**Klasifikacija tipova aviona korišćenjem konvolucionih neuronskih mreža (CNN) i SVM algoritma**

**Definicija problema:**  
Prepoznavanje i klasifikacija tipova aviona na osnovu vizuelnih podataka predstavlja izazovan problem u oblasti veštačke inteligencije. Zbog sličnosti karakteristika različitih modela njihovo razlikovanje tradicionalnim metodama obrade slike je otežano. Cilj projekta je razvoj sistema koji pomoću konvolucionih neuronskih mreža (CNN) vrši ekstrakciju osobina, a potom klasifikaciju tipova aviona primenom SVM (Support Vector Machine) algoritma. Na ovaj način kombinuju se prednosti dubokog učenja u prepoznavanju obrazaca i robustnost SVM-a u klasifikacionim zadacima.

**Motivacija:**  
Precizna klasifikacija tipova aviona ima značajnu primenu u avio-industriji, bezbednosnim i nadzornim sistemima, kao i u istraživačkim projektima. Automatsko prepoznavanje ubrzava identifikaciju i smanjuje rizik od grešaka. Ovakav sistem može doprineti unapređenju softverskih alata za praćenje saobraćaja i analizu avioflote. Spajanje CNN-a i SVM-a pruža pouzdano i praktično primenjivo rešenje.

**Skup podataka:**

U projektu će se koristiti skup podataka koji u sebi sadrži

[**https://www.robots.ox.ac.uk/~vgg/data/fgvc-aircraft/**](https://www.robots.ox.ac.uk/~vgg/data/fgvc-aircraft/)

U ovom projektu će biti korišćen FGVC-Aircraft dataset koji sadrži oko **10 000 slika(instanci)** aviona koji su raspoređeni **u 100 klasa (100 različitih tipova aviona)**.Za ovaj projekat biće korišćeno 10 klasa koje će biti odabrane po kriterijumu zastupljenosti ili vizuelne različitosti kako bi treniranje bilo stabilnije i evaluacija kvalitetnija.Organizacija podataka je takva da su slike u jedinstvenom folderu dok label fajlovi mapiraju naziv slike na klasu.Ovaj dataset sadrži i podelu na train/val/test skupove ali u projektu je implementirana vlastita podela podataka.

**Način pretprocesiranja podataka:**

Priprema podataka za ulazak u model mašinskog učenja vrši se u nekoliko faza pretprocesiranja:

1. **Učitavanje podataka:** U ovoj fazi se učitavaju slike koje se nalaze u jednoj folderu a label fajl mapira svaku sliku na klasu.
2. **Promena dimenzija slika (resize):** Neophodno je da se sve slike svedu na istu dimenziju (recimo 128x128) da bi ulaz u CNN bio uniforman.
3. **Normalizacija piksela:** Pikseli se skaliraju na [0,1] ili standardizuju prema srednjoj vrednosti i standardnoj devijaciji.
4. **Augmentacija podataka:** Na train skupu će biti primenjene tehnike augmentacije (rotacije, horizontalni/vertikalni flip, translacije, skaliranje, promene osvetljenja, dodavanje šuma) kako bi se povećala raznovrsnost podataka i smanjio rizik od overfittinga.

**Metodologija- nacin resavanja problema:**

Način rešavanja problema se odvija u nekoliko faza:

**1.Ekstrakcija osobina pomoću CNN-a:** Konvolucione neuronske mreže će se koristiti za automatsko izdvajanje reprezentativnih vizuelnih osobina slika. Model će se sastojati od niza konvolucionih i pooling slojeva dok će finalni vektori osobina biti prosleđeni SVM-u za klasifikaciju. Za treniranje CNN modela koristiće se optimizacioni algoritmi kako bi se minimizovala funkcija greške. Izbor optimizatora i vrednosti parametara će se podešavati eksperimentalno radi postizanja najboljih rezultata.

**2.Formiranje vektora osobina:** Rezultati koje vrati CNN će se pretvoriti u vektore koji sadrže najvažnije karakteristike za svaku sliku.

**3.Klasifikacija pomoću SVM-a:** Ovi dobijeni vektori se potom proslede Support Vector Machine algoritmu koji vrši konačnu klasifikaciju tipova aviona.

**4.Hiperparametri:** Za treniranje modela koristiće se različiti hiperparametri, kao što su veličina kernela, broj filtera i epoha za CNN, kao i izbor kernel funkcije i regularizacioni parametar za SVM. Ove vrednosti će biti podešavane tokom sprovođenja eksperimenta da bi se postigle najbolje performanse.

**Način evaluacije:**

**1.Podela podataka:** Podaci će biti podeljeni na train/validation/test skupove.Test skup će služiti za proveru konačnih performansi.

**2.Metrike performansi:** S obzirom da dataset sadrži približno isti broj instanci po klasi (70–100 slika), glavna metrika za evaluaciju modela biće **accuracy**. Pored toga koristiće se **confusion matrix** radi vizuelne analize grešaka i uvida u to koje klase se najviše mešaju(neki tipovi aviona liče jedni na druge),a metrike F1-score,Recall,Precision će se koristiti za detaljniju sliku performansi.Ovde se očekuje da će augmentacija koja je primenjena u predprocesiranju podataka doprineti boljim vrednostima metrika (accuracy, F1-score, precision, recall) u poredjenju sa baznim modelom koji je bez augmentacije.

**3.Cross-validation**: Koristiće se k-fold cross validation da bi se proverila stabilnost modela i da bi se smanjio rizik od toga da je rezultat posledica slučajne podele podataka.

**4. Poređenje modela:** Uporediće se CNN+SoftMax i pristup koji je ovde korišćen, CNN+SVM gde će se videti da li kombinacija daje bolje rezultate.

**5. Ablaciona evaluacija (bez/sa augmentacijom):** Radi proveru uticaja augmentacije na generalizaciju: treniranje i evaluacija **bez augmentacije** i **sa augmentacijom** (RandomFlip, RandomRotation, RandomZoom, RandomContrast). Porediće se accuracy, macro-F1, kao i **generalization gap** (razlika train vs. validation/test performansi). Očekivanje je da će augmentacija smanjiti taj razmak i poboljšati metrike na validacionom i test skupu.

**Tehnologije:**

**Python** – programski jezik za implementaciju modela.

**Google Colab**  – okruženje za razvoj i treniranje modela.

**TensorFlow /Keras**  – biblioteke za implementaciju CNN-a i augmentacije.

**scikit-learn** – biblioteka za implementaciju SVM-a i evaluaciju.

**NumPy, Pandas** – za manipulaciju podacima i rad sa nizovima.

**Matplotlib / Seaborn** – za vizualizaciju rezultata (grafici, confusion matrix).

**Relevantna literatura i resursi:**

**1. Dataset:** [**https://www.robots.ox.ac.uk/~vgg/data/fgvc-aircraft/**](https://www.robots.ox.ac.uk/~vgg/data/fgvc-aircraft/)

**2.Goodfellow, I., Bengio, Y., Courville, A. *Deep Learning.* MIT Press, 2016.**  
 https://www.deeplearningbook.org/

**3.scikit-learn dokumentacija – SVM:**  
 https://scikit-learn.org/stable/modules/svm.html

**4.TensorFlow Keras dokumentacija – CNN:**  
https://www.tensorflow.org/tutorials/images/cnn

**5. Sličan projekat AFAgarap/cnn-svm** **(GitHub):** https://github.com/AFAgarap/cnn-svm