

# Detekcija godina

Cilj projekta je predviđanje starosti na osnovu slike korišćenjem konvolucijske neuronske mreže.

Skup podataka pomoću kog se vrši treniranje predstavlja 26,537 slika u čijem nazivu se nalazi broj godina koji osoba na slici ima.

Treniranje i testiranje modela je izvršeno pomoću Google Colab-a.

Na početku je potrebno učitati sve biblioteke koje koristimo:

```
import tensorflow as tf
import cv2
import os
from keras_preprocessing.image import ImageDataGenerator
from keras.models import Sequential
from keras.layers import Conv2D, Dense, Dropout, Flatten, AveragePooling2D, GlobalAveragePooling2D
from keras.callbacks import ModelCheckpoint
import numpy as np
from sklearn.model_selection import train_test_split
from keras.utils import np_utils
```

Postoji 7 klasa u koje svrstavamo slike na osnovu godina i one se nalaze u sledećim intervalima:

1-2, 3-9, 10-20, 21-27, 28-45, 46-65, 66-116.

Sve slike zajedno sa klasom kojoj pripadaju (nizovi X i y) čuvamo u datoteci compressed image data.npz koristeći funkciju np.savez\_compressed.

Zatim skup podataka delimo u skup za treniranje i za testiranje, proporcijom 70:30 za trening skup.

Kako bi performanse modela bile bolje, koristimo i tehniku povećanja broja slika za treniranje na osnovu onih koje već imamo, poznatiju kao image augmentation.

Za to koristimo klasu ImageDataGenerator, koju omogućava Keras.

Konkretno koristimo: skaliranje, smicanje, horizontalno okretanje (ogledalo), zumiranje, pomeranje po dužini, pomeranje po širini i rotiranje.

Zatim kreiramo neuronsku mrežu, korišćenjem Sequential modela, dodajući redom sledeće slojeve:

- konvolucijski: 32 filtera dimenzija 3x3, relu
- average pooling: 2x2
- konvolucijski: 64 filtera dimenzija 3x3, relu
- average pooling: 2x2
- konvolucijski: 128 filtera dimenzija 3x3, relu
- average pooling: 2x2
- konvolucijski: 256 filtera dimenzija 3x3, relu
- average pooling: 2x2
- global average pooling
- potpuno povezan sloj, relu (137)
- potpuno povezan, softmaks (7)

Koristimo i Adamov optimizacioni algoritam.

Zatim treniramo mrežu, broj epoha je 60, a batch je veličine 128.

Najbolji model, tačnije onaj sa najvećom tačnošću, čuvamo u final\_cnn\_model\_checkpoint.h5.

Dobijena je tačnost od 61%, taj model je sačuvan i time se završava proces treniranja.

Model učitavamo funkcijom load\_model i možemo započeti testiranje.

Pre svega je potrebno detektovati lice na slici koju korisnik unese.

Za detektovanje lica, koristi se jedan od OpenCv modela, koji se mogu naći na sledećem sajtu:

<https://github.com/opencv/opencv/tree/master/data/haarcascades>

Detektovano lice pretvaramo u crno-belu sliku dimenzija 200x200, jer se na takvim slikama vršilo treniranje i kao rezultat program vraća istu sliku sa detektovanim licem i grupom godina koju predviđa osobi na slici.

*Ana Cvijović, 646/2019*  
*Kristina Mojsić, 626/2019*