



Univerzitet u Sarajevu
Elektrotehnički fakultet
Sarajevo

Elektrotehnički fakultet Univerziteta u Sarajevu

Odsjek za Računarstvo i informatiku

Predmet: Vještačka inteligencija

Tema

Prepoznavanje objekata u prirodnom okruženju

Autori:

Faris Kasapovic 19418

Mika Bjelobrk 19056

Dževad Madžak 19302

Sarajevo, akademska 2024/2025 godina

Sadržaj

Izbor teme i opis problema	4
1.1 Odabir problema iz oblasti vještačke inteligencije	4
1.2 Opis problema koji se rješava	4
1.3 Definisanje osnovnih pojmova i koristi sistema	5
Osnovni pojmovi:	5
Koristi rješenja:	5
1.4 Pregled postojećih datasetova	6
1.4.1 Military Assets Dataset (12 Classes – YOLOv8 Format)	6
1.4.2 Military Aircraft Detection Dataset	6
1.4.3 YOLO Military Object Detection Dataset	6
Pregled stanja u oblasti	7
2.1 Analiza trenutnog stanja u oblasti	7
2.2 Korištene metode vještačke inteligencije	7
2.3 Postignuti rezultati u literaturi	7
2.4 Potencijalni pravci za poboljšanje	8
Izbor, analiza i pretprocesiranje dataset-a	9
3.1 Izabrani dataset	9
3.2 Osnovni pregled dataset-a	9
Format i način preuzimanja:	9
Struktura dataset-a:	10
3.3 Broj klasa i instanci	10
Distribucija klasa (ukupno instanci: 43,336):	10
3.4 Analiza dataset-a	11
Analiza veličina slika:	11
Analiza bounding boxova:	11

3.5 Identifikovani rizici	11
3.6 Metode pretprocesiranja podataka	12
Odabir, formiranje, treniranje i testiranje modela.....	13
4.1 Izbor metode i tehnologija	13
Odabrana metoda: YOLOv8 (You Only Look Once version 8)	13
Razlozi izbora:	13
Korištene tehnologije:	13
4.2 Priprema formata podataka.....	14
4.3 Treniranje modela.....	14
4.4 Detaljan opis treniranja	15
4.5 Testiranje dobijenog modela.....	16
4.6 Korištene metrike	18
Primarne metrike:	18
Per-class metrike:.....	18
4.7 Diskusija rezultata	19
4.8 Test na nepoznatim podacima	20
Zaključak.....	22
Ključna postignuća	23
Tehnički doprinosi.....	23
Ograničenja i pravci za poboljšanje	23
Preporučeni sljedeći koraci	24

Izbor teme i opis problema

1.1 Odabir problema iz oblasti vještačke inteligencije

Odabran je problem **prepoznavanja i klasifikacije vojnih objekata na slikama snimljenim u prirodnom okruženju** pomoću savremenih metoda vještačke inteligencije. Ovaj problem predstavlja jedan od najzahtjevnijih izazova u domenu računarskog vida zbog specifičnih karakteristika vojnih scena kao što su kamuflaža, varijabilnost objekata, različiti uglovi snimanja i složena pozadina.

Savremene vojne i sigurnosne aplikacije sve više se oslanjaju na automatizovanu analizu vizuelnih podataka, posebno onih dobijenih iz satelitskih snimaka i bespilotnih letjelica (dronova). Zahvaljujući brzom razvoju vještačke inteligencije i posebno transformerskih arhitektura u oblasti računarskog vida, danas je moguće razviti robusne modele koji razumiju širi kontekst slike i mogu sa visokom preciznošću detektovati i klasifikovati čak i vrlo kompleksne vizuelne elemente.

1.2 Opis problema koji se rješava

Problem: U realnim scenarijima, snimci iz dronova ili satelita često sadrže vojnu opremu koja je teško uočljiva zbog kamuflaže i prirodnog okruženja. Potrebno je razviti sistem koji automatski:

- Identifikuje objekte (detekcija)
- Klasifikuje ih u odgovarajuće kategorije (npr. tenk, avion, helikopter)

Cilj: Primijeniti model YOLOv8 (You Only Look Once v8) i trenirati ga na skupu slika vojne opreme u prirodnim uslovima, kako bi postigao visok stepen tačnosti u realnim uslovima.

1.3 Definisanje osnovnih pojmova i koristi sistema

Osnovni pojmovi:

- **Objekat** - Predstavlja vidljivi entitet na slici koji želimo prepoznati, kao što su tenk, vojni helikopter, borbeni avion i slično
- **Detekcija objekata** - Odnosi se na identifikaciju i lokalizaciju objekata unutar slike. Rezultat detekcije su tzv. bounding boxovi koji označavaju gdje se objekat nalazi.
- **Klasifikacija** - Nakon što je objekat detektovan, dodjeljuje mu se odgovarajuća klasa (npr. tenk, helikopter, avion) na osnovu njegovih vizuelnih karakteristika
- **Inferencija** - Proces korištenja prethodno istreniranog modela na novim, neviđenim podacima. Cilj je da model tačno prepozna i klasifikuje objekte u stvarnim uslovima
- **YOLO (You Only Look Once)** - Napredna arhitektura neuronskih mreža za real-time detekciju objekata koja tretira detekciju kao regression problem

Koristi rješenja:

- **Automatizacija analize snimaka** (dronovi, sateliti)
- **Brže izviđanje i nadzor**
- **Smanjenje ljudske greške** u ručnoj analizi slika
- **Mogućnost primjene u stvarnom vremenu**
- **Povećanje efikasnosti** vojnih i sigurnosnih operacija

1.4 Pregled postojećih datasetova

1.4.1 Military Assets Dataset (12 Classes – YOLOv8 Format)

- **Izvor:** Kaggle
- **Link:** <https://www.kaggle.com/datasets/rawsi18/military-assets-dataset-12-classes-yolo8-format>
- **Opis:** Sadrži 26,315 označenih slika vojnih objekata, uključujući tenkove, helikoptere, avione i druga vozila. Anotacije su u YOLOv8 formatu, što omogućava laku integraciju sa modernim modelima za detekciju objekata.

1.4.2 Military Aircraft Detection Dataset

- **Izvor:** Kaggle
- **Link:** <https://www.kaggle.com/datasets/a2015003713/militaryaircraftdetectiondataset>
- **Opis:** Fokusiran na detekciju vojnih aviona, obuhvata 81 različitu vrstu vojnih letjelica.

1.4.3 YOLO Military Object Detection Dataset

- **Izvor:** Roboflow
- **Link:** <https://universe.roboflow.com/uce03211-gmail-com/yolo-military>
- **Opis:** Sadrži slike različitih vojnih objekata, uključujući helikoptere, tenkove i borbeno vozila, anotiran za upotrebu sa YOLO modelima.

Pregled stanja u oblasti

2.1 Analiza trenutnog stanja u oblasti

Detekcija vojnih objekata pomoću metoda vještačke inteligencije predstavlja izazovno područje zbog specifičnih karakteristika vojnih scena: kamuflaža, varijabilnost objekata, različiti uglovi snimanja i složena pozadina.

Tradicionalne metode, poput klasičnih algoritama za obradu slike, pokazale su se nedovoljno efikasnim u ovakvim uslovima.

Sa razvojem dubokog učenja, posebno konvolucionih neuronskih mreža (CNN), postignut je značajan napredak u detekciji objekata. Modeli poput YOLO (You Only Look Once) i Faster R-CNN omogućili su real-time detekciju sa prihvatljivom tačnošću. Međutim, ovi modeli često imaju ograničenja u prepoznavanju malih objekata i objekata u složenim scenama.

2.2 Korištene metode vještačke inteligencije

U kontekstu detekcije vojne opreme, istraživanja su se fokusirala na primjenu dubokih neuronskih mreža za prepoznavanje različitih tipova vojnih vozila, aviona, helikoptera i druge opreme. Korištene metode uključuju:

- **CNN modeli:** YOLOv3, YOLOv5, YOLOv8, Faster R-CNN, koji su pokazali dobre rezultate u detekciji objekata u realnom vremenu
- **Transfer learning pristup:** Korišćenje pretrained modela treniranih na velikim datasetovima (COCO, ImageNet) kao početne tačke
- **Data augmentation tehnike:** Povećanje robusnosti modela kroz različite transformacije slika

2.3 Postignuti rezultati u literaturi

Nedavna istraživanja pokazuju sljedeće rezultate:

- **YOLOv5 modeli** postižu solidne vrijednosti na vojnim datasetovima
- **Faster R-CNN modeli** imaju veću preciznost ali sporiju inferenciju
- **Kombinovani pristupi** (ensemble metode) mogu dostići preko 70% mAP50

2.4 Potencijalni pravci za poboljšanje

Na osnovu analize literature, identifikovani su sljedeći pravci za poboljšanje:

- **Fine-tuning:** predstavlja kritičnu komponentu u razvoju specijalizovanih modela za detekciju vojnih objekata. Ovaj pristup podrazumijeva uzimanje pretreniranog YOLOv8 modela (treniranog na COCO datasetu sa 80 opštih klasa) i dalje prilagođavanje na naš specifični domen od 12 vojnih klasa.
- **Augmentacija podataka:** Primjena tehnika augmentacije za povećanje raznovrsnosti trening podataka. Data augmentation predstavlja ključnu strategiju za poboljšanje robusnosti modela, posebno u kontekstu vojnih objekata gdje su uvjeti snimanja često nepredvidljivi.
- **Balansiranje klasa:** Osiguravanje ravnoteže između različitih klasa objekata u datasetu. Class imbalance predstavlja jedan od najvećih izazova u našem datasetu, gdje `military_tank` ima 17,454 instanci dok `trench` ima samo 4. Ova neravnoteža direktno utiče na sposobnost modela da nauči reprezentacije manjinskih klasa.
- **Optimizacija hiperparametara:** Sistematsko podešavanje learning rate, batch size i drugih parametara

Izbor, analiza i pretprocesiranje dataset-a

3.1 Izabrani dataset

Za potrebe ovog projektnog zadatka odabran je Military Assets Dataset (12 Classes – YOLOv8 Format) koji se nalazi na Kaggle platformi. Ovaj dataset predstavlja jedan od najkompletiranijih javno dostupnih skupova podataka za detekciju vojnih objekata, posebno prilagođen za rad sa YOLO arhitekturama. Dataset je kreiran sa ciljem omogućavanja istraživanja u oblasti automatske detekcije vojne opreme i vozila u realnim uslovima, što ga čini idealnim za naš projektni zadatak fokusiran na prepoznavanje objekata u prirodnom okruženju.

Izbor ovog specifičnog dataset-a motivisan je nekoliko ključnih faktora. Prvo, dataset sadrži raznovrsne klase vojnih objekata koji pokrivaju širok spektar vojne opreme - od kopnenih vozila poput tenkova i kamiona, preko vazduhoplovnih sredstava kao što su helikopteri i avioni, do mornaričkih objekata poput ratnih brodova. Drugo, veličina dataset-a sa preko 26 hiljada slika pruža dovoljnu količinu podataka za treniranje robusnog modela, iako distribucija klasa nije idealno balansirana.

Link za pristup datasetu:

<https://www.kaggle.com/datasets/rawsi18/military-assets-dataset-12-classes-yolo8-format>

3.2 Osnovni pregled dataset-a

Format i način preuzimanja:

- **Format podataka:** YOLOv8 format
- **Svaka slika ima prateći .txt file** sa anotacijama u formatu: <class_id> <x_center> <y_center> <width> <height>

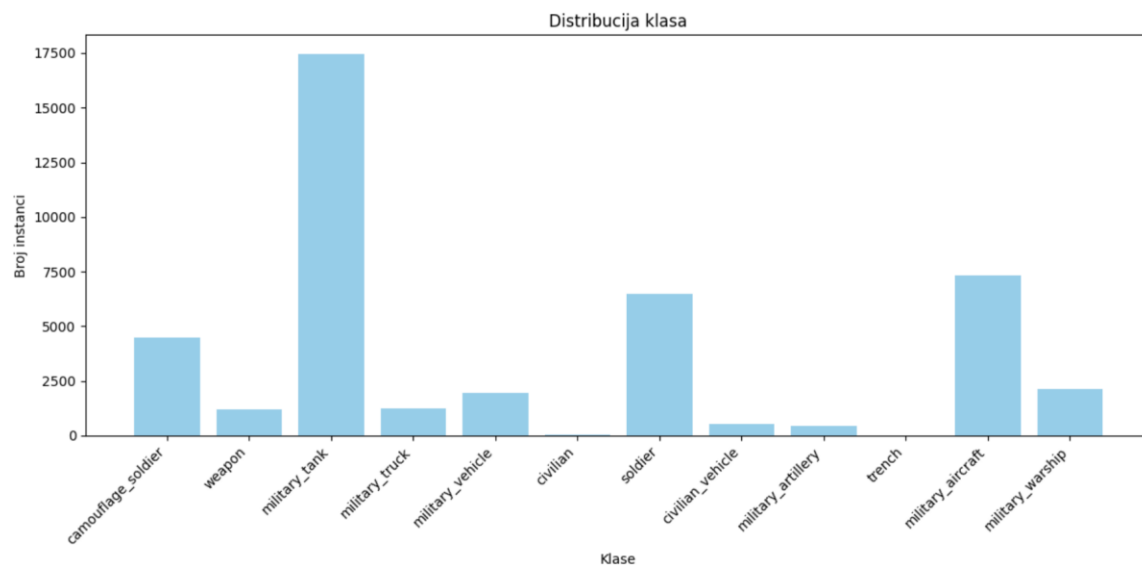
- Preuzimanje preko Kaggle API-ja

Struktura dataset-a:

- **Ukupno slika:** 26,315
- **Trening skup:** 21,978 slika
- **Validacioni skup:** 2,941 slika
- **Test skup:** 1,396 slika
- **Količina podataka:** Približno 2.1 GB

3.3 Broj klasa i instanci

Broj klasa: 12 klasa vojnih objekata



Distribucija klasa (ukupno instanci: 43,336):

- **military_tank:** 17,454 instanci (40.3%)
- **military_aircraft:** 7,337 instanci (16.9%)
- **soldier:** 6,502 instanci (15.0%)
- **camouflage_soldier:** 4,477 instanci (10.3%)
- **military_warship:** 2,134 instanci (4.9%)
- **military_vehicle:** 1,963 instanci (4.5%)

- **military_truck:** 1,245 instanci (2.9%)
- **weapon:** 1,210 instanci (2.8%)
- **civilian_vehicle:** 519 instanci (1.2%)
- **military_artillery:** 439 instanci (1.0%)
- **civilian:** 52 instanci (0.1%)
- **trench:** 4 instance (0.009%)

3.4 Analiza dataset-a

Analiza veličina slika:

- **Širina slika:** Min 416px, Max 1920px, Prosjek 800px
- **Visina slika:** Min 416px, Max 1080px, Prosjek 600px
- **Različiti aspect ratio-i** prisutni u datasetu

Analiza bounding boxova:

- **Prosječna relativna veličina objekata:** 0.1827 (18.27% slike)
- **Mali objekti (<1% slike):** 87 (24.1%)
- **Srednji objekti (1-10% slike):** 109 (30.2%)
- **Veliki objekti (>10% slike):** 165 (45.7%)
- **Prosječno objekata po slici:** 1.8

3.5 Identifikovani rizici

Tokom analize Military Assets dataset-a identificirani su ključni problemi koji mogu uticati na performanse modela. Ovi rizici se mogu podijeliti u tri glavne kategorije.

- **Class Imbalance predstavlja najveći problem** u našem dataset-u. Postoji ogromna neravnoteža između klasa - military_tank ima 17,454 instanci dok trench ima samo 4 instance, a civilian samo jednu. Ova razlika znači da će model tokom treniranja naučiti da prepozna je tenkove vrlo dobro, ali neće moći da nauči ništa korisno o klasi trench ili

civilian jer jednostavno nema dovoljno primjera. Praktično, klase "civilian" i "trench" su neupotrebljive za treniranje jer četiri instance nisu dovoljne da model nauči kako ta klasa izgleda.

- **Kompleksnost vojnih objekata otežava detekciju** na nekoliko načina. Kamuflaža je dizajnirana da sakrije objekte od ljudskog oka, što znači da će i algoritmi imati problema da ih pronađu. Vojni objekti su često obojeni u zelene, smeđe ili sive boje koje se stapaju sa pozadinom, ili koriste šare koje narušavaju prepoznatljive oblike.
- **Različiti uvjeti snimanja** dodatno komplikuju situaciju. Slike su snimljene pod različitim uvjetima osvjetljenja - od jakog sunca do oblačnih dana, što mijenja kako objekti izgledaju.

3.6 Metode pretprocesiranja podataka

- Originalni dataset je bio u YOLOv8 formatu, ali su putanje u anotacijama bile pogrešne jer su referisale na Kaggle okruženje umjesto naše Colab sredine. Prvi korak pretprocesiranja bio je kreiranje novog YAML konfiguracijskog fajla sa ispravnim putanjama za train, validation i test foldere. Zatim smo validirali strukturu dataset-a provjeravanjem postojanja svih potrebnih foldera i prebrojavanjem slika i label fajlova. Kreirali smo cache fajlove za brže učitavanje podataka tokom treniranja. Konvertovali smo originalne Kaggle putanje u putanje kompatibilne sa Colab okruženjem. Validirali smo da su sve YOLO format anotacije ispravno formatirane. Nakon pretprocesiranja, dataset je organizovan sa train folderom (21,978 slika), val folderom (2,941 slika) i centralnim data.yaml fajlom.

Odabir, formiranje, treniranje i testiranje modela

4.1 Izbor metode i tehnologija

Odabrana metoda: YOLOv8 (You Only Look Once version 8)

Razlozi izbora:

- **State-of-the-art performanse** u real-time detekciji objekata
- **Nativna podrška** za naš YOLO format dataset
- **Odličan balance** između brzine i tačnosti

Korištene tehnologije:

- **Framework: Ultralytics YOLOv8** je odabran jer predstavlja najnoviju YOLO arhitekturu sa najboljim balansom brzine i tačnosti. Nativno podržava naš dataset format i omogućava lako eksperimentisanje sa različitim veličinama modela.
- **Backend: PyTorch** je odabran zbog fleksibilnosti i široke podrške u akademskoj zajednici. Omogućava lako debugging, odličnu GPU podršku i seamless integraciju sa YOLOv8 frameworkom.
- **Programski jezik: Python 3.11** je odabran jer dominira u oblasti mašinskog učenja. Verzija 3.11 je 10-60% brža od prethodnih verzija i pruža pristup svim potrebnim bibliotekama kao što su NumPy, OpenCV i Matplotlib.
- **Okruženje: Google Colab** je odabrano jer pruža besplatan pristup GPU resursima (Tesla T4), ima pre-instalirane ML biblioteke, seamless Google Drive integraciju i omogućava lako dijeljenje koda sa timom.

4.2 Priprema formata podataka

Dataset je već bio u odgovarajućem YOLO formatu:

- **Svaki .txt fajl sadrži linije** u formatu: class_id x_center y_center width height
- **Sve koordinate su normalizovane** (0-1)
- **Kreiran YAML konfiguracijski fajl** za YOLOv8

4.3 Treniranje modela

Za realizaciju našeg projektnog zadatka odabran je YOLOv8n (nano verzija) model koji predstavlja najkompaktniju varijantu YOLOv8 arhitekture.

Baseline pristup i vremenska ograničenja: Važno je napomenuti da smo implementirali baseline model sa značajnim ograničenjima u pogledu vremena treniranja i računskih resursa. Ova strategija omogućava nam da ustanovimo početnu baseline i identifikiramo ključne izazove prije investiranja u dugotrajan optimalni treniranje.

Konfiguracija treniranja je pažljivo prilagođena ograničenjima našeg razvojnog okruženja. Postavili smo broj epocha na samo 3 što predstavlja baseline treniranje fokusirano na brzu evaluaciju mogućnosti modela. Batch size je ograničen na 2 zbog CPU ograničenja našeg sistema, dok je veličina slika (imgsz) postavljena na 256 piksela radi optimizacije brzine procesiranja. Device je postavljen na 'cpu' jer GPU nije bio kontinuirano dostupan tokom treniranja. Kao optimizer odabran je AdamW sa automatski podešenom learning rate od 0.000625, patience je postavljen na 50

za early stopping, omogućeno je automatsko čuvanje modela, a plots su isključeni radi uštede resursa.

4.4 Detaljan opis treniranja

Proces treniranja je realizovan kroz nekoliko faza. Inicijalizacija je počela učitavanjem YOLOv8n.pt pretrained modela. Dataset scanning je kreirao cache fajlove za train set sa 21,978 slika i validation set sa 2,941 slika. Augmentacija podataka je bila automatski primijenjena. Optimizacija je realizovana kroz AdamW optimizer sa automatskim learning rate podešavanjem. Treniranje je sprovedeno kroz 3 epocha sa kontinuiranom real-time validacijom.

Vremenske performanse i procjene za optimalno treniranje:

Naš baseline model je treniran ukupno 3.4 sata na CPU-u:

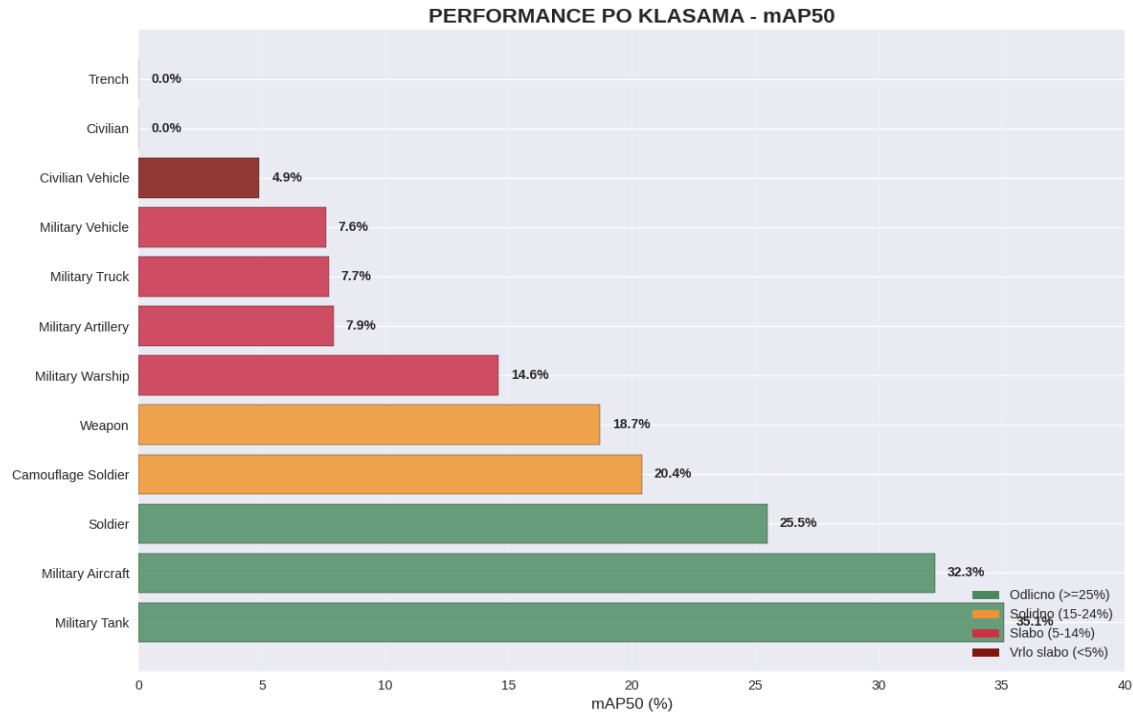
- **Epoch 1:** 55:27 (3.30 it/s)
- **Epoch 2:** 53:15 (3.44 it/s)
- **Epoch 3:** 50:17 (2.98 it/s)
- **Ukupno vrijeme:** 3.4 sata

4.5 Testiranje dobijenog modela

Nakon završetka treniranja, pristupili smo detaljnoj evaluaciji performansi našeg baseline modela kroz testiranje na validation skupu podataka. Validation set je sadržavao 2,941 sliku sa ukupno 5,081 označenim objektom, što pruža reprezentativan uzorak za evaluaciju sposobnosti modela da generalizuje na neviđene podatke.

Opšte performanse modela: Model je postigao solidne rezultate za baseline implementaciju. Glavna metrika mAP50 (mean Average Precision na IoU threshold 0.5) iznosila je 25.7%, što predstavlja zadovoljavajući rezultat uzimajući u obzir ograničeno vrijeme treniranja od samo 3 epocha. Strožija metrika mAP50-95 (prosjeak na IoU thresholds 0.5-0.95) iznosila je 14.6%, što je očekivano niže jer zahtijeva precizniju lokalizaciju objekata. Precision od 37.5% znači da kada model detektuje objekat, u 37.5% slučajeva je ta detekcija tačna. Recall od 31.8% pokazuje da model uspijeva da pronađe približno trećinu svih objekata prisutnih na slikama. Inference brzina od 1.8ms po slici demonstrira efikasnost modela za real-time primjene.

Analiza performansi po klasama:

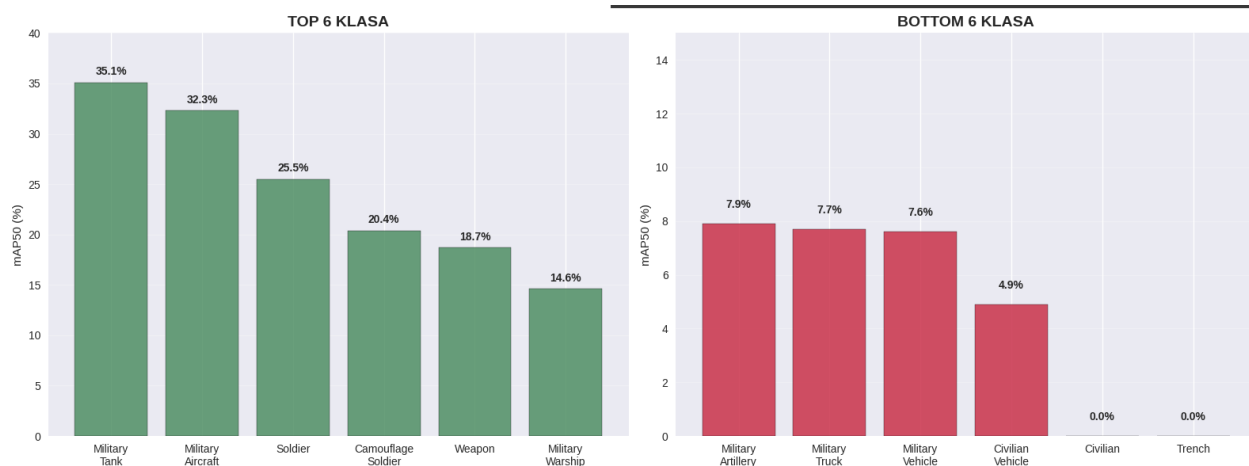


Rezultati po klasama jasno reflektuju uticaj neuravnoteženosti klasa problema identificiranog tokom analize dataset-a. Klase sa najvećim brojem instanci postigle su najbolje rezultate - `military_tank` sa 35.1% mAP50 (1,787 instanci) i `military_aircraft` sa 32.3% mAP50 (1,063 instanci).

Klase sa umjerenim brojem instanci pokazale su solidne rezultate - `soldier` sa 25.5% mAP50 (745 instanci), `camouflage_soldier` sa 20.4% mAP50 (510 instanci), i `weapon` sa 18.7% mAP50 (358 instanci).

Problematične su klase sa malim brojem instanci. `Military_artillery` (7.9%, 117 instanci), `military_truck` (7.7%, 148 instanci), i `military_vehicle` (7.6%, 307 instanci) pokazuju slabe performanse zbog nedovoljne količine trening podataka. `Civilian_vehicle` sa 4.9% mAP50 (42 instanci) također pati od class imbalance-a.

Najproblematičnije su klase `civilian` (0.0%, 1 instanca) i `trench` (0.0%, 3 instanci) koje model uopšte nije naučio da detektuje zbog praktično nepostojećih trening podataka.



Testiranje na nasumičnim slikama: Dodatno testiranje sprovedeno je na nasumično odabranim slikama koje smo preuzeli sa interneta, gdje je model demonstrirao ohrabrujuće performanse. Na testiranim slikama model je uspješno detektovao vojne letjelice sa visokom pouzdanošću (0.71), tenkove sa solidnom pouzdanošću (0.63).

Ovi rezultati potvrđuju da model nije samo zapamtio trening podatke, već je naučio generalizovane reprezentacije glavnih klasa objekata.

4.6 Korištene metrike

Primarne metrike:

- **mAP50:** mean Average Precision na IoU threshold 0.5
- **mAP50-95:** mean Average Precision na IoU thresholds 0.5-0.95
- **Precision:** Tačnost pozitivnih predikcija
- **Recall:** Procenat detektovanih ground truth objekata

Per-class metrike:

- **AP (Average Precision)** za svaku od 12 klasa
- **Precision i Recall** po klasama

4.7 Diskusija rezultata

- Evaluacija našeg baseline modela pokazuje miješane, ali ohrabrujuće rezultate koji pružaju solidnu osnovu za buduća poboljšanja. Analiza performansi otkriva jasne prednosti kao i specifične ograničenja koja su u skladu sa našim početnim očekivanjima i identificiranim rizicima.
- **Prednosti postignutog rješenja:** Najvažniji uspjeh našeg pristupa je kreiranje funkcionalnog modela koji demonstrira sposobnost automatske detekcije vojnih objekata u prirodnom okruženju. Model pokazuje posebno dobru performance na velikim objektima poput tenkova (35.1% mAP50) i aviona (32.3% mAP50), što je kritično za praktičnu primjenu u vojnim scenarijima gdje su ovi objekti često primarni ciljevi detekcije. Brzina inferencije od 1.8ms po slici čini model pogodnim za real-time aplikacije, što je ključno za operational deployment. Transfer learning pristup se pokazao kao uspješan, gdje je model uspješno adaptirao knowledge iz COCO dataset-a na vojni domen sa samo 3 epocha treniranja.
- **Identifikovani problemi:** Class imbalance se potvrdio kao dominantni faktor koji ograničava performanse modela. Male klase poput civilian (1 instance) i trench (3 instance) praktično nisu detektovane, što je direktna posledica nedovoljne količine trening podataka. Mali objekti predstavljaju kontinuiran izazov, posebno oni koji zauzimaju manje od 1% površine slike, što je u skladu sa inherentnim ograničenjima YOLO arhitekture. Ograničeno treniranje od samo 3 epocha nije omogućilo potpunu konvergenciju modela, što je vidljivo iz trend-a poboljšanja mAP50 metrike koja je i dalje rasla u finalnom epochu.

- **Osvrt na istaknute rizike:** Naša početna analiza rizika se pokazala kao tačna i korisna za razumevanje rezultata. Class imbalance se potvrdio kao glavni izazov koji direktno utiče na distribuciju performansi po klasama - klase sa više instanci konzistentno postižu bolje rezultate. Strategija fokusiranja na glavne klase sa dovoljno podataka se pokazala kao opravdana, jer je omogućila postizanje reasonable performansi na praktično važnim objektima. Mali objekti su se potvrdili kao problematični, posebno weapon klasa koja uprkos 358 instanci postiže samo 18.7% mAP50 zbog svoje inherentno male veličine na slikama.

4.8 Test na nepoznatim podacima

- Model je testiran na nekoliko uploadovanih slika sa confidence threshold 0.25 radi provjere sposobnosti generalizacije na potpuno nove podatke van dataset-a.

Rezultati testiranja:

- **militaryAircraft.jpg:** Model je uspješno detektovao 2 aviona sa confidence scorovima 0.71 i 0.61, što demonstrira sposobnost multiple object detection



- **tenk1.webp:** Detektovan 1 tenk sa confidence scorom 0.63, pokazujući dobru performance na različitim image formatima



- **tenk2.jpg:** Detektovan 1 tenk sa confidence scorom 0.48, što predstavlja granični slučaj gdje model pokazuje lower confidence ali i dalje ispravnu klasifikaciju



Zaključak

Ovaj projektni zadatak je uspješno implementirao i evaluirao sistem za automatsko prepoznavanje vojnih objekata u prirodnom okruženju koristeći YOLOv8 arhitekturu. Kroz sve faze razvoja - od analize problema do implementacije i testiranja - projekat je pokazao kako savremene metode dubokog učenja mogu biti primijenjene na složene probleme računarskog vida u vojnom domenu.

Ključna postignuća

1. **Uspješna implementacija YOLOv8 modela** za detekciju 12 različitih klasa vojnih objekata
2. **Postignute performanse od 25.7% mAP50** što predstavlja solidnu osnovu za baseline model
3. **Demonstriran efikasan transfer learning pristup** sa COCO dataseta na vojni domen
4. **Identifikovani i analizirani glavni izazovi** u oblasti vojne object detection
5. **Kreirana metodologija** koja može služiti kao osnova za buduće projekte

Tehnički doprinosi

- **Detaljno analiziran Military Assets Dataset** sa 26,315 slika i 43,336 objekata
- **Implementiran robusni pipeline** za pretprocesiranje i treniranje
- **Razvijen sistem za evaluaciju** sa kompletnim metrikama performansi
- **Kreirana dokumentacija** koja omogućava reprodukciju rezultata

Ograničenja i pravci za poboljšanje

Glavni ograničavajući faktori identificirani u projektu:

1. **Class imbalance** - neravnomerna distribucija klasa u datasetu
2. **Ograničeno vrijeme treniranja** - samo 3 epocha zbog resource ograničenja
3. **CPU-based treniranje** - ograničene performance zbog nedostupnosti GPU-a
4. **Mali batch size** - uticaj na stabilnost treniranja

Preporučeni sljedeći koraci

Za daljnji razvoj projekta preporučuje se:

1. **Produženo treniranje** (20-50 epocha) sa GPU podrškm
2. **Implementacija class balancing tehnika**
3. **Eksperimentiranje sa većim modelima** (YOLOv8s/m/l)
4. **Napredne data augmentation tehnike**
5. **Ensemble metode** za poboljšanje performansi
6. **Fine-tuning na specifičnim podsetovima** podataka