**Prepoznavanje cifara u znakovnom jeziku**

Kurs: Prepoznavanje oblika i obrada slike

**Tim:**

Irhad Halilović

Rasim Šabanović

Haris Osmanbegović

# Dataset

1. **Obrazložiti temu projekta**

Tema našeg projekta je prepoznavanje cifara u znakovnom jeziku. Naš cilj je napraviti klasifikator koji će priložene slike cifara u znakovnom jeziku klasificirati u njihove odgovarajuće klase (0-9)

1. **Preuzeti ili kreirati data set slika**

Dataset koji će biti korišten je preuzet sa sljedećeg linka: <https://www.kaggle.com/ardamavi/sign-language-digits-dataset?fbclid=IwAR2geHMhSsAq6BKQGDPS_VDEa6UA2XOk6XCsG6z50C6xZjHwpKWEuuXZa6s>.

Dataset se također nalazi i na repozitoriju.

1. **Koliko ima klasa i koje su klase?**

Ukupno postoji 10 klasa, jer postoji 10 cifara (kako inače, tako i u znakovnom jeziku) i to su klase cifara od 0 do 9.

1. **Koliko svaka klasa ima uzoraka**

Uzimajući u obzir da su uzorci uzimani od ukupno 218 studenata i da je svaki student davao ukupno 10 uzoraka (po jedan za svaku cifru), to znači da svaka klasa ima po 218 uzoraka.

# DataPrep1

1. **Kreirati maske slika na osnovu regiona od interesa.**

Nakon što su na dataturks.com označeni regioni od interesa, kreiran je json fajl koji sadrži koordinate ovih regiona. Maske slika se kreiraju pozivom funkcije   
*createMasks(jsonPath, masksFolderPath).* Ova funkcija se nalazi u fajlu Skripte/masks.py I prima dva parametra. Prvi parameter je putanja do kreiranog json fajla sa koordinatama regiona od interesa, a drugi parameter je lokacija foldera u koji će se spremiti kreirane maske.

1. **Primijeniti maske na slike**  
   Kreirane maske se primjenjuju na slike pozivom funkcije   
   *maskImages(maskPath, originalImagePath, maskedImagePath).* Parametri redom predstavljaju: putanju foldera u kojem se nalaze maske slika, putanju foldera u kojem se nalaze originalne slike, putanju foldera u koji će se spremiti kreirane slike.
2. **Spremiti nove slike u novi folder. Ovo je potrebno uraditi koristeći funkciju koja se treba sačuvati i koristiti u svim narednim koracima gdje se radi spremanje slika**Za spremanje slika se koriste funkcije iz fajla *fileScripts.py*. Funkcija za spremanje slika je *saveImage(image, location).* Ova funkcija kao parametre prima sliku te lokaciju na koju se sprema.

# DataPrep2

1. Korišten je median filter, zbog svoje mogućnosti da dobro sačuva kvalitet slike, jer su slike već poprilično kvalitetne.

filteredImage = ndimage.median\_filter(image, 1)

1. Uklanjanje neoštrina se izvelo nad setom slika koji je prošao kroz filtriranje sa median filterom, jer su one zamućene u tom procesu.

lap = cv2.Laplacian(image, cv2.CV\_64F)

sharpImage = image - 0.1\*lap

sharpImage = sharpImage.astype(numpy.uint8)

# DataPrep3

1. Poboljšanje kontrasta

image = Image.open(filename)

enhancer = ImageEnhance.Contrast(image)

enhanced = enhancer.enhance(2)

1. Poboljšanje svjetlosti

image = Image.open(filename)

enhancer = ImageEnhance.Brightness(image)

enhanced = enhancer.enhance(1.5)

Enhancer klasa poboljšava odgovarajuće svojstvo slike za onoliko koliko se navede kao parametar metode enhance.