, a njihova izvedba procijenjena je na temelju standardnih metrika kao što su točnost, preciznost i prisjećanje. Komparativna analiza istaknula je prednosti i slabosti svakog pristupa, pružajući dragocjene uvide u učinkovitost različitih arhitektura modela i strategija učenja.

Rezultati su pokazali da je prilagođeni model YOLOv8 nadmašio prilagođeni CNN i zadani model YOLOv8 u pogledu točnosti detekcije i brzine obrade. Ovo istraživanje naglašava važnost prilagodbe modela i finog podešavanja njegovih težina za specifične primjene kako bi se postigli najbolji rezultati iako već postojeći modeli poput YOLOv8 nude čvrstu osnovu za primjenu, ali značajna poboljšanja mogu se ostvariti kroz ciljano treniranje i prilagodbe parametara.

Također je razvijena i vlastita CNN mreža prilagođena za detekciju osoba snimljenih dronom. Kako bi se usporedile performanse naučenih modela i kako bismo dobili sveobuhvatniji uvid u učinkovitost i točnost različitih pristupa detekciji osoba.

Ukratko, cilj ovog rada je unaprijediti područje detekcije osoba pomoću bespilotnih letjelica istraživanjem i usporedbom različitih arhitektura modela i metodologija treniranja. Uvidi dobiveni ovim istraživanjem ne samo da pridonose akademskom razumijevanju detekcije osoba pomoću bespilotnih letjelica, već imaju i praktične implikacije za povećanje učinkovitosti i djelotvornosti aplikacija u stvarnom svijetu.



Slika 1 Primjer automatske detekcije objekta

2. Računalni vid i detekcija objekta

Računalni vid je područje umjetne inteligencije koje se bavi razvojem metoda koje omogućuju izlučivanje značajnih informacija iz digitalnih slika i videa i njihovu interpretaciju koristeći metode strojnog učenja kao što je npr. detekcija dvodimenzionalnih i/ili trodimenzionalnih objekata na slici – npr. ljudskog lica, osobe, psa, mačke, automobila, aviona, ili bilo kojeg drugog objekta od interesa.

Detekcija objekata osnovni je zadatak računalnog vida koji uključuje raspoznavanje i lociranje objekata unutar slike ili videa. Ovaj je zadatak ključan za širok raspon primjena, uključujući autonomnu vožnju, videonadzor i robotiku [3]. Sustavi za detekciju objekata moraju precizno detektirati i klasificirati različite objekte u stvarnom vremenu, što ih čini vrlo složenim i računalno intenzivnim [4].

Tradicionalne metode detekcije objekata uvelike su se oslanjale na ručno izlučene značajke i klasifikatore. Ove metode, kao što je Viola-Jones detektor, bile su ograničene svojom jednostavnom arhitekturom koja nije omogućavala dobru generalizaciju različite okoline i varijacije oblika u kojima se isti objekt može pojaviti. Pojava dubokog učenja revolucionirala je detekciju objekata, omogućujući razvoj modela koji automatski izlučuju značajke iz podataka.

Moderni detektori objekata koriste duboke neuronske mreže i mogu se kategorizirati u dvije glavne vrste: jednostupanjski detektori objekata i dvostupanjski. Dvostupanjski detektori objekata kao što je R-CNN i njegove varijante (Brzi R-CNN, Brži R-CNN), najprije izlučuju prijedloge regija od interesa (RoI) na kojima postoji vjerojatnost da se objekt nalazi, a potom te regije klasificiraju u odgovarajuće klase [5]. Iako u pravilu postižu veću točnost nego druga vrsta metoda, često su spore i računalno zahtjevne.

Jednostupanjski detektori, kao što su Single Shot MultiBox Detector (SSD) i You Only Look Once (YOLO), rade sve u jednom prolazu i izravno predviđa granične okvire i oznake klasa za svaki granični. Ove metode nemaju postupak ekstrakcije regije od interesa već postavljaju predefinirane usidrene okvire u slici i klasifikaciju područja slike koje okvir obuhvaća tretiraju kao regresiju. Rezultat detekcije su područja slike koja imaju najveću pouzdanost da sadrže objekt i oznaka objekta. Ovaj pristup značajno smanjuje troškove računanja i omogućuje detekciju objekata u stvarnom vremenu [6].

2.1. Povijest

Razvoj računalnog vida ima svoj začetak u ranim danima razvoja umjetne inteligencije, početkom 1960-ih, kada su istraživači prvi put počeli razmišljati o tome kako računala mogu interpretirati slike. Jedan od pionira bio je Larry Roberts, koji je 1963. godine napisao svoju doktorsku disertaciju na MIT-u o mogućnostima strojnog prepoznavanja trodimenzionalnih objekata iz dvodimenzionalnih slika [7].

U ranim fazama, istraživanje računalnog vida bilo je usmjereno na jednostavne zadatke poput prepoznavanja rubova i osnovnih oblika. Jedan od prvih značajnih doprinosa bila je metoda

Houghove transformacije, koju je Paul Hough razvio 1962. godine. Ova metoda omogućila je detekciju ravnih linija na slikama, što je bio važan korak naprijed u analizi slike [8].

U 1970-ima, istraživanja su se usredotočila na razumijevanje scena i interpretaciju objekata u složenijim okruženjima. Pogled na računalni vid značajno je promijenjen razvojem algoritama za analizu pokreta i strukture. Na primjer, 1979. godine David Marr s MIT-a objavio je radove koji su postavili temelje za računalno prepoznavanje slika kroz teoriju primarne skice, gdje je analizirao slojeve obrade slike, od primarne do trodimenzionalne reprezentacije [7].

1980-ih i 1990-ih, naglasak je bio na razvoju metoda za izlučivanje značajki i algoritama za prepoznavanje uzoraka. Detekcija objekata kao specifičan zadatak počinje se intenzivnije istraživati u tom razdoblju. Klasične metode uključivale su korištenje ručno izlučenih značajki, poput SIFT (Scale-Invariant Feature Transform [9]) i SURF (Speeded-Up Robust Features [9]), koje su omogućavale prepoznavanje objekata neovisno o ljestvici i rotaciji.

Prvi značajan skok u detekciji objekata dogodio se 2001. godine s pojavom Viola-Jones detektora lica [5]. Ovaj algoritam, kojeg su razvili Paul Viola i Michael Jones, omogućio je brzu i relativno točnu detekciju lica u stvarnom vremenu, čime je otvorio put za široku primjenu u komercijalnim uređajima poput digitalnih kamera i sustava za nadzor.

Prava revolucija u detekciji objekata nastupila je s razvojem dubokog učenja i dubokih konvolucijskih neuronskih mreža (CNN) u 2010-ima. AlexNet [10], model koji je pobijedio na natjecanju ImageNet 2012., pokazao je superiorne performanse u klasifikaciji slika koristeći duboke neuronske mreže. Ovaj uspjeh potaknuo je daljnji razvoj složenijih modela poput R-CNN (Regions with Convolutional Neural Networks) koji su kombinirali prijedloge regija s konvolucijskim mrežama za preciznu detekciju objekata [2].

Daljnje varijante R-CNN-a, kao što su Fast R-CNN i Faster R-CNN, optimizirale su proces detekcije smanjujući vrijeme izvođenja i poboljšavajući točnost [2]. Paralelno, razvijeni su jednostupanjski detektori poput SSD (Single Shot MultiBox Detector) i YOLO (You Only Look Once) koji omogućuju detekciju objekata u stvarnom vremenu predviđanjem graničnih okvira i klasifikacija u jednom prolazu [11].

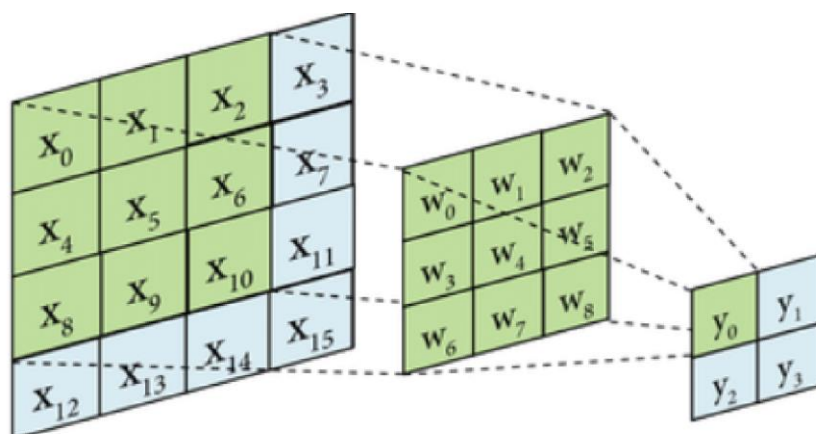
Danas, detekcija objekata koristi se u širokom rasponu primjena, od autonomnih vozila i industrijske automatizacije do medicinske dijagnostike i sustava za nadzor. Razvoj računalnog vida i detekcije objekata ne prestaje, s kontinuiranim istraživanjima usmjerenim na povećanje točnosti, brzine i robusnosti modela u stvarnim uvjetima. U budućnosti, očekuje se daljnji napredak u integraciji s drugim tehnologijama poput interneta stvari (IoT) i proširene stvarnosti (AR), čime će se dodatno proširiti mogućnosti i primjene računalnog vida.

3. Konvolucijske neuronske mreže

Konvolucijske neuronske mreže (CNN) su klasa dubokih neuronskih mreža koje su revolucionirale zadatke računalnog vida, uključujući klasifikaciju slika, segmentaciju i detekciju objekata. CNN-ovi su dizajnirani da automatski i adaptivno uče prostorne hijerarhije značajki iz ulaznih slika, što ih čini vrlo učinkovitim za zadatke vizualnog prepoznavanja [12].

Tipična CNN arhitektura sastoji se od nekoliko ključnih slojeva:

1. Konvolucijski slojevi (slika 2): U ovim slojevima primjenjuje se operacija konvolucije na ulaznu sliku ili mapu, detektirajući lokalne uzorke kao što su rubovi, teksture i oblici. Svaki filter stvara mapu značajki koja ističe specifične vizualne značajke sadržane u slici.



Slika 2 Operacija konvolucije

2. Sloj pojednostavljivanja (Polling layer) (slika 3): Smanjuje uzorkovanje i dimenziju mape značajki na način da umjesto vrijednosti svake težine u mapi uzima u obzir statistiku lokalne okoline (pravokutnog susjedstva) za primijenjeni filter zadržavajući važne informacije iz reprezentacije. Ovaj proces pomaže u postizanju invarijantnosti modela i smanjuje računsku složenost. Postoji nekoliko funkcija koje se koriste u ovom koraku kao što su prosjek, L2 norma ili najveća vrijednost pravokutnog susjedstva. Najveća vrijednost maksimalno susjedstva se najčešće koristi.

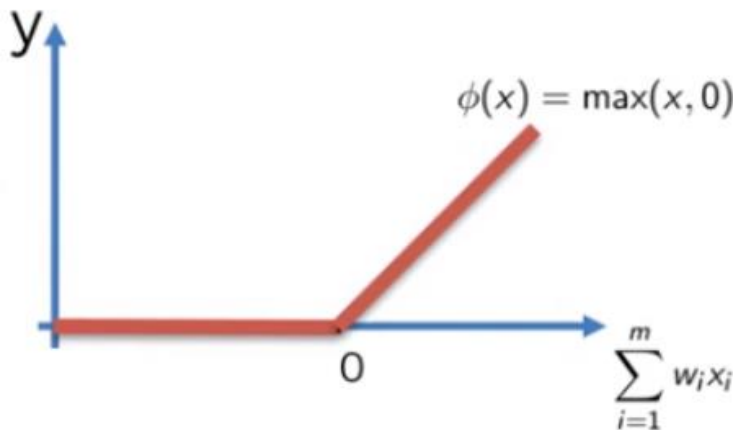


Slika 3 Max pool operacija

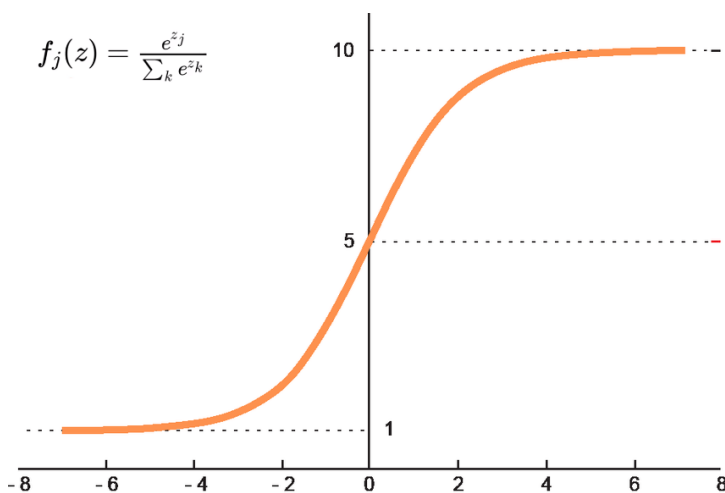
4. Potpuno povezani slojevi: Ovi se slojevi koriste za klasifikaciju ili regresiju. Oni povezuju sve neurone iz prethodnog sloja sa svakim neuronom u trenutnom sloju zbog čega se mogu izračunati kao i potpuno povezane neuronske mreže (FCNN) koja se izračunava običnim matričnim množenjem. Ovi slojevi omogućavaju mreži da integrira značajke i donosi zaključke.

5. Aktivacijske funkcije: Nelinearne aktivacijske funkcije, kao što je ReLU (Rectified Linear Unit), uvode nelinearnost u mrežu, omogućujući joj da nauči složene obrasce između ulaznih i izlaznih podataka (slika 4).

Funkcija Softmax obično se koristi u zadnjem sloju neuronske mreže za predviđanje klase ulazne slike ili objekta na slici (slika 5). Ona normalizira ulazne vrijednosti u distribuciju vjerojatnosti, osiguravajući da zbroj svih izlaznih vrijednosti bude 1 što je čini prikladnom za probleme klasifikacije u više klasa gdje svaki element izlaznog vektora predstavlja vjerojatnost da ulaz pripada odgovarajućoj klasi.

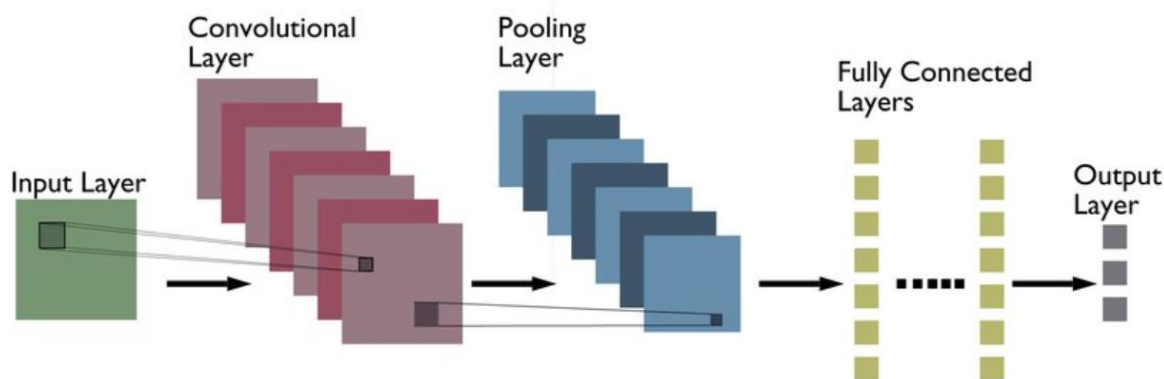


Slika 4 ReLU Funkcija i graf



Slika 5 Softmax funkcija i graf

CNN-ovi se treniraju korištenjem velikih označenih skupova podataka i optimizacijskih algoritama kao što je stohastički gradijentni spust (SGD) (slika 6). Tijekom treniranja, mreža prilagođava svoje težine kako bi minimizirala funkciju gubitka koja kvantificira pogrešku između predviđenih i pravih oznaka.



Slika 6 Standardna arhitektura CNN modela

3.1. Povijest

Konvolucijske neuronske mreže (CNN) najčešće se koriste u području računalnog vida. Njihov razvoj revolucionirao je prepoznavanje slika, klasifikaciju i brojne druge primjene, omogućujući značajan napredak u području umjetne inteligencije.

Rani počeci koncepta CNN-a može se pratiti unazad do 1980-ih s radom Kunihika Fukushime, koji je razvio Neocognitron 1980 [7]. Neocognitron je bila rana hijerarhijska, višeslojna umjetna neuronska mreža inspirirana ljudskim vizualnim sustavom. Postavio je temelje za budući razvoj CNN-a uvođenjem ideje o konvolucijskim slojevima i hijerarhijskom izdvajanju značajki.

LeNet-5 (1998)

Veliki napredak dogodio se 1998. godine kada su Yann LeCun i njegovi suradnici predstavili LeNet-5 [13]. Ovaj je model dizajniran za prepoznavanje rukom pisanih znamenki i bio je značajan napredak u praktičnoj primjeni CNN-a. LeNet-5 je sadržavao nekoliko slojeva konvolucije i poduzorkovanja, praćenih potpuno povezanim slojevima, tvoreći robusnu arhitekturu koja je pokazala potencijal CNN-a u zadacima prepoznavanja slike.

AlexNet (2012)

Sljedeća značajna prekretnica dogodila se 2012. s uvođenjem AlexNeta od strane Alexa Križhevskog, Ilye Sutskevera i Geoffreyja Hinton [7]. AlexNet je pobijedio na ImageNet Large Scale Visual Recognition Challenge (ILSVRC) velikom razlikom, značajno smanjujući postotak pogreške u usporedbi s prethodnim metodama [10]. Model je sadržavao 8 slojeva, ReLU aktivacijske funkcije, ispadanje za regularizaciju i GPU ubrzanje, označavajući početak revolucije dubokog učenja. Za ono vrijeme, to je bilo izuzetno puno slojeva, ali u usporedbi s današnjim modelima, ovaj broj se smatra vrlo malim.

VGGNet (2014)

Godine 2014. Visual Geometry Group iz Oxforda predstavila je VGGNet [14], koji je dodatno pokazao važnost dubine u CNN arhitekturama. VGGNet je koristio vrlo male (3x3) konvolucijske filtre, ali je povećao broj slojeva na 16-19, jedan od najpoznatijih modela iz ove serije, imao je oko 138 milijuna parametara. Ovaj pristup pokazao je da dublje mreže mogu postići bolje performanse u zadacima klasifikacije slika.

GoogLeNet i početak (2014.)

Također 2014. Google je predstavio GoogLeNet [14], temeljen na arhitekturi Inception. Ovaj model uveo je koncept "početnih modula", koji je omogućio učinkovitije izračunavanje kombiniranjem višestrukih vijuga različitih veličina unutar istog sloja. GoogLeNet je imao 22 sloja, ali je zahvaljujući svojoj inovativnoj arhitekturi uspio zadržati broj parametara na oko 5 milijuna, što je značajno manje u usporedbi s VGGNetom. Ovaj model je pobijedio na natjecanju ILSVRC 2014, ističući učinkovitost svoje inovativne arhitekture.

ResNet (2015)

Kaiming He i njegov tim uveli su Residual Networks (ResNet [14]) 2015. godine i riješili su problem treniranja vrlo dubokih mreža. ResNet je koristio "preskakanje veza" ili "preostale veze" kako bi omogućio gradijentima da zaobiđu jedan ili više slojeva, omogućavajući obuku mreža sa stotinama ili čak tisućama slojeva. ResNet-50, jedan od popularnih modela iz ove serije, imao je 50 slojeva i oko 25 milijuna parametara. Ovaj pristup značajno je poboljšao performanse i pobijedio na natjecanju ILSVRC 2015.

Od ResNeta razvijena su brojna poboljšanja i varijacije CNN-a, uključujući DenseNet, EfficientNet i druge [15]. Ovi modeli uključuju nove tehnike i optimizacije, kao što je ponovna upotreba značajki, složeno skaliranje i bolje metode inicijalizacije, neprestano pomičući granice onoga što CNN može postići.

Danas su CNN-ovi sveprisutni u raznim zadacima računalnog vida. Koriste se u detekciji objekata, semantičkoj segmentaciji, analizi medicinske slike, analizi videa, pa čak i zadacima obrade prirodnog jezika. Njihova sposobnost da automatski uče hijerarhijske značajke iz podataka čini ih nezamjenjivima u modernoj umjetnoj inteligenciji. Ukratko, povijest CNN-a obilježena je stalnim inovacijama i usavršavanjem. Od ranih koncepata Neocognitrona do današnjih sofisticiranih arhitektura, CNN-ovi su iz temelja transformirali računalni vid i umjetnu inteligenciju, pokrećući napredak u tehnologiji i industriji.

3.2. YOLO familija modela

YOLO je familija najsuvremeniji metoda za otkrivanje objekata koje su stekle široku popularnost zbog svoje brzine i točnosti. Za razliku od metoda temeljenih na regiji, YOLO detekciju objekta tretira kao pojedinačni regresijski problem, predviđajući granične okvire i

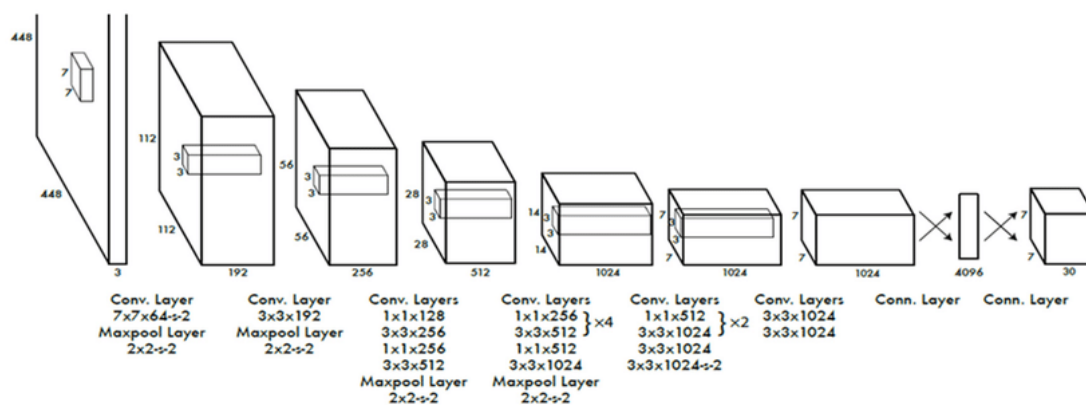
vjerojatnosti klasa izravno iz slika u jednoj evaluaciji. Ovaj objedinjeni pristup omogućuje YOLO-u postizanje performansi u stvarnom vremenu uz održavanje visoke točnosti detekcije [11].

YOLOv8, novija iteracija YOLO obitelji, uvodi nekoliko poboljšanja u odnosu na svoje prethodnike. Ova poboljšanja uključuju učinkovitiju mrežnu arhitekturu, poboljšanu generaciju sidrenih okvira i napredne tehnike treniranja. Arhitektura YOLOv8 dizajnirana je kako bi održala ravnotežu između brzine i točnosti, što ga čini prikladnim za aplikacije u stvarnom vremenu na uređajima s ograničenim resursima poput dronova. Ključno je osigurati dovoljan broj slika za treniranje modela kako bismo postigli vidljive i pouzdane rezultate. Veliki i raznovrsni skup podataka omogućuje modelu da nauči detekciju osobe u različitim situacijama i uvjetima, čime se poboljšava njegova sposobnost generalizacije i preciznosti u stvarnim aplikacijama. Što je veći i raznolikiji skup podataka, to su rezultati modela pouzdaniji i robusniji, omogućujući mu da bolje detektira osobe u različitim scenarijima.

Osnovne komponente YOLOv8 arhitekture uključuju (slika 7):

1. Kralježnica: Mreža kralježnice služi za izlučivanje značajki iz ulaznih slika i stvaranje mape značajki u kojoj su pohranjene bogate vizualne značajki u višestrukim slojevima.
2. Vrat: Mreža vrata povezuje značajke iz različitih slojeva kralježnice, poboljšavajući sposobnost modela da detektira objekte različitih veličina.
3. Glava: Mreža glave služi za detekciju i generiranje predviđanja graničnog okvira i vjerojatnosti klasa iz agregiranih mapa značajki. YOLOv8 koristi detekciju temeljenu na sidrištima, gdje se unaprijed definirane sidreni granični okviri koriste za predviđanje lokacija objekata.

Proces treniranja za YOLOv8 uključuje optimizaciju višedijelne funkcije gubitka koja uravnotežuje točnost lokalizacije, pouzdanost klasifikacije i broj otkrivenih objekata. Ova funkcija gubitka osigurava da model nauči napraviti precizna i pouzdana predviđanja.



Slika 7 Standardna arhitektura YOLO modela

2.2.1. Vrste YOLOv8 modela

YOLOv8 modeli dolaze u pet različitih verzija: nano, small, medium, large i extra large. Svaka verzija ima specifične karakteristike i koristi se u različitim scenarijima ovisno o zahtjevima preciznosti i dostupnim računalnim resursima [1].

Tablica 1 Usporedba performansi YOLO modela

Verzija	Broj Parametara	Performansa na COCO datasetu	Upotreba
YOLOv8 Nano	1.0 milijun	45%	Brza detekcija, uređaji s ograničenim resursima
YOLOv8 Small	3.5 milijun	55%	Real-time aplikacije, mobilni uređaji
YOLOv8 Medium	7.0 milijun	65%	Balans između preciznosti i brzine
YOLOv8 Large	15.0 milijun	75%	Visoka preciznost, dovoljno resursa
YOLOv8 XL	30.0 milijun	80%	Najviša preciznost, snažan hardver

1. YOLOv8 Nano

YOLOv8 Nano je najmanji i najbrži model iz YOLOv8 obitelji. Dizajniran je za primjene gdje su računalni resursi ograničeni, poput ugrađenih sustava, mobilnih uređaja i aplikacija u stvarnom vremenu. Nano model ima manji broj parametara i slojeva, što ga čini bržim za treniranje i izvođenje. Međutim, zbog svoje jednostavnosti, može imati manju točnost u usporedbi s većim modelima.

2. YOLOv8 Small

YOLOv8 Small je sljedeći korak iznad Nano modela. Ima nešto složeniju arhitekturu, što rezultira boljom točnošću detekcije uz umjereno povećanje računalnih zahtjeva. Small model je prikladan za aplikacije gdje je potrebna balans između brzine i preciznosti, poput sigurnosnih kamera ili autonomnih vozila.

3. YOLOv8 Medium

YOLOv8 Medium model nudi još veću točnost i robusnost u detekciji objekata. S većim brojem slojeva i parametara, Medium model zahtijeva značajno više računalnih resursa za treniranje i

izvođenje. Ovaj model je prikladan za situacije gdje je preciznost ključna, kao što su medicinska dijagnostika ili napredna istraživanja.

4. YOLOv8 Large

YOLOv8 Large model predstavlja još složeniju i precizniju verziju. Namijenjen je za vrlo zahtjevne aplikacije koje mogu podnijeti visoke računalne troškove. Model pruža izvanrednu točnost i može se koristiti u industrijskim aplikacijama gdje je precizna detekcija objekata od najveće važnosti

5. YOLOv8 ExtraLarge

YOLOv8 Extra Large je najmoćniji model u YOLOv8 seriji. Ima najviše slojeva i parametara, što rezultira najvećom točnošću, ali i najvišim računalnim zahtjevima. Ovaj model je prikladan za vrhunske istraživačke projekte i aplikacije koje zahtijevaju maksimalnu preciznost bez obzira na resurse.

U ovom radu fokus je stavljen isključivo na Nano i Small modele zbog činjenice da treniranje većih modela zahtijeva značajnu računalnu moć i puno vremena. Na primjer, treniranje Medium YOLOv8 modela na NVIDIA GeForce GTX 1060 grafičkoj kartici traje više od sat i petnaest minuta po jednoj epohi. Takvi zahtjevi čine treniranje većih modela nepraktičnim u postojećem okruženju. Stoga su Nano i Small modeli odabrani kao optimalna rješenja, pružajući dobar balans između performansi i potrebnih resursa.

3.2.2. Povijest

YOLO (You Only Look Once) modeli značajno su utjecali na područje detekcije objekata u računalnom vidu. Razvio ga je Joseph Redmon i njegov tim [16].

YOLOv1 (2015)

Prva verzija, YOLOv1, predstavljena je u radu Redmona i sur. iz 2015. Pod nazivom "You Only Look Once: Unified, Real-Time Object Detection" [17], model je predstavio novi pristup. Umjesto primjene klasifikatora na različite regije slike, YOLO je preoblikovao detekciju objekta kao pojedinačni regresijski problem, predviđajući granične okvire i vjerojatnosti klasa izravno iz ulaznih slika u jednoj evaluaciji. Ova je inovacija omogućila YOLOv1 da postigne impresivne brzine, obrađujući slike u stvarnom vremenu.

YOLOv2 (2016)

Godine 2016. Redmon i njegovi kolege objavili su YOLOv2 [17], također poznat kao YOLO9000. Ova je verzija poboljšana u odnosu na svoju prethodnicu uvođenjem skupne normalizacije, klasifikatora visoke razlučivosti i sidrenih okvira. Također je sadržavao novu metodu treniranja koja je modelu omogućila predviđanje preko 9000 klasa objekata. YOLOv2 ima uravnoteženu brzinu i točnost, što ga čini svestranijim za razne primjene.

YOLOv3 (2018)

YOLOv3, objavljen 2018., donio je dodatna poboljšanja. Predstavio je dublju mrežu s 53 konvolucijska sloja, poznatu kao Darknet-53, za bolje izlučivanje značajki. YOLOv3 je također implementirao predviđanja u više razmjera, omogućujući modelu točniju detekciju objekata različitih veličina. Ova verzija zadržala je mogućnosti obrade u stvarnom vremenu uz je poboljšanje performansi.

YOLOv4 (2020.)

YOLOv4, koji je 2020. objavio Alexey Bochkovskiy, nastavio je evoluciju YOLO obitelji. Ova je verzija integrirala nekoliko najsuvremenijih tehnika kao što su CSPDarknet53, Mish aktivacija i korištenje Path Aggregation Network (PAN). YOLOv4 je postigao izvanrednu ravnotežu između brzine i točnosti, što ga čini vrlo učinkovitim za praktične primjene.

YOLOv5 (2020.)

Kasnije 2020., YOLOv5 je izdala zajednica otvorenog koda, predvođena Ultralyticsom. Iako ga izvorni kreatori nisu razvili, YOLOv5 je stekao popularnost zbog svoje jednostavne upotrebe i implementacije. Nudi poboljšanja u treniranju, implementaciji i optimizaciji modela, čineći ga dostupnim široj publici.

YOLOv6 i dalje

Naknadne verzije, uključujući YOLOv6 i dalje, nastavile su graditi na temeljnim načelima YOLO arhitekture. Ove verzije uključuju napredne tehnike i optimizacije, pomičući dalje granice detekcije objekata u stvarnom vremenu.

Ukratko, YOLO obitelj modela značajno je unaprijedila područje detekcije objekata, kombinirajući mogućnosti obrade u stvarnom vremenu s visokom preciznošću. Od YOLOv1 revolucionarnog pristupa do sofisticiranih verzija današnjice, YOLO modeli nastavljaju biti kamen temeljac računalne vizije, pokrećući inovacije i praktične primjene u raznim industrijama.

4. Radna okolina za provedbu eksperimenta

Primarna svrha ovog istraživanja je povećati točnost i učinkovitost detekcije osoba snimljenih iz ptičje perspektive pomoću dronova [5]. Ova je sposobnost ključna za različite aplikacije u stvarnom svijetu, uključujući:

4.1 Dron

Za izradu ovoga rada korišten je DJI Mavic Air 2 (Slika 8) je jedan od najpopularnijih dronova na tržištu od 2020. godine kada je predstavljen, poznat po svojoj svestranosti, performansama i kvaliteti izrade [18].

Mavic Air 2 ima prepoznatljiv sklopivi dizajn koji je postao sinonim za DJI-ove dronove, što im omogućuje da su prenosivi i jednostavni za nošenje. Težina Mavic Air 2 je samo 570 grama. Dron ima maksimalno vrijeme leta od 34 minute, zahvaljujući bateriji od 3500 mAh, lako je upravljiv i uključuje napredne sigurnosne zaštite. Kamara drona ima 1/2-inčni CMOS senzorom koji omogućuje snimanje fotografija visoke rezolucije od 48 MP i video rezoluciji od 4K Ultra HD pri 60 fps (sličica u sekundi).

Specifikacije kamere uključuju:

Rezolucija fotografija: 48 MP (8000x6000 piksela)

Video rezolucija: 4K Ultra HD pri 60 fps, 2.7K pri 60 fps, 1080p pri 240 fps

Objektiv: FOV (Field of View) od 84°, otvor blende f/2.8, ekvivalent žarišne duljine 24 mm

ISO raspon: 100-6400 za video, 100-3200 za fotografije (može se proširiti na 100-6400)

Video kodiranje: H.264 i H.265

Bitrate: Maksimalno 120 Mbps

Dodatne značajke: SmartPhoto, za automatski odabir najboljih postavki za snimanje, kombinirajući HDR, HyperLight i priznavanje scena; 8K Hyperlapse, za timelaps videozapise u 8K rezoluciji što uz ActiveTrack 3.0, omogućuje precizno praćenje objekata u pokretu, čineći Mavic Air 2 pogodnim za snimanje akcijskih snimaka i sportskih događaja.

Osim impresivne kamere, Mavic Air 2 je opremljen naprednim sustavom senzora na prednjoj, stražnjoj i donjoj strani kako bi detektirao i izbjegavao prepreke, čime se smanjuje rizik od nesreća. APAS (Advanced Pilot Assistance Systems) 3.0 omogućuje dronu da automatski zaobiđe prepreke tijekom leta.

Za upravljanje Mavic Air 2 dronom koristi se daljinski upravljače koji koristi OcuSync 2.0 tehnologiju koja omogućuje stabilnu i pouzdanu video vezu do 10 km, što je znatno poboljšanje u odnosu na prethodne modele. Također podržava 1080p prijenos uživo, pružajući jasan i oštar prikaz u stvarnom vremenu.



Slika 8 Korišteni dron

4.2 Windows podsustav za Linux - WSL

Windows podsustav za Linux (WSL) [19] je hibridno operativno okruženje koji je razvio Microsoft kako bi omogućio izvršavanje Linux binarnih izvršnih datoteka nativno za Windows 10 i Windows Server 2019 (Slika 9). WSL podržavajući različite distribucije Linuxa kao što su Ubuntu, Debian i Fedora te spaja robusnost Linux alata i uslužnih programa s korisnički prilagođenim sučeljem Windowsa omogućujući korisnicima instaliranje i pokretanje Linux aplikacija izravno iz Microsoft Storea. Ova integracija koristi prednosti oba okruženja i omogućuje prebacivanja između Windows i Linux bez ponovnog pokretanja sustava i pojednostavljeno upravljanje radnim procesom te uklanja potrebu za dual-boot konfiguracijama ili virtualnim strojevima.

GPU ubrzanje s WSL-om

Jedna od ključnih prednosti WSL-a, posebice WSL-a 2, je njegova podrška za GPU ubrzanje [19]. Treniranje modela dubokog učenja, kao što su YOLO (You Only Look Once) i konvolucijske neuronske mreže (CNN), računalno je intenzivna i ima značajne koristi od mogućnosti paralelne obrade GPU-a. Izvorna Windows okruženja obično predstavljaju izazove u korištenju GPU-a zbog problema s upravljačkim programom i kompatibilnošću, ali WSL 2 omogućuje izravan pristup NVIDIA GPU-ovima i učinkovito učenje modela unutar Linux okruženja na Windows stroju.

Izazovi postavljanja i konfiguracije

Konfiguriranje WSL-a za korištenje GPU-a je proces koji često zahtijeva detaljne prilagodbe na razini sustava i BIOS-a koji se sastoji od sljedećih koraka:

1. Instalacija WSL-a i distribucije Linuxa: Bilo je potrebno omogućiti WSL putem značajki sustava Windows ili naredbi PowerShell i instalirati željenu distribuciju Linuxa.

2. Nadogradnja na WSL 2: ažurirati WSL kernel i postaviti WSL 2 kao zadanu verziju kako bi se iskoristio poboljšane performanse i kompatibilnost sa sistemskim pozivima.
3. Instalacija GPU drivera: Instalacija odgovarajući NVIDIA GPU drivera koji su kompatibilni s WSL-om, uključujući CUDA toolkit i cuDNN biblioteke neophodne za aplikacije dubokog učenja.
4. Konfiguracija sustava: Prilagodba postavki sustava i BIOS-a, kao što je omogućavanje virtualizacije u BIOS-u i konfiguriranje varijabli sustava, kako bi se osigurala dostupnost GPU-a za WSL.
5. Postavljanje razvojnog okruženja: unutar distribucije Linuxa bilo je potrebno instalirati i konfigurirati alate strojnog učenja kao što su TensorFlow ili PyTorch za korištenje GPU-a.

Proces postavljanja WSL-a i proces konfiguracije trajao je otprilike dva dana, uključujući rješavanje problema i fino podešavanje sistemskih varijabli i postavki BIOS-a, ali isplatilo se jer osigurava da je razvojno okruženje optimizirano za duboko učenje i korištenje GPU-a koje je značajno smanjilo vrijeme treniranja modela i fleksibilnost tijekom rada.

```

rene@DESKTOP-D3HEL40: /mnt/h/frlan_diplomski$ ls
CNN_confusion_matrix.png          dataset
PREDLOZAK.docx                   dataset.yaml
README.md                          diplomski.docx
best_model_3_nano.pt              extract_frames.py
best_model_4_small.pt            frames
cnn_model.h5                      nan_frames
cnn_model_2.h5                   nan_videos
cnn_train.py                      nms
create_cnn_confusion_matrix.py    test.py
create_dataset.py                 yolo_best_nano_confusion_matrix.png
create_yolo_confusion_matrixes.py yolo_best_small_confusion_matrix.png
best_model_3_nano.pt              yolo_frames
best_model_4_small.pt            yolo_nano_confusion_matrix.png
cnn_model.h5                     yolo_small_confusion_matrix.png
cnn_model_2.h5                   yolo_train.py
cnn_train.py                     yolo_visualize_image.py
create_cnn_confusion_matrix.py    yolo_visualize_video.py
create_dataset.py                 yolov8m.pt
create_yolo_confusion_matrixes.py yolov8n.pt
dataset                           yolov8s.pt
dataset.yaml                      '~$EDLOZAK.docx'
diplomski.docx                   '~$plomski.docx'
extract_frames.py                 '~WRL0005.tmp'
frames
nan_frames
nan_videos
nms
test.py
yolo_best_nano_confusion_matrix.png
yolo_best_small_confusion_matrix.png
yolo_frames
yolo_nano_confusion_matrix.png
yolo_small_confusion_matrix.png
yolo_train.py
yolo_visualize_image.py
yolo_visualize_video.py
yolov8m.pt
yolov8n.pt
yolov8s.pt
'~$EDLOZAK.docx'
'~$plomski.docx'
'~WRL0005.tmp'

```

Slika 9 WSL Sučelje

3. Provedba eksperimenta za detekciju osoba pomoću dronova

Eksperiment se sastojao od nekoliko ključnih koraka, od kojih he svaki pridonio ukupnom uspjehu projekta. Prije samog početka snimanja, proveden je preliminarni test kako bi se utvrdilo kakve bi slike bile najpogodnije za treniranje YOLO modela. Iz testova je zaključeno da je potrebno snimati osobu iz različitih kutova, pri različitim brzinama drona i visinama kako bi se uključile sve moguće situacije (slika 10).



Slika 10 Razlika u odabranim okolinama snimanja

Također, eksperimentiralo se sa različitim postavkama kamere kako bi se pronašle optimalne postavke za jasnoću i prepoznatljivost slika. Nakon odabira lokacija, snimanje je provedeno tijekom nekoliko dana kako bi se prikupio dovoljan broj slika za treniranje modela. Svaki let snimanja je planiran tako da se obuhvate različite osobe u različitim pozicijama, čime se osigurala raznolikost podataka. Fotografije su prikupljene uz korištenje visokokvalitetne kamere montirane na dron. ukupno 26 videozapisa. Snimanje je obavljeno na različitim visinama, od 10 do 120 metara, pri različitim brzinama do 60 m/s. Na snimkama se nalaze 1 ili 2 osobe. Posebna pažnja posvećena je stabilnosti drona tijekom snimanja kako bi se minimizirale greške uzrokovane vibracijama tijekom leta.

Sljedeći korak bio je obrada prikupljenih podataka. Fotografije su izdvojene iz snimljenih videa i sortirane prema nazivu snimke. Ovaj proces označavanja bio je ključan za treniranje modela u nadziranom učenju, jer je omogućio algoritmu da pravilo spoji odgovarajuće slike i anotacije. Korišteni su napredni opensource alati za označavanje, labelImg, kako bi se osigurala visoka preciznost i dosljednost u podacima.

Nakon obrade podataka, započelo je treniranje svih modela. Testirani su različiti hiperparametri i pristupi učenja modela kako bi se postigli najbolji mogući rezultati. Svaka

iteracija modela testirana je i uspoređivana kako bi se kontinuirano poboljšavala točnost detekcije. Tijekom učenja, identificirani su i ispravljani eventualni problemi, te su provedene optimizacije kako bi se povećala učinkovitost modela.

Završni korak bio je testiranje i evaluacija razvijenog modela u stvarnim uvjetima. Model je testiran na testnom skupu koji je uključivao razne scenarije kako bi se procijenila njegova točnost i robusnost. Rezultati su analizirani i uspoređeni sa stvarnim podacima kako bi se potvrdila njegova uspješnost.

3.1. Snimanje video datoteka

Početni korak bio je postavljanje drona za snimanje video zapisa. Prije samog snimanja, bilo je potrebno odabrati odgovarajuće lokacije koje će omogućiti prikupljanje raznolikog skupa podataka. Kako bi se osigurao širok raspon uvjeta osvjetljenja, pozadina i scenarija, odlučeno je snimati u različitim okruženjima.

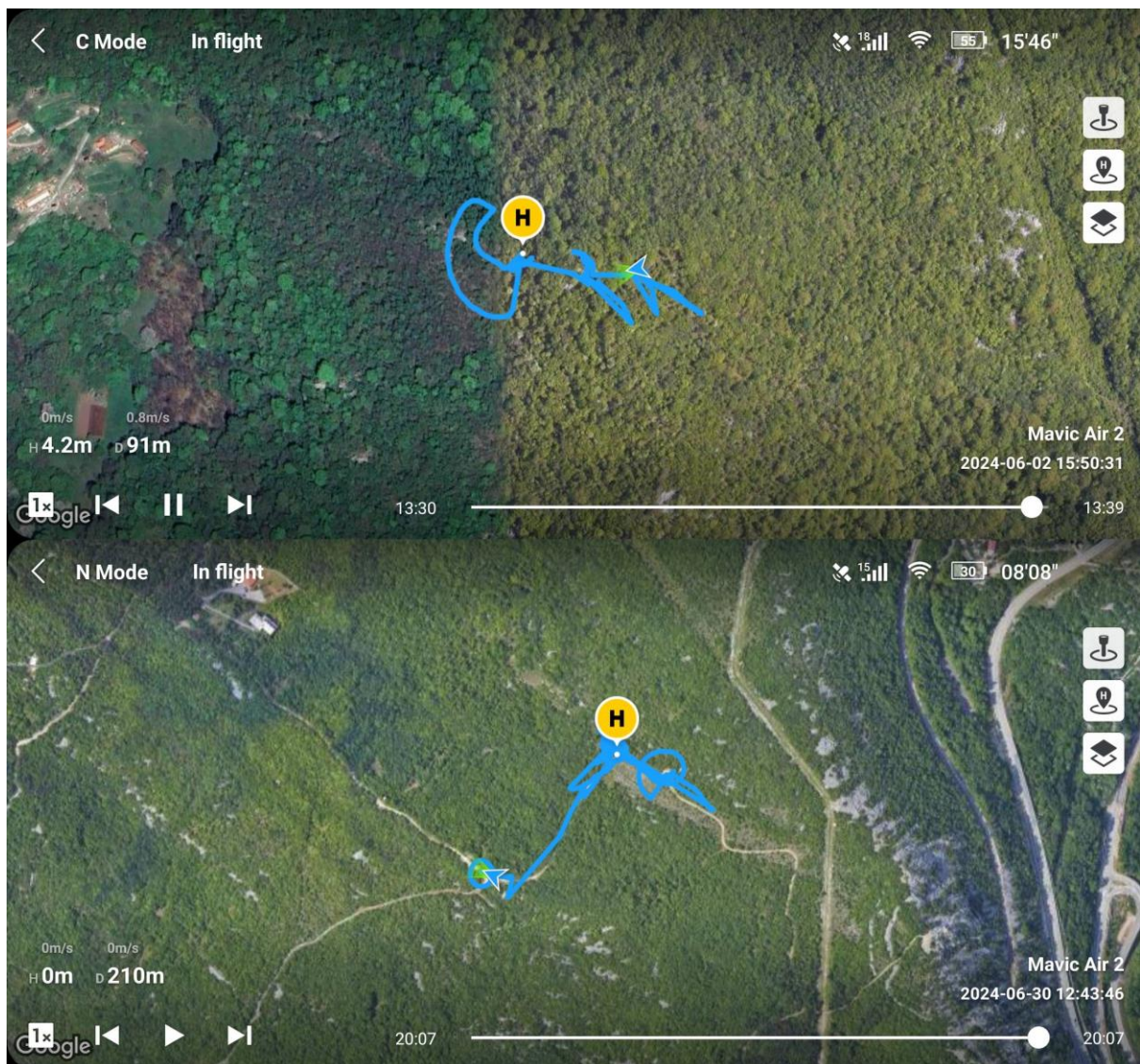
Dron je bio opremljen kamerom visoke rezolucije koja je omogućavala snimanje jasnih i detaljnih snimaka. Prva odabrana lokacija bila je gusta šuma s visokim stablima i gustim raslinjem, koja je pružala složeniju pozadinu za prepoznavanje osoba. Druga lokacija bila je rjeđa šuma s nižim grmljem i kamenitom podlogom, koja je omogućavala sasvim drugačije uvjete snimanja.

Dron je letio na različitim visinama i različitim brzinama kako bi se simulirale različite situacije i osigurala sveobuhvatnost prikupljenih podataka. Dron je letio oko objekata i snimao iz različitih perspektiva i kutova snimanja (slika 11). Tijekom snimanja, posebna pažnja posvećena je stabilnosti drona kako bi se izbjegle vibracije koje bi mogle utjecati na kvalitetu snimaka.

Ukupno je snimljeno 26 video datoteka u visokoj rezoluciji 1920x1080 piksela koje su korištene za treniranje modela. Snimke u prosjeku trajanja 2 minute, obuhvatile su različite scenarije, uključujući različite kutove snimanja, osvjetljenja i vremenske uvjete, što je osiguralo raznovrstan skup podataka za treniranje modela. Letovi su trajali po otprilike 15 minuta, a video materijali su snimani na planinskim vrhovima Orjak (Rukavac) i Sopalj (Kostrena) (slika 11).

Osim različitih lokacija, snimanja su provedena u različito doba dana i sa različitim brojem ljudi. Ova varijabilnost u uvjetima omogućila je prikupljanje podataka u različitim scenarijima osvjetljenja, od jakog sunčevog svjetla do slabije osvijetljenih uvjeta poput hladovina. Također, različiti vremenski uvjeti, poput vedrog neba i oblaka, dodatno su obogatili skup podataka.

Nakon završetka svakog snimanja, prikupljeni podaci su pregledani i pohranjeni. Organizacija podataka bila je ključna za kasniju fazu obrade i označavanja, stoga su snimke pažljivo kategorizirani prema lokaciji i uvjetima osvjetljenja. Ovaj sustavni pristup omogućio je učinkovitu i točnu pripremu podataka za treniranje modela.



Slika 11 Lokacija izvedenih letova

3.2. Izdvajanje kadrova

Nakon što su video snimke prikupljene, sljedeći korak uključivao je izdvajanje pojedinačnih kadrova iz videa. Ovaj proces bio je ključan za transformaciju kontinuiranih video podataka u zasebne slike koje se zatim mogu pojedinačno analizirati.

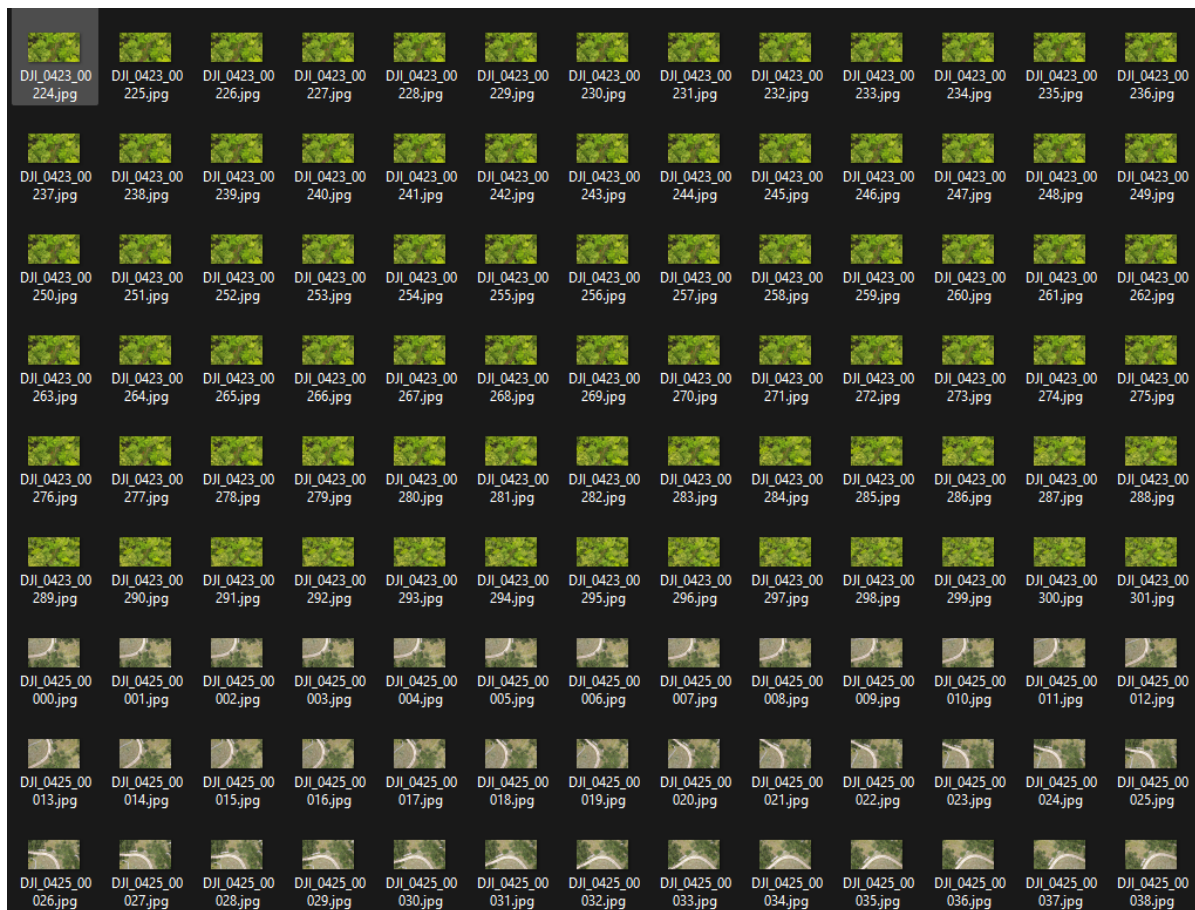
Za izdvajanje kadrova kreirana je skripta za obradu videa koja omogućuje rastavljanje videa u niz fotografija. Izdvajanje kadra u redovitim intervalima osiguralo je da skup podataka bude uravnotežen i reprezentativan te da sadrži različite trenutke unutar videozapisa. Konkretno, odlučeno je da se iz svakog videa izdvaja svaki šesti kadar. Ovaj pristup bio je odabran kako bi se smanjila količina podataka na upravljivu razinu, ali i kako bi se osigurala raznolikost podataka.

Skripta je automatski izdvajala kadrove i spremala ih u odgovarajuće direktorije, organizirane prema originalnim videozapisima. Ovakva organizacija podataka omogućio je lakše rukovanje i kasniju analizu.

Nakon izdvajanja, svaki je kadar pregledan kako bi se osigurala da samo najkvalitetniji i najrelevantniji kadrovi budu korišteni za daljnju analizu i treniranje modela. Kadrovi koji su bili zamagljeni, neprikladni ili nisu sadržavali slike planinara bili su uklonjeni iz skupa podataka.

Osim toga, izabrani kadrovi trebali su prikazivati osobe u različitim pozama, kutovima i okruženjima kako bi model mogao naučiti prepoznavati osobe u planini u različitim situacijama i kako bi model mogao postići visoku točnosti u stvarnim uvjetima (slika 12).

Proces odabira kadrova bio je temelj za formiranja baze slika i daljnje faze razvoja modela detekcije osoba pomoću dronova.



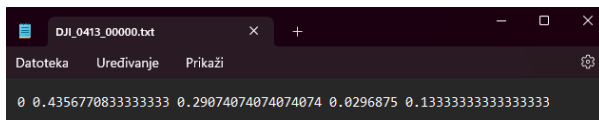
Slika 12 Izdvojeni kadrovi

3.3 Anotacija slika i priprema podataka

Svaki kadar je bilo potrebno anotirati i pravilno označiti kako bi se identificirale sve osobe na slikama. Ovaj korak zahtijevao je dosta vremena i pozornosti na detalje obzirom da točnost ovih oznaka izravno utječe na izvedbu modela detekcije. Označavanje slika uključivalo je crtanje graničnih okvira oko svake osobe u kadru i dodjeljivanje odgovarajuće oznake. Ovaj je korak neophodan za formiranje označenog skupa podataka koji će se koristiti za učenje i testiranje modela detekcije.

Kako bi se olakšao proces označavanja osoba u slici, korišten je alat LabelImg [20] koji je naširoko korišten u zajednici računalnog vida zbog svog korisničkog sučelja i učinkovitog rada. Ovaj grafički alat za označavanje slika otvorenog koda omogućuje korisnicima crtanje graničnih okvira oko objekata na slikama i spremanje komentara u .XML formatu. Alat podržava različite formate anotacija, što ga čini svestranim za različite projekte. Korištenje LabelImg pojednostavilo je proces označavanja, osiguravajući dosljedne i točne oznake. Ukoliko generirani .XML dokument nije u odgovarajućem formatu koji model može koristiti, potrebno ga je transformirati u .TXT dokument. koji YOLO može koristiti za svoje treniranje. Proces transformacije koordinata u one koje YOLO može koristiti sastoji se od nekoliko koraka:

1. Parsiranje XML datoteka: Svaka .XML datoteka sadržavala je informacije o oznakama, uključujući koordinate graničnih okvira i oznaku klase „person“. Korištenjem Python skripte, te su informacije izdvojene iz .XML datoteka (slika 14).
2. Konverzija koordinata: Koordinate graničnih okvira u .XML datotekama bile su navedene u pikselima, dok YOLO zahtijeva normalizirane koordinate (vrijednosti između 0 i 1). Konverzija je obavljena tako da su koordinate podijeljene s ukupnom širinom i visinom slike.
3. Formatiranje u .TXT datoteke (slika 13): Nakon normalizacije, podaci su formatirani u .TXT datoteku u formatu koji YOLO može koristiti. Svaka linija u .TXT datoteci sadržavala je oznaku klase, te normalizirane koordinate središta graničnog okvira, širinu i visinu.



Slika 13 Primjer txt anotacije

4. Automatizacija procesa: Python skripta je automatizirala cijeli proces konverzije, iterirajući kroz sve .XML datoteke i generirajući odgovarajuće .TXT datoteke.

YOLOv8 model prilikom treniranja koristi ove ovu .TXT datoteku sa točno označenim pozicijama osobe na slikama.

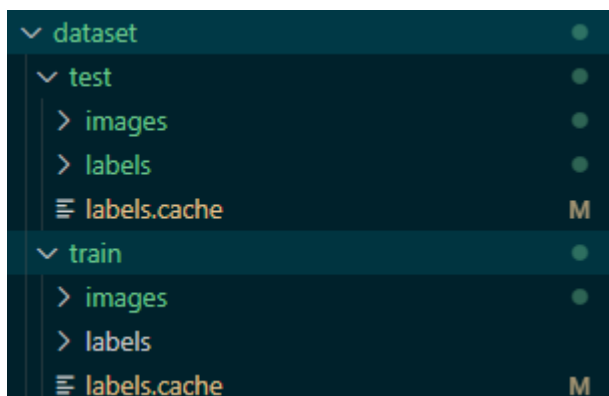


Slika 14 Primjer anotirane slike

3.4 Podjela podatka na skup za treniranje i testiranje

Iz snimljenih videozapisa generiran je skup podataka koji se sastoji od 4235 pod nazivom „gorskiPutolovci“. Označene slike i njihove odgovarajuće anotacije uparene su i podijeljene u dva skupa: skup za treniranje i skup za testiranje (slika 14). Omjer podataka bio je kao i obično 80% podataka bilo je u skupu za treniranje dakle 3388 slika, dok je preostalih 20% bilo rezervirano za testiranje 847 slika.

Proces dijeljenja podataka učinjen je nasumično kako bi se osigurala reprezentativna distribucija slika u oba skupa kako bi se izbjegla pristranost u podacima i osiguralo da oba skupa imaju sličnu varijabilnost i da sadrže slike iz oba okruženja i sa raznovrsnim uvjetima osvjetljenja. Na taj način, model je mogao naučiti značajke iz različitih situacija tijekom treniranja i biti učinkovitiji prilikom testiranja na nevidljivim podacima.



Slika 15 Generirana struktura

3.5 Treniranje modela

Nakon uspješne pripreme i podjele podataka, sljedeći korak bio je treniranje modela za detekciju osoba. Cilj ovog koraka bio je na temelju prikupljenih i označenih podataka naučiti modele da mogu pouzdano i točno prepoznavati osobe u različitim scenarijima.

3.5.1. YOLOv8

YOLOv8 dolazi u pet razina: nano, small, medium, large i extra large. Nano model je najmanji i najbrži, te zahtijeva minimalne resurse za treniranje dok verzije small, medium, large i extra large modeli postupno zahtijevaju sve više računalnih resursa i sve više vremena potrebnog za treniranje, ali nude i veću točnost u detekciji zbog složenije arhitekture i većeg broja parametara. Na primjer, treniranje nano modela traje oko 1:33 minute po epohi, dok small modelu treba oko 30 minuta po epohi. Medium model zahtijeva više od 1 sat i 10 minuta po epohi.

Zbog tehničkih ograničenja i opreme s kojom smo raspolagali, zadatak je bio eksperiment je proveden sa nano i small YOLOv8 modelima. Modeli su trenirani u 50 epoha.

Proces treniranja i finog podešavanja YOLOv8 modela uključivao je optimiziranje parametara modela na temelju označenih slika u skupu za učenje kako bi se pogreška detekcije svela na najmanju moguću mjeru.

Tijekom treniranja, model je kontinuirano evaluiran na skupu za validaciju kako bi se pratila njegova izvedba. Parametri modela su iterativno prilagođavani na temelju rezultata evaluacije kako bi se postigla optimalna točnost. Na kraju treniranja, YOLOv8 model je postigao visoku preciznost u detekciji osoba, što je potvrđeno na testnom skupu podataka.

Skripta za treniranje yolo modela

```
from ultralytics import YOLO

def train_yolo():
    # Load the YOLOv8 model
    model = YOLO('yolov8s.pt') # Using the small model

    # Train the model
    model.train(data='dataset.yaml', epochs=50, imgsz=640, batch=16,
name='yolo_v8_drone_people_4_small')

    # Validate the model
    results = model.val()
    print(results)

    # Save the model
    model.save('best_model_4_small.pt')

    return model

if __name__ == "__main__":
    trained_model = train_yolo()
```

Tijekom treniranja YOLO modela, skripta automatski generira direktorij pod nazivom "runs/detect". Ovaj direktorij služi kao centralno mjesto za pohranu svih rezultata treniranja

modela, omogućujući jednostavan pristup i analizu podataka. U "runs/detect" direktorij se pohranjuju različiti artefakti treniranja, uključujući:

- Model Checkpoints: Ovi kontrolni točke omogućuju nastavak treniranja od određene točke, čime se štedi vrijeme i računalni resursi u slučaju prekida treniranja.
- Grafikoni Performansi: Vizualni prikazi kao što su gubici treniranja (loss), preciznost i odziv tijekom epoha, omogućujući praćenje napretka modela i identifikaciju potencijalnih problema poput prekomjernog učenja (overfitting).
- Izlazne Slike: Slike iz testnog skupa s označenim detekcijama objekata, što omogućuje vizualnu provjeru točnosti modela i identifikaciju lažno pozitivnih i lažno negativnih rezultata (slika 15).
- Log Datoteke: Detaljni zapisi treniranja koji uključuju informacije o hiperparametrima, metrike po epohama i druge relevantne podatke, olakšavajući dubinsku analizu performansi modela.

Ova automatska organizacija podataka u "runs/detect" direktorij ne samo da olakšava evaluaciju i usporedbu različitih modela, već također omogućuje reproducibilnost i transparentnost eksperimenta.



Slika 16 Pravilno označene fotografije korištene za treniranje YOLO modela

3.5.2. CNN

Paralelno s podešavanjem YOLOv8 modela, definiran je i vlastiti CNN model za detekciju osoba. CNN arhitektura sadrži ukupno 10 slojeva, od kojih su 3 konvolucijska sloja, 3 sloja za udruživanje (pooling), 1 sloj za izravnavanje (flatten), 2 potpuno povezana sloja i 3 sloja za ispuštanje (dropout). Model ima jedan ulazni sloj dimenzija 128x128 piksela s 3 kanala (RGB). Izlazni sloj modela ima 4 izlaza koji predstavljaju koordinate središta graničnog okvira (x_center , y_center) i njegove dimenzije (širina, visina). Kao aktivacijske funkcije korišteni su ReLU za konvolucijske i potpuno povezane slojeve, dok je izlazni sloj bez aktivacijske funkcije.

Proces treniranja uključivao je slične korake kao kod YOLOv8 modela, uključujući tehnike povećanja podataka i regularizacije za poboljšanje robusnosti i točnosti modela. CNN model je treniran pomoću algoritama za optimizaciju Adam. Optimizacijski algoritam Adam omogućava brže konvergiranje modela i bolju prilagodbu na podatke.

Arhitektura CNN modela iterativno je usavršavana na temelju rezultata evaluacije na validacijskom skupu podataka. Korištenjem slojeva za udruživanje, smanjena je složenost modela, dok su konvolucijski slojevi omogućili učenje značajki iz slika na različitim razinama apstrakcije.

Arhitektura CNN modela korištena u ovom radu je sljedeća:

```
model = Sequential([
    Conv2D(32, (3, 3), activation='relu', input_shape=input_shape),
    BatchNormalization(),
    MaxPooling2D(pool_size=(2, 2)),
    Dropout(0.25),

    Conv2D(64, (3, 3), activation='relu'),
    BatchNormalization(),
    MaxPooling2D(pool_size=(2, 2)),
    Dropout(0.25),

    Conv2D(128, (3, 3), activation='relu'),
    BatchNormalization(),
    MaxPooling2D(pool_size=(2, 2)),
    Dropout(0.25),

    Flatten(),
    Dense(256, activation='relu'),
    Dropout(0.5),
    Dense(4) # For bounding box coordinates: x_center, y_center, width, height
])
```

Arhitektura CNN mreže opisana je u nastavku:

1. **Conv2D (32 filtera dim. 3x3):** Prvi konvolucijski sloj koristi 32 filtera veličine 3x3 za ekstrakciju značajki iz ulaznih slika, uz ReLU aktivacijsku funkciju za uvođenje nelinearnosti sa slojem BatchNormalizacije koja normalizira vrijednosti značajka u sloju kako bi se ubrzao proces učenja i stabilizirala mreža te operacijom MaxPooling2D (2x2 pool size) kojom se smanjuju dimenzije značajki, zadržavajući najvažnije informacije u okolini 2x2. Kako bi se spriječilo prekomjerno učenje (overfitting) tijekom treniranja primijenjen je Dropout (0.25), regularizacijska tehnika koja nasumično isključuje 25% neurona u sloju. To znači da je u svakom trening prolazu 25% neurona isključeno kako bi se spriječilo prekomjerno prilagođavanje modela trenažnim podacima, što pomaže modelu da generalizira bolje na neviđene podatke.
2. **Conv2D (64 filtera dim. 3x3):** Drugi konvolucijski sloj s 64 filtera za složenije značajke uz ReLU aktivacijsku funkciju, zatim sloj BatchNormalizacije, MaxPooling2D (2x2 pool size) za isticanje najjačeg signala i smanjivanja dimezije te regularizacija sa Dropout (0.25).
3. **Conv2D (128 filtera dim. 3x3):** Treći konvolucijski sloj s 128 filtera za još složenije značajke, uz ReLU aktivacijsku funkciju, zatim sloj BatchNormalizacije, te operacija MaxPooling2D (2x2 pool size) i regularizacija Dropout (0.25).

4. Flatten: Pretvara dvodimenzionalne značajke u jednodimenzionalni vektor i koristi se kao ulaz u potpuno povezane slojeve.
5. Dense (256 neurons, ReLU activation): Potpuno povezani sloj s 256 neurona za učenje visokorazinskih značajki.
Dropout (0.5): Snažna regularizacija za smanjenje prekomjernog učenja.
6. Dense (4 neurons): Izlazni sloj s 4 neurona za predikciju koordinata bounding boxa koji sadrži osobu (x_center, y_center, width, height).

3.6. Vizualizacija i validacija rada modela

Nakon što su svi modeli uspješno istrenirani, sljedeći korak bio je vizualno prikazati rad modela i napraviti usporedba njihovih performansi. Ovaj zadatak bio je ključan za razumijevanje načina na koji svaki model detektira osobe i za identifikaciju potencijalnih područja za poboljšanje. Za ovaj korak korištene su posebno izrađene skripte koje su testirale modele na skupu testnih slika. Skripte su bile osmišljene tako da uzimaju svaki model, primjenjuju ga na testne slike i na temelju rezultata generiraju i pohranjuju matrice konfuzije. Matrice konfuzije su korištene za kvantitativnu analizu performansi svakog modela, omogućujući jasan uvid u točnost, preciznost, odziv i F1 mjeru.

Vizualni prikaz rezultata pružio je detaljan uvid u sposobnosti svakog modela da prepozna osobe iz različitih kutova i pod različitim uvjetima osvjetljenja. Generirane matrice konfuzije bile su ključne za evaluaciju modela, jer su omogućile jednostavno prepoznavanje pravih pozitivnih, lažnih pozitivnih, pravih negativnih i lažnih negativnih rezultata. Na temelju matrica konfuzije i evaluacije, bilo je moguće zaključiti koji je model najpogodniji za primjenu u detekciji osoba snimljenih dronom te identificirati mogućnosti za daljnje poboljšanje i optimizaciju modela.

4. Rezultati

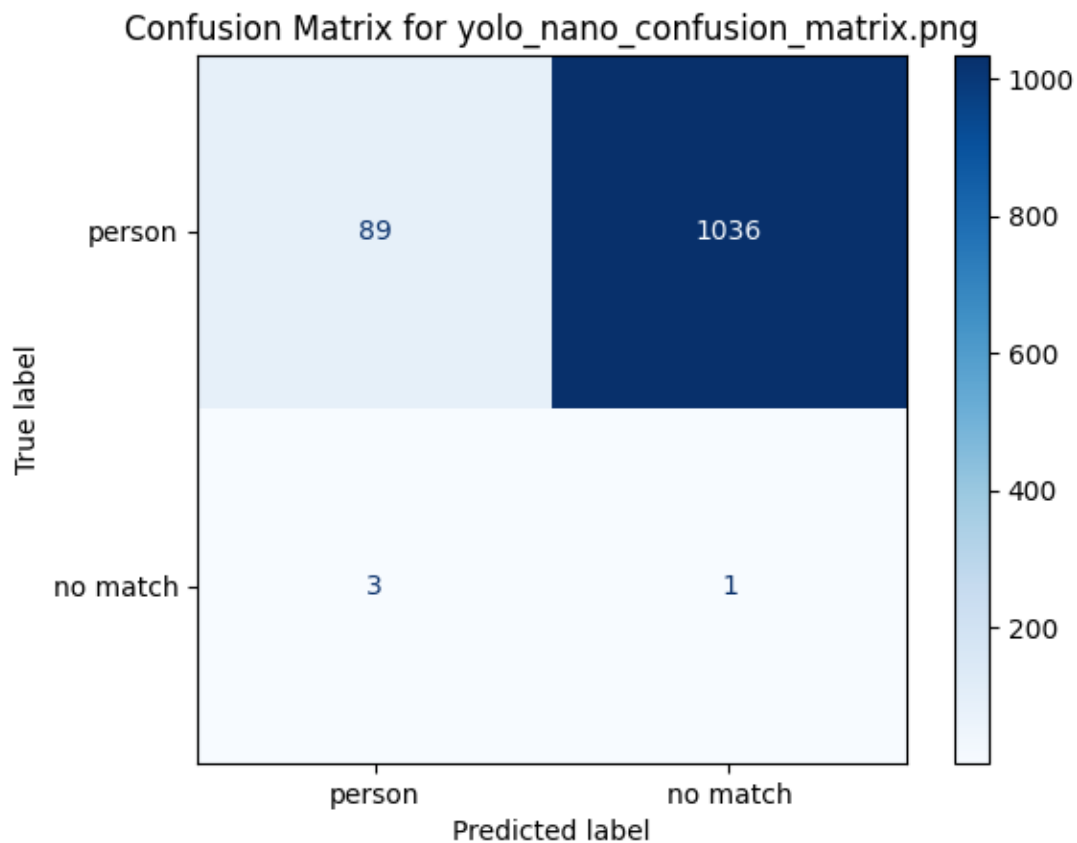
Detaljno su mjerene performanse pet različitih modela za detekciju osoba: prilagođenog CNN modela, originalnog YOLOv8 Small (YOLOv8s), originalnog YOLOv8 Nano (YOLOv8n), te njihove varijacije modela koji su fino podešeni s podacima prikupljenim dronom. Svaki model je evaluiran koristeći matrice konfuzije kako bi se kvantificirala njihova točnost, preciznost, odziv i F1 mjera.

Tablica 2 Prikaz evaluacijskih metrika korištenih modela

Model	Točnost	Preciznost	Odziv	F1 Mjera	IoU
YOLOv8n	7,97%	96,7%	7,91%	14,5%	30,47%
YOLOv8s	23,27%	98,5%	23,38%	37,67%	25,59%
YOLOv8n (Gorski putolovci)	7,97%	7,91%	96,74%	14,63%	30,47%
YOLOv8s (Gorski Putolovci)	97,63%	98,53%	99,02%	98,80%	53,38%
CNN	38,53%	100%	38,5%	55,6%	0%

Posebna pažnja posvećena je modelima treniranima sa slikama iz skupa GorskiPutolovci snimljenih dronom, zbog specifičnih izazova koje ti podaci predstavljaju, uključujući različite kutove snimanja, promjenjive uvjete osvjetljenja i složenost pozadina. Rezultati su pokazali značajne razlike u performansama među modelima.

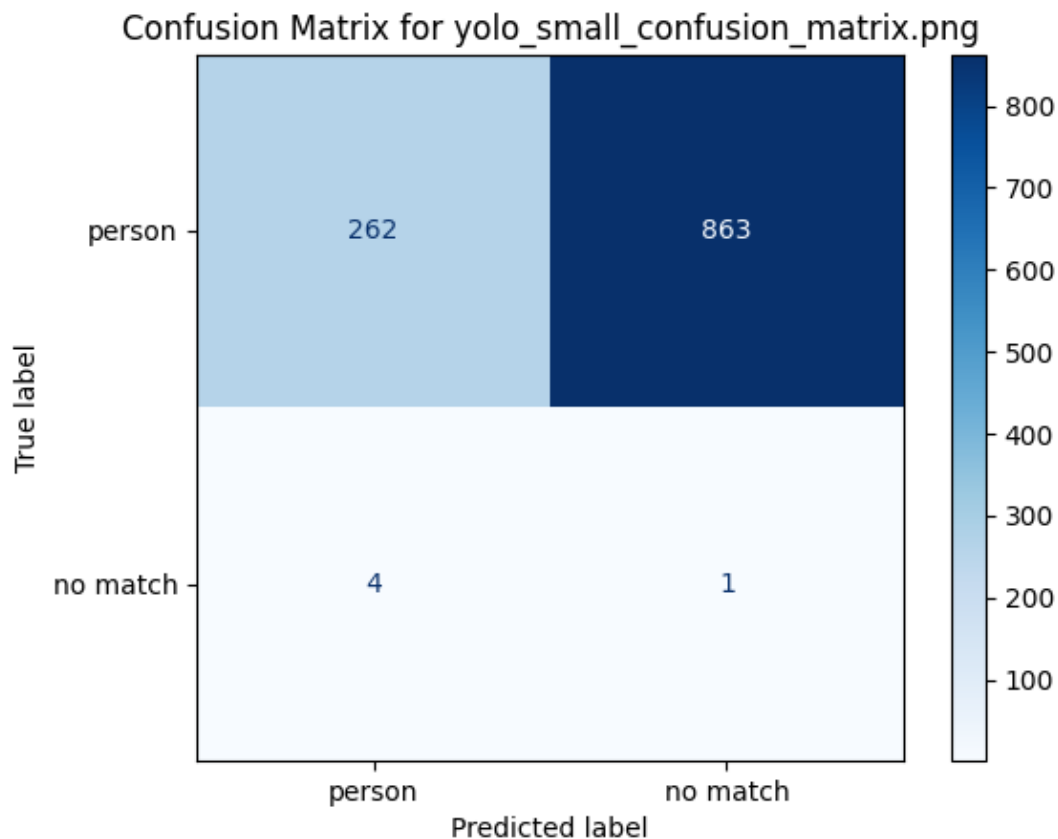
4.1. YOLOv8 Nano



Slika 17 Matrica konfuzije YOLOv8 nano modela

Matrica konfuzije (slika 16) prikazuje performanse originalnog YOLOv8 Nano modela koji nije dodatno učen na slikama iz skupa podataka GorskiPutolovci. Ovaj model ima najnižu točnost među testiranim modelima, sa samo 7.97% ispravno detektiranih primjera. Preciznost modela je vrlo visoka, sa 96,7% predviđenih osoba koje su stvarno osobe. Unatoč tome, model ima nizak odziv od 7,91%, što ukazuje da nije uspio prepoznati većinu stvarnih osoba. F1 mjera modela iznosi 14.5%, što odražava mali broj osoba koje su zapravo detektirane (nizak odziv) iako je preciznost detektiranih osoba visoka.

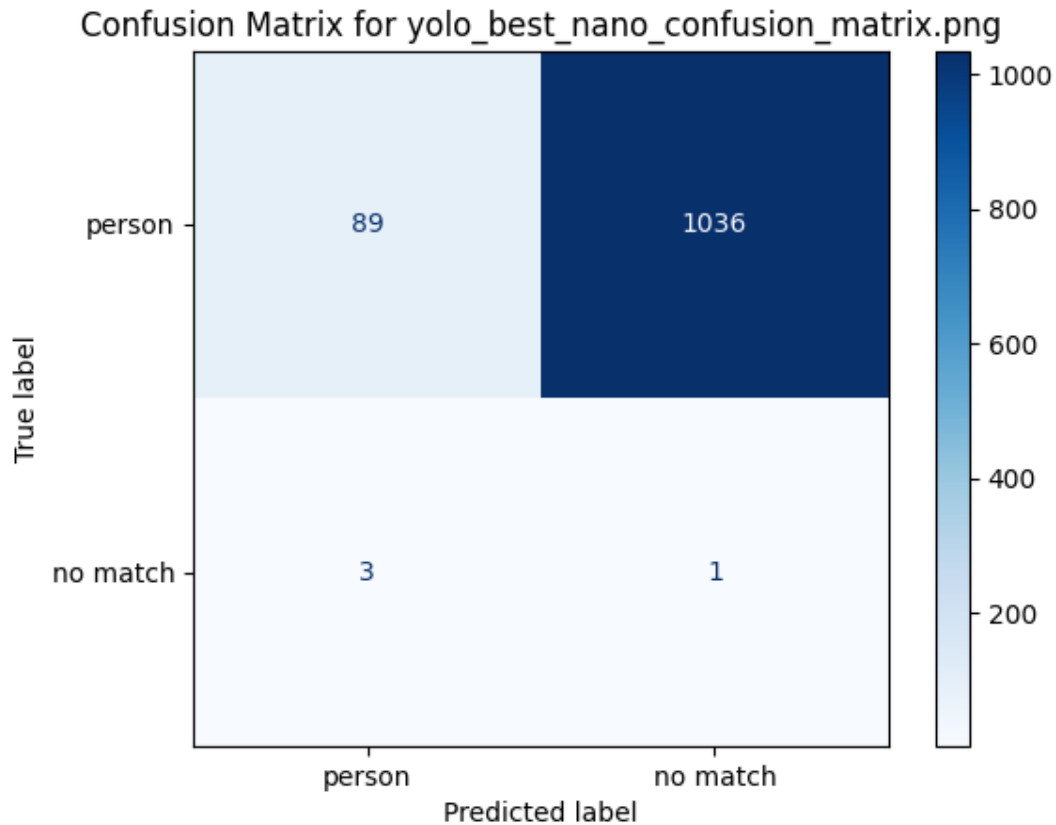
4.2 YOLOv8 Small



Slika 18 Matrica konfuzije YOLOv8 small modela

Matrica konfuzije (slika 17) prikazuje performanse originalnog YOLOv8 Small modela koji nije dodatno učen na slikama iz skupa podataka GorskiPutolovci. Ovaj model ima vrlo nisku točnost među testiranim modelima, sa samo 23,27% ispravno detektiranih primjera. Preciznost modela je vrlo visoka, sa 98,5% predviđenih osoba koje su stvarno osobe. Unatoč tome, model ima nizak odziv od 23,38%, što ukazuje da nije uspio prepoznati većinu stvarnih osoba. F1 mjera modela iznosi 37,67%, što odražava mali broj osoba koje su zapravo detektirane (nizak odziv) iako je preciznost detektiranih osoba visoka.

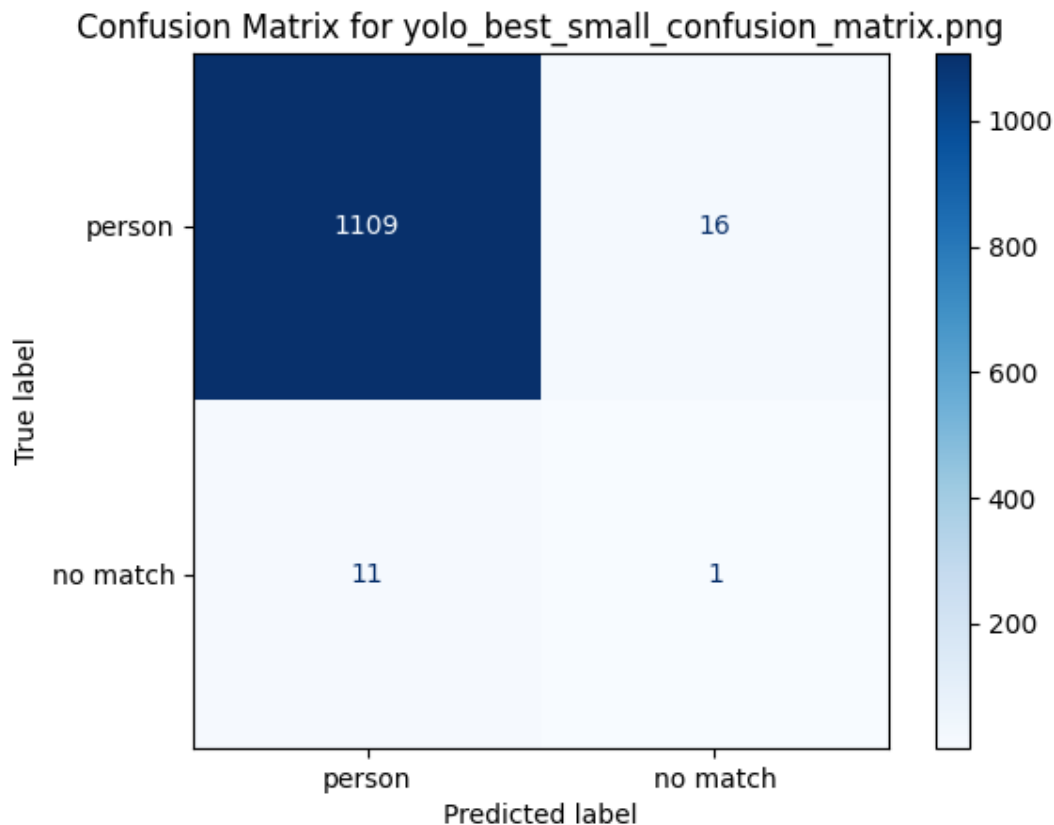
4.3. Personalizirani YOLOv8 Nano



Slika 19 Matrica konfuzije personaliziranog YOLOv8 nano modela

Matrica konfuzije (slika 18) prikazuje performanse personaliziranog YOLOv8 Nano modela koji je dodatno učen na slikama iz skupa podataka GorskiPutolovci. Ovaj model ima vrlo nisku točnost među testiranim modelima, od samo 7.97% ispravno detektiranih primjera. Preciznost modela je vrlo visoka, sa 96,74% predviđenih osoba koje su stvarno osobe. Unatoč tome, model ima nizak odziv od 7,91%, što ukazuje da nije uspio prepoznati većinu stvarnih osoba. F1 mjera modela iznosi 14.63%, što odražava mali broj osoba koje su zapravo detektirane (nizak odziv) iako je preciznost detektiranih osoba visoka.

4.4. Personalizirani YOLOv8 Small



Slika 20 Matrica konfuzije personaliziranog YOLOv8 small modela

Matrica konfuzije (slika 19) prikazuje performanse personaliziranog YOLOv8 Small modela koji je dodatno učen na slikama iz skupa podataka GorskiPutolovci. Ovaj model ima daleko najvišu točnost među testiranim modelima, 95,58% ispravno detektiranih primjera. Preciznost modela je vrlo visoka, 98,58% predviđenih osoba koje su stvarno osobe. Unatoč tome, model također ima vrlo visok odziv od 99,02%, što ukazuje da je vrlo uspješno prepoznao gotov sve stvarne osobe. F1 mjera modela iznosi 98,80%, što odražava velik broj osoba koje su zapravo detektirane (visok odziv) dok je istovremeno preciznost detektiranih osoba vrlo visoka (slika 20, slika 21).

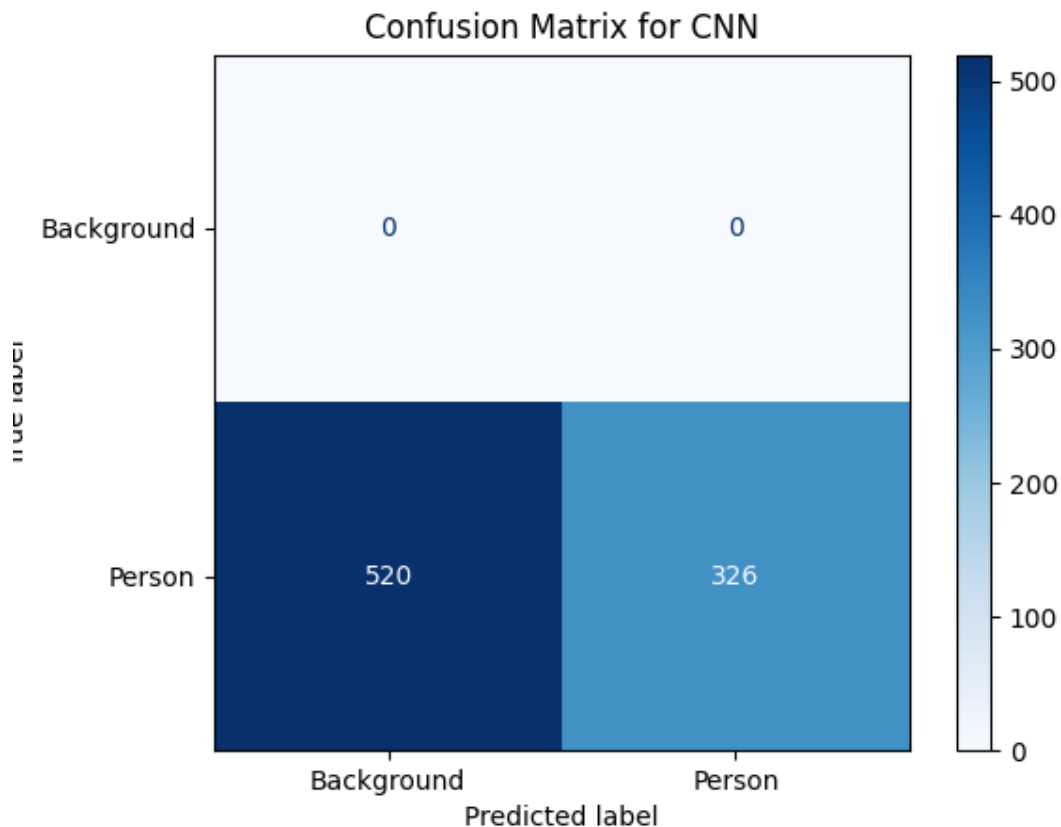


Slika 21 Primjer validacije najboljeg modela. Validacijski batch 1 - anotacije



Slika 22 Primjer validacije najboljeg modela. Validacijski batch 1 - predviđanja

4.5. Vlastiti CNN



Slika 23 Matrica konfuzije CNN modela

Matrica konfuzije (slika 22) prikazuje performanse vlastitog CNN modela koji je učen na slikama iz skupa podataka GorskiPutolovci. Ovaj model ima vrlo nisku točnost među testiranim modelima, sa samo 38,53% ispravno detektiranih primjera. Preciznost modela je vrlo visoka, sa 100% predviđenih osoba koje su stvarno osobe. Unatoč tome, model ima nizak odziv od 38,50%, što ukazuje da nije uspio prepoznati većinu stvarnih osoba. F1 mjera modela iznosi 55,6%, što odražava mali broj osoba koje su zapravo detektirane (nizak odziv) iako je preciznost detektiranih osoba visoka.

4.6. Analiza rezultata

Analizom rezultata evaluacije modela jasno je da je YOLOv8 Best Nano model postigao najbolje rezultate u pogledu ukupne preciznosti, točnosti i odziva, u usporedbi s ostalim testiranim modelima. Ovaj model je nadmašio sve ostale, posebno kada se uspoređuje s netreniranim verzijama YOLOv8 Small i YOLOv8 Nano modela. Model je postigao gotovo savršene rezultate u detekciji osoba, s vrlo visokim vrijednostima svih ključnih metričkih parametara. Točnost od 97.63% znači da je model ispravno klasificirao gotovo sve primjere. Preciznost i odziv su također izuzetno visoki, što ukazuje na minimalan broj lažno pozitivnih i lažno negativnih rezultata. Kada usporedimo prilagođene modele sa standardnim vidljivo je da su modeli YOLOv8 Small i YOLOv8 Nano pokazali su značajno slabije performanse.

Rezultati jasno pokazuju koliko je velika razlika u performansama između netreniranih i prilagođenih modela. Prilagođeni YOLOv8 Nano model ima više nego četiri puta veću točnost u odnosu na YOLOv8 Small i više nego deset puta veću točnost u odnosu na YOLOv8 Nano. Preciznost i F1 mjera također pokazuju značajna poboljšanja, što ukazuje na daleko bolju sposobnost modela da točno prepozna i klasificira stvarne osobe.

Iako su rezultati YOLOv8 Best Nano modela izuzetni, postoji mogućnost da bi korištenje većeg modela, kao što je YOLOv8 Medium, moglo dodatno poboljšati performanse. Veći modeli imaju više parametara i složeniju arhitekturu, što omogućuje učenje finijih detalja i značajki iz podataka. Nažalost, zbog ograničenja dostupnih GPU resursa, treniranje YOLOv8 Medium modela nije bilo izvedivo. Također, korištenje tehnika poput povećanja podataka i finog podešavanja hiperparametara moglo bi dodatno optimizirati performanse većeg modela.

2.4. Usporedba YOLOv8 i vlastitog CNN-a

U ovom radu provedena je usporedna analiza između četiri YOLOv8 modela i jednog CNN modela kako bi se ocijenila njihova izvedba u detekciji osoba pomoću dronova. Kriteriji ocjenjivanja uključivali su točnost detekcije, brzinu obrade i robusnost u različitim uvjetima okoline.

Istrenirani CNN model, unatoč prilagođenom dizajnu, pokazao je sporije brzine obrade u usporedbi s YOLOv8. Međutim, ovaj model je pokazao malo veću točnost u kontroliranim okruženjima sa statičnim pozadinama.

2.5. Prednosti

Detekcija objekata korištenjem naprednih modela kao što su YOLOv8 i CNN nudi značajne prednosti u raznim primjenama. Ove tehnologije poboljšavaju učinkovitost i točnost detekcije objekata u stvarnom vremenu, čineći ih prikladnima za dinamična i složena okruženja.

2.5.1. YOLOv8

Brzina - YOLOv8 je dizajniran za obradu u stvarnom vremenu, što ga čini vrlo prikladnim za aplikacije gdje je brzina kritična, kao što su nadzor, autonomna navigacija i praćenje u stvarnom vremenu.

Preciznost - Sa svojom naprednom arhitekturom, YOLOv8 postiže visoku točnost detekcije, posebno u detekciji malih i preklapajućih objekata, što ga čini učinkovitim u prepunim ili složenim okruženjima.

Unificirana arhitektura - YOLOv8 način detekcije pojednostavljuje implementaciju i smanjuje računalne zahtjeve u usporedbi s kompleksnijim detektorima. Ovaj jedinstveni pristup dovodi do niže latencije i bržeg vremena zaključivanja.

Skalabilnost - YOLOv8 se može povećati ili smanjiti ovisno o dostupnim računalnim resursima, što ga čini svestranim za implementaciju i na vrhunskim poslužiteljima i na uređajima s ograničenim resursima kao što su dronovi.

2.5.2. CNN

Fleksibilnost: CNN-ovi se mogu prilagoditi specifičnim zadacima i vrlo su svestrani za razne aplikacije za obradu slike. Ova fleksibilnost omogućuje prilagođene arhitekture koje se mogu optimizirati za posebne slučajeve upotrebe, kao što je otkrivanje osoba.

Učenje značajki: CNN-ovi automatski uče hijerarhijske značajke iz podataka, što smanjuje potrebu za ručnim inženjeringom značajki i omogućuje mreži da adaptivno poboljša svoje performanse kako se pruža više podataka.

Prijenos učenja: Unaprijed obučeni modeli CNN-a mogu se fino prilagoditi novim zadacima s relativno manjim skupovima podataka, koristeći prethodno naučene značajke za postizanje dobrih performansi s manje podataka za treniranje.

2.6. Mane

Unatoč svojim prednostima, i YOLOv8 i CNN-ovi imaju određena ograničenja koja se moraju riješiti kako bi se povećala njihova učinkovitost.

2.6.1. YOLOv8

Složenost: Arhitektura i proces treniranja YOLOv8 su složeniji u usporedbi s jednostavnijim modelima, zahtijevajući značajne računalne resurse za treniranje i fino podešavanje. Ova složenost može biti prepreka za korisnike s ograničenim pristupom visokoučinkovitoj računalnoj infrastrukturi.

Generalizacija: YOLOv8 može zahtijevati opsežno povećanje podataka i pažljivo podešavanje za dobru generalizaciju u različitim okruženjima i kategorijama objekata. Bez odgovarajućeg treniranja, model može imati loše rezultate u novim ili nepredviđenim scenarijima.

Zahtijeva resurse: Dok je YOLOv8 optimiziran za brzinu, njegova faza treniranja je intenzivna za resurse, zahtijeva veliku računalnu snagu i velike skupove podataka kako bi se postigla optimalna izvedba.

2.6.2. CNN

Računalna složenost: treniranje dubokih CNN-ova može biti računalno skupa i dugotrajna, osobito s velikim skupovima podataka. Ovaj zahtjev može biti ograničenje za aplikacije koje zahtijevaju brzu implementaciju ili ograničene računalne resurse.

Prekomjerno prilagođavanje: Bez pravilne regulacije i povećanja podataka, CNN-ovi se mogu previše prilagoditi podacima o obuci, što dovodi do loše generalizacije nevidljivih podataka. Prekomjerno opremanje zahtijeva pažljivo upravljanje pomoću tehnika kao što su ispadanje, reguliranje težine i unakrsna provjera.

Složenost arhitekture: Dizajniranje optimalne CNN arhitekture može biti složeno i zahtijeva stručnost u dubinskom učenju. Pogrešni odabiri dizajna mogu dovesti do suboptimalnih performansi i povećanih troškova računanja.

5. Zaključak

Ovaj diplomski rad istraživao je upotrebu bespilotnih letjelica za otkrivanje osoba, fokusirajući se na razvoj i usporedbu tri modela: prilagođene konvolucijske neuronske mreže (CNN), prilagođenog modela YOLOv8 i zadanog modela YOLOv8. Rad je imao za cilj procijeniti djelotvornost i učinkovitost ovih modela u detekciji osoba u stvarnom vremenu iz snimke dronom, što je zadatak od ključne važnosti za aplikacije kao što su potraga i spašavanje, nadzor i upravljanje katastrofama.

Projekt je započeo prikupljanjem neobrađenih video snimaka pomoću drona, nakon čega je uslijedilo izdvajanje pojedinačnih kadrova iz videa. Svaki okvir je pažljivo označen kako bi se identificirale osobe, a označavanje je izvršeno pomoću alata LabelImg. Ovaj je alat spremio komentare u .XML formatu, koji su zatim pretvoreni u .TXT format prikladan za YOLOv8. Podaci su naknadno podijeljeni u skupove za obuku i testiranje kako bi se osigurala robusna evaluacija modela.

Prilagođeni CNN i YOLOv8 modeli obučeni su pomoću pripremljenog skupa podataka. Model YOLOv8, poznat po svojoj mogućnosti obrade u stvarnom vremenu, dodatno je fino podešen kako bi poboljšao svoje performanse. Modeli su procijenjeni na temelju različitih metrika, uključujući točnost, preciznost, prisjećanje i vrijeme obrade. Usporedna analiza pokazala je da je prilagođeni model YOLOv8 nadmašio ostale modele u pogledu točnosti detekcije i brzine obrade.

Najznačajniji zaključak studije je da model YOLOv8, posebno kada je fino podešen, nudi vrhunske performanse za otkrivanje osoba u stvarnom vremenu pomoću dronova. Njegova visoka točnost i brzina čine ga prikladnim za aplikacije koje zahtijevaju trenutačnu reakciju i pouzdanu detekciju, kao što su hitne operacije spašavanja i nadzor sigurnosti. Prilagođeni CNN, iako također učinkovit, zaostajao je za YOLOv8 u učinkovitosti obrade, naglašavajući važnost arhitekture modela u postizanju optimalnih rezultata.

Nadalje, proces pretvaranja .XML komentara u .TXT format i automatizacija radnog tijeka pripreme podataka pokazao se ključnim za uspješnu implementaciju YOLOv8 modela. Ovaj je korak osigurao da se model može učinkovito trenirati s točno označenim podacima, naglašavajući važnost pravilnog rukovanja podacima u projektima strojnog učenja.

Preporuke za daljnji rad uključuju proširenje skupa podataka kako bi se uključila raznovrsnija okruženja i uvjeti, što bi poboljšalo robusnost modela i mogućnost generalizacije. Osim toga, istraživanje drugih naprednih modela detekcije objekata i hibridnih pristupa moglo bi dovesti do daljnjih poboljšanja u performansama detekcije. Buduća bi se istraživanja također mogla usredotočiti na korištenje fotografskih termalnih kamera i kombinaciju normalnih slika i termalnih slika.

Ukratko, ovaj rad pridonosi napretku detekcije osoba pomoću drona demonstrirajući učinkovitost modela YOLOv8. Uvidi dobiveni ovim istraživanjem pružaju čvrstu osnovu za buduće studije i praktične primjene, utirući put pouzdanijim i učinkovitijim sustavima detekcije u kritičnim scenarijima u stvarnom vremenu. Ovdje navedene metodologije i nalazi mogu poslužiti kao vodič za slične projekte, osiguravajući da se razvoj i implementacija naprednih modela detekcije nastavi razvijati i poboljšavati

Literatura

- [1] Ultralytics, »Ultralytics,« [Mrežno]. Available: <https://docs.ultralytics.com/>. [Pokušaj pristupa 20 06 2024].
- [2] R. Gandhi, »R-CNN, Fast R-CNN, Faster R-CNN, YOLO — Object Detection Algorithms,« 9 7 2018. [Mrežno]. Available: <https://towardsdatascience.com/r-cnn-fast-r-cnn-faster-r-cnn-yolo-object-detection-algorithms-36d53571365e>. [Pokušaj pristupa 20 06 2024].
- [3] Technical, »Object detection,« [Mrežno]. Available: <https://visionplatform.ai/object-detection/>. [Pokušaj pristupa 20 06 2024].
- [4] »IBM Object detection,« 3 1 2021. [Mrežno]. Available: <https://www.ibm.com/topics/object-detection>. [Pokušaj pristupa 20 06 2024].
- [5] U. S. Hansen, »Object Detection: Models, Use Cases, Examples,« 14 04 2023. [Mrežno]. Available: <https://encord.com/blog/object-detection/>. [Pokušaj pristupa 20 06 2024].
- [6] G. Boesch, »Object Detection in 2024: The Definitive Guide,« 4 1 2024. [Mrežno]. Available: <https://viso.ai/deep-learning/object-detection/>. [Pokušaj pristupa 20 06 2024].
- [7] »History of Computer Vision,« [Mrežno]. Available: <https://letsdatascience.com/learn/history/history-of-computer-vision/>. [Pokušaj pristupa 20 06 2024].
- [8] M. Švero, »Houghova transformacija,« 1 5 2009. [Mrežno]. Available: <https://www.zemris.fer.hr/~ssegvic/project/pubs/sverko09sem2.pdf>. [Pokušaj pristupa 20 06 2024].
- [9] M. Kennerley, »A Comparison of SIFT, SURF and ORB on OpenCV,« 21 5 2021. [Mrežno]. Available: <https://mikhail-kennerley.medium.com/a-comparison-of-sift-surf-and-orb-on-opencv-59119b9ec3d0>. [Pokušaj pristupa 20 06 2024].
- [10] S. Bangar, »AlexNet Architecture Explained,« 24 6 2022. [Mrežno]. Available: <https://medium.com/@siddheshb008/alexnet-architecture-explained-b6240c528bd5>. [Pokušaj pristupa 20 06 2024].
- [11] »YOLO Object Detection Explained,« 1 09 2022. [Mrežno]. Available: <https://www.datacamp.com/blog/yolo-object-detection-explained>. [Pokušaj pristupa 20 06 2024].
- [12] B. Kumar, »Convolutional Neural Networks: A Brief History of their Evolution,« 31 8 2021. [Mrežno]. Available: <https://medium.com/appyhigh-technology-blog/convolutional-neural-networks-a-brief-history-of-their-evolution-ee3405568597>. [Pokušaj pristupa 20 06 2024].

- [13] »LeNet-5 Architecture,« 24 5 2024. [Mrežno]. Available: <https://www.geeksforgeeks.org/lenet-5-architecture/>. [Pokušaj pristupa 20 06 2024].
- [14] S. Das, »CNN Architectures: LeNet, AlexNet, VGG, GoogLeNet, ResNet and more...,« Medium, 16 12 2017. [Mrežno]. Available: <https://medium.com/analytics-vidhya/cnns-architectures-lenet-alexnet-vgg-googlenet-resnet-and-more-666091488df5>. [Pokušaj pristupa 20 06 2024].
- [15] N. Adaloglou, »Best deep CNN architectures and their principles: from AlexNet to EfficientNet,« theaisummer, 21 1 2021. [Mrežno]. Available: <https://theaisummer.com/cnn-architectures/>. [Pokušaj pristupa 20 06 2024].
- [16] D. Team, »dec.ai,« 5 06 2023. [Mrežno]. Available: <https://dec.ai/blog/history-yolo-object-detection-models-from-yolov1-yolov8/>. [Pokušaj pristupa 20 06 2024].
- [17] S. A. Khowaja, »Brief Timeline of YOLO models from v1 to v8,« Medium, 1 8 2023. [Mrežno]. Available: <https://sandar-ali.medium.com/brief-timeline-of-yolo-models-from-v1-to-v8-7d383140afe2>. [Pokušaj pristupa 20 06 2024].
- [18] »DJI,« [Mrežno]. Available: <https://www.dji.com/hr/support/product/mavic-air-2>. [Pokušaj pristupa 20 06 2024].
- [19] »Learn Microsoft,« 29 11 2023. [Mrežno]. Available: <https://learn.microsoft.com/en-us/windows/wsl/about>. [Pokušaj pristupa 20 06 2024].
- [20] »labelImg 1.8.6,« pypi, [Mrežno]. Available: <https://pypi.org/project/labelImg/>. [Pokušaj pristupa 20 06 2024].

Popis tablica

Tablica 1 Usporedba performansi YOLO modela

Tablica 2 Prikaz evaluacijskih metrika korištenih modela

Popis slika

Slika 1 Primjer automatske detekcije objekta

Slika 2 Operacija konvolucije

Slika 3 Max pool operacija

Slika 4 ReLU Funkcija i graf

Slika 5 Softmax funkcija i graf

Slika 6 Standardna arhitektura CNN modela

Slika 7 Standardna arhitektura YOLO modela

Slika 8 Korišteni dron

Slika 9 WSL Sučelje

Slika 10 Razlika u odabranim okolinama snimanja

Slika 11 Lokacija izvedenih letova

Slika 12 Izdvojeni kadrovi

Slika 13 Primjer txt anotacije

Slika 13 Primjer anotirane slike

Slika 14 Generirana struktura

Slika 15 Pravilno označene fotografije korištene za treniranje YOLO modela

Slika 16 Matrica konfuzije YOLOv8 nano modela

Slika 17 Matrica konfuzije YOLOv8 small modela

Slika 18 Matrica konfuzije personaliziranog YOLOv8 nano modela

Slika 19 Matrica konfuzije personaliziranog YOLOv8 small modela

Slika 20 Primjer validacije najboljeg modela. Validacijski batch 1 - anotacije

Slika 21 Primjer validacije najboljeg modela. Validacijski batch 1 - predviđanja

Slika 22 Matrica konfuzije CNN modela

Popis priloga

- Baza slika pripremljena za učenje modela za detekciju osoba „Gorski Putolovci“
- YOLOv8s (Gorski Putolovci)