

Zadatak 1 – semantička segmentacija - U-Net – izdvajanje building footprint extraction

Učitati ortofoto područja od interesa. Kreirati obučavajući skup (maske zgrada ili shp konvertovan u raster). Prikazati preklapanje maske i ortofoto. Pripremiti podatke za segmentaciju i podeliti skup na **train/val/test**. Implementirati U-Net i trenirati mrežu da prepozna piksele zgrada (building=1, background=0). Evaluirati model pomoću standardnih metrika (IoU, F1-score). Prikazati originalnu sliku, ground truth masku i predikciju modela. Uraditi postprocessing dobijenih rezultata i pretvoriti ih u poligone (vektore). Dokumentovati rešenje.

Na odbrani predati word/pdf dokument u kom je opisano rešenje, izvorni kod i demonstrirati kako radi.

Zadatak 1a – koristiti DeepLab umesto U-Net

Zadatak 2 – segmentacija instanci - Mask R-CNN - building footprint extraction

Učitati ortofoto područja od interesa. Kreirati obučavajući skup (maske zgrada ili shp konvertovan u raster). Prikazati preklapanje maske i ortofoto. Pripremiti podatke za segmentaciju i podeliti skup na **train/val/test**. Implementirati **Mask R-CNN**. Pripremiti podatke tako da svaka zgrada ima poseban ID u maski. Trenirati mrežu da izdvoji pojedinačne zgrade sa ortofotoa. Evaluirati model pomoću standardnih metrika (Average Precision - AP). Prikazati originalnu sliku, ground truth masku i predikciju modela. Uraditi postprocessing dobijenih rezultata i pretvoriti ih u poligone (vektore). Dokumentovati rešenje.

Na odbrani predati word/pdf dokument u kom je opisano rešenje, izvorni kod i demonstrirati kako radi.

Zadatak 3 – semantička segmentacija - U-Net – ekstrakcija vodenih površina

Učitati ortofoto područja od interesa. Kreirati obučavajući skup (maske zgrada ili shp konvertovan u raster). Prikazati preklapanje maske i ortofoto. Pripremiti podatke za segmentaciju i podeliti skup na **train/val/test**. Implementirati U-Net i trenirati mrežu da prepozna piksele zgrada (building=1, background=0). Evaluirati model pomoću standardnih metrika (IoU, F1-score). Prikazati originalnu sliku, ground truth masku i predikciju modela. Uraditi postprocessing dobijenih rezultata i pretvoriti ih u poligone (vektore). Dokumentovati rešenje.

Na odbrani predati word/pdf dokument u kom je opisano rešenje, izvorni kod i demonstrirati kako radi.

Zadatak 4 – semantička segmentacija - DeepLabV3 – ekstrakcija vodenih površina

Učitati ortofoto područja od interesa. Kreirati obučavajući skup (maske zgrada ili shp konvertovan u raster). Prikazati preklapanje maske i ortofoto. Pripremiti podatke za segmentaciju i podeliti skup na **train/val/test**. Implementirati U-Net i trenirati mrežu da prepozna piksele

zgrada (building=1, background=0). Evaluirati model pomoću standardnih metrika (IoU, F1-score). Prikazati originalnu sliku, ground truth masku i predikciju modela. Uraditi postprocessing dobijenih rezultata i pretvoriti ih u poligone (vektore). Dokumentovati rešenje.

Na odbrani predati word/pdf dokument u kom je opisano rešenje, izvorni kod i demonstrirati kako radi.

Zadatak 5 – semantička segmentacija - U-Net – ekstrakcija puteva

Učitati ortofoto područja od interesa. Kreirati obučavajući skup (maske zgrada ili shp konvertovan u raster). Prikazati preklapanje maske i ortofoto. Pripremiti podatke za segmentaciju i podeliti skup na **train/val/test**. Implementirati U-Net i trenirati mrežu da prepozna piksele zgrada (building=1, background=0). Evaluirati model pomoću standardnih metrika (IoU, F1-score). Prikazati originalnu sliku, ground truth masku i predikciju modela. Uraditi postprocessing dobijenih rezultata i pretvoriti ih u poligone (vektore). Dokumentovati rešenje.

Na odbrani predati word/pdf dokument u kom je opisano rešenje, izvorni kod i demonstrirati kako radi.

Zadatak 6 - Klasifikacija zemljišnog pokrivača sa satelitskog snimka

Sa multispektralnog satelitskog snimka (Sentinel-2, Landsat) klasifikovati zemljišni pokrivač u nekoliko klasa (vodene površene, urbano područje, poljoprivredno zemljište, šume, livade i pašnjaci...). Kao maske (ground-truth labele) koristiti postojeće skupove podataka kao što su Copernicus CORINE land cover, ESA WorldCover, INSPIRE, GeoSrbija, open.data.rs, geofabrik vektorske podatke koji se konvertuju u raster... Implementirati U-Net ili DeepLabV3 za multi-class semantičku segmentaciju. Definiši loss funkciju. Podeliti podatke na train/val/test. Trenirati model određeni broj epoha. Evaluirati pomoću standardnih metrika (Overall Accuracy, IoU po klasi, Confusion matrix). Prikazati predikcije na test snimcima pored ground truth maske. Generisati klasifikacionu mapu u boji (svaka klasa posebna boja). Napraviti statistiku površina po klasama (u hektarima).

Zadatak 7 – Detakcija objekata – detekcija sportsko rekreativnih površina

Potrebno je razviti model za detekciju sportsko-rekreativnih površina u gradu (parkovi, park-šume, izletišta, fudbalska i košarkaška igrališta, teniski tereni, bazeni, otvorene teretane, igrališta za decu...) korišćenjem metoda detekcije objekata na ortofoto snimcima. Studenti treba da pripreme ili pronađu dataset sa odgovarajućim oznakama, da treniraju model (npr. YOLO ili Faster R-CNN) i da detektuju lokacije sportskih i rekreativnih objekata u odabranom urbanom području. Nakon detekcije, potrebno je izvršiti analizu dostupnosti i pokrivenosti, odnosno uporediti broj i raspored otkrivenih objekata sa brojem stanovnika u različitim delovima grada, i proceniti da li trenutni kapacitet sportskih i rekreativnih površina zadovoljava potrebe lokalnog stanovništva.

Zadatak 8 - SQL

- Napravite bazu u PostgreSQL pod nazivom **univerzitet**.
- Povežite se na tu bazu iz Python-a koristeći biblioteku **psycopg2**.

Korišćenjem biblioteke u bazi kreirati tri tabele:

- **Student**
 - student_id (PRIMARY KEY, automatski rastući broj – koristite SERIAL)
 - ime
 - prezime
 - godina_rođenja
- **Kurs**
 - kurs_id (PRIMARY KEY, SERIAL)
 - naziv
 - espb
- **Upis**
 - student_id (FOREIGN KEY → Student)
 - kurs_id (FOREIGN KEY → Kurs)
 - ocena
- Ubacite najmanje **5 studenata** u tabelu Student.
- Ubacite najmanje **5 kurseva** u tabelu Kurs.
- Ubacite zapise u tabelu Upis (npr. koji student je upisan na koji kurs i koju ocenu je dobio).
- Pomoću **pandas** biblioteke pročitajte sve podatke iz tabela i učitajte ih u **DataFrame**.
- Izmenite neki podatak u tabeli (npr. promenite ocenu jednom studentu).
- Dodajte novog studenta i upišite ga na jedan kurs.

Napravite i testirajte sledeće upite (rezultate prikažite pomoću pandas DataFrame-a):

1. Prikaz svih studenata sa njihovim kursevima i ocenama.
2. Studenti koji imaju ocenu veću od **8**.
3. Prosečna ocena po kursu.

4. Broj studenata upisanih na svaki kurs.
5. Pretraga studenata po prezimenu (npr. svi studenti sa prezimenom *Petrović*).

Sve raditi kroz Python okruženje.

Zadatak 9 – Geo 1

- Preuzmite shapefile sa granicama opština u Srbiji (npr. geofabrik).
- U Python-u učitajte te podatke pomoću **geopandas** biblioteke
- Napravite DataFrame (pandas) sa informacijama o broju stanovnika za nekoliko opština:
 - opština, stanovništvo
- Spojite shapefile (geopandas) sa vašom tabelom broja stanovnika koristeći kolonu opština.
- Dodajte novu kolonu gustoća = broj stanovnika / površina opštine
- Izmenite broj stanovnika za jednu opštinu (ažuriranje podataka).
- Nacrtajte mapu sa opštinama obojenim prema broju stanovnika.
- Nacrtajte mapu sa opštinama obojenim prema gustini stanovništva.

Zadatak 10 – Geo 2

- Preuzmite shapefile sa rekama i naseljima u Srbiji (npr. geofabrik).
- Učitati shapefile sa rekama (linije).
- Učitati shapefile ili GeoJSON sa naseljima (tačke).
- Pronaći **najdužu reku** u datasetu
- Dodati novu kolonu dužina_km.
- Pronaći sva naselja koja se nalaze u radijusu od **5 km** od reke Dunav.
- Dodajte novo naselje u GeoDataFrame (ručno unesite tačke sa koordinatama).
- Izlistajte sva naselja čija populacija prelazi 10,000 i koja su na reci.
- Nacrtajte reke i naselja na istoj mapi.
- Obeležite naselja u blizini Dunava različitom bojom.

