VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

# Rozpoznávanie hlasu a obrazu

## Projekt SUR

Karol Šugár (xsugark00)

4.5.2025

# 1. Dataset

Základom projektu bol dataset poskytnutý v archíve SUR_projekt2024-2025.zip. Tento archív obsahoval adresáre train a dev, ktoré boli použité na trénovanie a vyhodnocovanie modelov počas vývoja. Pre finálne testovanie bol k dispozícii adresár eval. Adresáre train a dev obsahovali 31 podadresárov, označených číslami 1 až 31, ktoré zodpovedali jednotlivým osobám (triedam).

Každý podadresár v train adresári obsahoval 6 vzoriek pre danú osobu vo formáte WAV (hlas) a 6 vzoriek vo formáte PNG (tvár). V dev adresári boli pre každú osobu 2 WAV a 2 PNG súbory. Celkovo bolo teda k dispozícii 186 trénovacích vzoriek 62 testovacích vzoriek.

Názvy súborov mali štruktúru ako napr. f401_01_f12_i0_0.png, kde prvá časť (napr. f401) interne identifikovala osobu a druhá časť (napr. 01) označovala číslo nahrávacieho sedenia. Bolo pozorované, že interný identifikátor osoby (napr. f427) nekorešpondoval priamo s číselným názvom nadradeného adresára (napr. súbor f427 bol v adresári 28). Pre účely trénovania klasifikátorov boli ako referenčné triedy použité názvy adresárov.

## 1.1 Prostredie

Riešenie je implementované v Python 3 pomocou Jupyter Notebook. Projekt využíva nasledujúce knižnice:

NumPy a Matplotlib na prácu s dátami a vizualizáciu,OpenCV (cv2) na spracovanie obrazu, scikit-learn na strojové učenie (GaussianMixture, SVC, GridSearchCV, preprocessing), scikit-image na HOG extrakciu, joblib na ukladanie modelov,ikrlib na MFCC extrakciu

Projekt je štruktúrovaný na jednoduché použitie: modely sa trénujú, ukladajú do zložky a následne načítavajú pre evaluáciu. Vďaka tomuto prístupu stačí na iných počítačoch spustiť len konfiguračnú časť s importmi a stiahnuť si poskytnutý model, bez potreby trénovania na novo. Pre jednoduché zapnutie celého skriptu stačí pustit tlačítko kompilácie pre celý skript a pri dostupnosti zložiek test a train sa natrénuje a vyhodnotí test zložku( poprípade aj eval zložku)

# 2. Modely

Pri výbere modelov sme museli dávať pozor, či už nie sú natrénované, keďže sme mali zakázaný transfer learning. Pre jednotlivé modality boli zvolené nasledujúce prístupy:

Audio: Identifikácia na základe hlasu bola realizovaná pomocou Gaussových zmesových modelov (GMM) trénovaných na Mel-frekvenčných kepstrálnych koeficientoch (MFCC).

Obraz: Identifikácia na základe obrazu tváre využívala klasifikátor Support Vector Machine (SVM) trénovaný na príznakoch extrahovaných metódou Histogramov orientovaných gradientov (HOG).

## 2.1 Rozpoznávanie hlasu (Audio)

Model: Gaussove zmesové modely (GMM) s MFCC príznakmi.

Knižnica: sklearn.mixture.GaussianMixture zo scikit-learn a ikrlib waw16hz2mfcc().

### 2.1.1 Extrakcia príznakov a predspracovanie

Na extrakciu audia použijeme príznaky MFCC. Na  extrakciu bola použitá funkcia wav16khz2mfcc z poskytnutej knižnice ikrlib, ktorá spracovala WAV súbory (na prednáške nám bolo povedané, že sú 16kHz).

Po extrakcii MFCC príznakov pre každý súbor nasledoval krok filtrovania ticha. Tento krok bol realizovaný priamo na MFCC príznakoch. Rámce, ktorých celková energia (prvý MFCC koeficient) bola pod stanoveným prahom, boli považované za ticho alebo šum a boli odstránené.

Následne boli všetky zostávajúce MFCC príznaky zo všetkých trénovacích súborov normalizované pomocou StandardScaler z knižnice scikit-learn. Normalizácia bola potrebná pre GMM modely, lebo sú citlivé na rôzne rozsahy hodnôt vstupných príznakov a pomáha predchádzať dominancii príznakov s vyššou energiou.

### 2.1.2 Konfigurácia modelu a hyperparametre

Pre GMM modely boli zvolené nasledujúce kľúčové parametre:

- Počet Gaussových komponentov (n_components): 8
- Počet EM iterácií (max_iter): 20
- Typ kovariančnej matice (covariance_type): Plná ('full')

Voľba typu kovariancie bola výsledkom experimentovania. Pôvodne testovaný model s diagonálnou kovariančnou maticou dosahoval presnosť okolo 58% a zdalo sa, že ďalšie ladenie hyperparametrov neprináša zlepšenie. Testovanie s plnou kovariančnou maticou však ukázalo konzistentne lepšie výsledky (cca o 5%) pri náhodných hyperparametroch. Ostatné najlepšie hyperparametre sme našli cez gridsearch.

### 2.1.3 Trénovanie a klasifikácia

Pre každú z 31 tried bol natrénovaný samostatný GMM model pomocou implementácie GaussianMixture. Ako trénovacie dáta pre model i-tej triedy boli použité všetky predspracované MFCC vektory extrahované z trénovacích nahrávok osoby. Parametre modelov (váhy zmesí, stredné hodnoty a kovariančné matice) boli odhadnuté iteračným Expectation-Maximization (EM) algoritmom.

Identifikácia neznámej nahrávky prebiehala nasledovne: Z nahrávky boli extrahované, filtrované a normalizované MFCC vektory rovnakým postupom ako pri trénovaní. Následne bola pre každý z 31 natrénovaných GMM modelov vypočítaná priemerná logaritmická vierohodnosť týchto MFCC vektorov pomocou metódy score. Nahrávka bola priradená tej triede (osobe), ktorej GMM model poskytol najvyššiu priemernú log-vierohodnosť. Tieto log-vierohodnosti pre každú triedu boli tiež použité ako požadované skóre pre výstupný súbor.

Výsledná presnosť (Accuracy): 72.58% (na dev/test sade)

# 3.2 Rozpoznávanie obrazu (Image)

Model: Support Vector Machine (SVM) s HOG.

Knižnica: sklearn.svm.SVC, sklearn.model_selection.GridSearchCV, skimage.feature.hog.

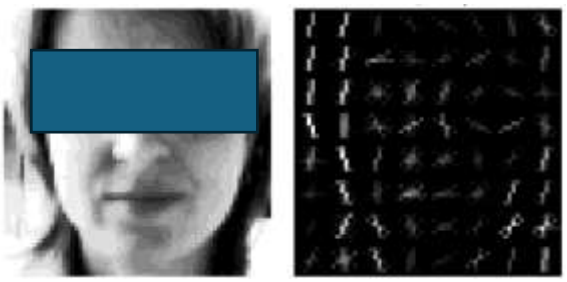
### 3.2.1 Predspracovanie a extrakcia príznakov

Obrazové dáta (PNG súbory) boli načítané knižnicou OpenCV a transformované do odtieňov šedej, keďže HOG pracuje s gradientmi intenzity. Všetky obrázky boli zmenšené na 64×64 pixelov pre konzistentnú dĺžku vektorov a aplikovaná bola globálna ekvalizácia histogramu (cv2.equalizeHist) na normalizáciu kontrastu.

HOG príznaky boli extrahované s parametrami:

- orientations=9
- pixels_per_cell=(8, 8)
- cells_per_block=(2, 2)
- block_norm='L2-Hys'

Výsledné HOG vektory boli normalizované pomocou StandardScaler.



*Obrazok 1Priklad IMG do HOG priznakov*

### 3.2.2 Konfigurácia modelu a hyperparametre

Ako klasifikátor bol použitý SVC (Support Vector Classification) z knižnice scikit-learn. SVM je diskriminatívny klasifikátor hľadajúci optimálnu nadrovinu (hyperplane) oddeľujúcu dátové body rôznych tried s maximálnou možnou rezervou (margin).

Hyperparametre SVM boli optimalizované pomocou GridSearchCV so 5-násobnou krížovou validáciou. Hľadané boli parametre C, kernel a gamma.

Najlepšia konfigurácia: C=0.1, kernel='linear', gamma='scale'.

### 3.2.3 Trénovanie a klasifikácia

Trénovanie prebiehalo už na normalizovaných HOG príznakoch s najlepšími hyperparametrami. Klasifikátor bol súčasťou Pipeline, ktorá zabezpečovala aplikáciu StandardScaler pred samotnou klasifikáciou.

Pre zabezpečenie výstupu logaritmických pravdepodobností, bola pri inicializácii SVC nastavená voľba probability=True. Toto nám pri trénovaní dovolilo použiť metódu predict_log_proba, ktorá vracia odhady  logaritmických pravdepodobností ku každej triede. Tvrdé rozhodnutie bolo získané štandardnou metódou predict.

Výsledná presnosť (Accuracy): 72.58% (na dev sade)

### 3.2.4 Chyby pri obrázkoch

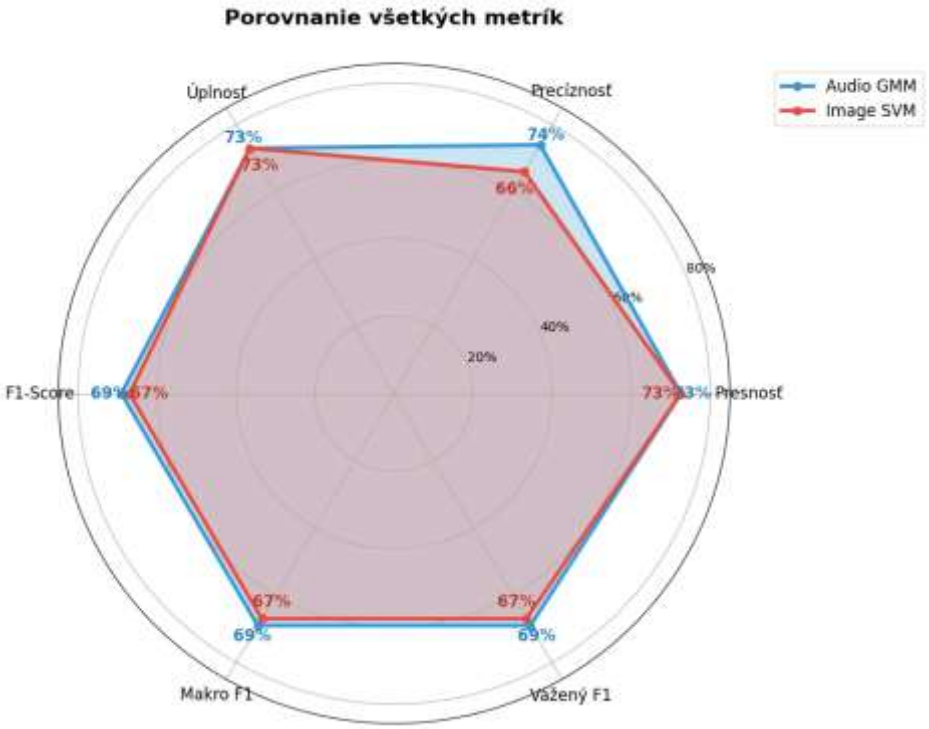
Najprv bol pre rozpoznávanie obrazu vyskúšaný model založený na jednoduchej neurónovej sieti (viacvrstvový perceptrón). Avšak vzhľadom na relatívne malý počet trénovacích dát (186 obrázkov pre 31 tried) tento model nedosahoval dobré výsledky. Aj pri rôznych nastaveniach architektúry, hyperparametrov a augumentacií dát dosahovala neurónová sieť maximálnu presnosť len okolo 30%. Neurónové siete sú známe svojou potrebou veľkého množstva dát aby generalizovali a preto sme sa potom presunuli na model SVM+HOG.

# 4. Výsledky

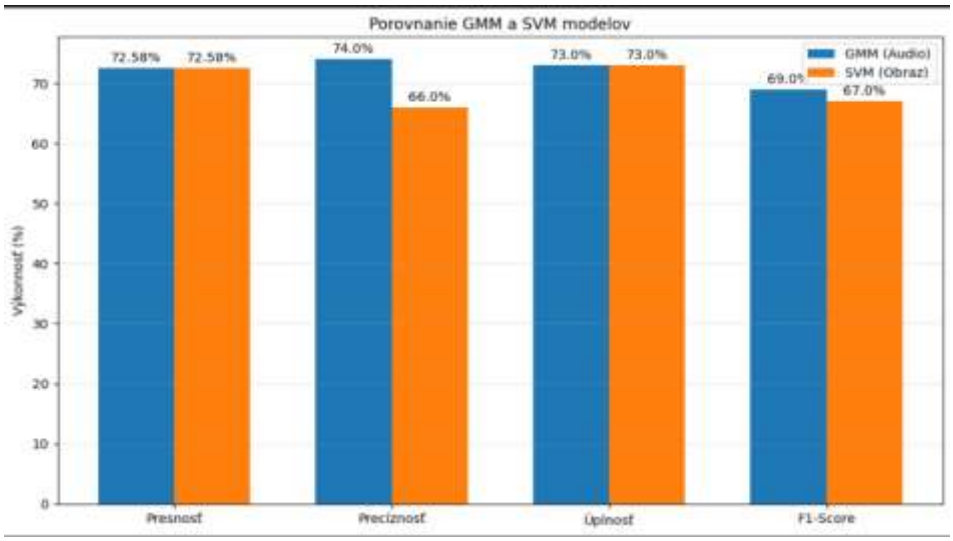
## 4.1 Report

Výstup z classification_report() – scikit-learn funkcia

| Úloha | Model | Presnosť/Accuracy | Precíznosť/Precision | Úplnosť/Recall | F1-Score |
|---|---|---|---|---|---|
| Hlas (AUDIO) | GMM | 72.58% | 0.74 | 0.73 | 0.69 |
| Obraz (IMAGE) | SVM+HOG | 72.58% | 0.66 | 0.73 | 0.67 |

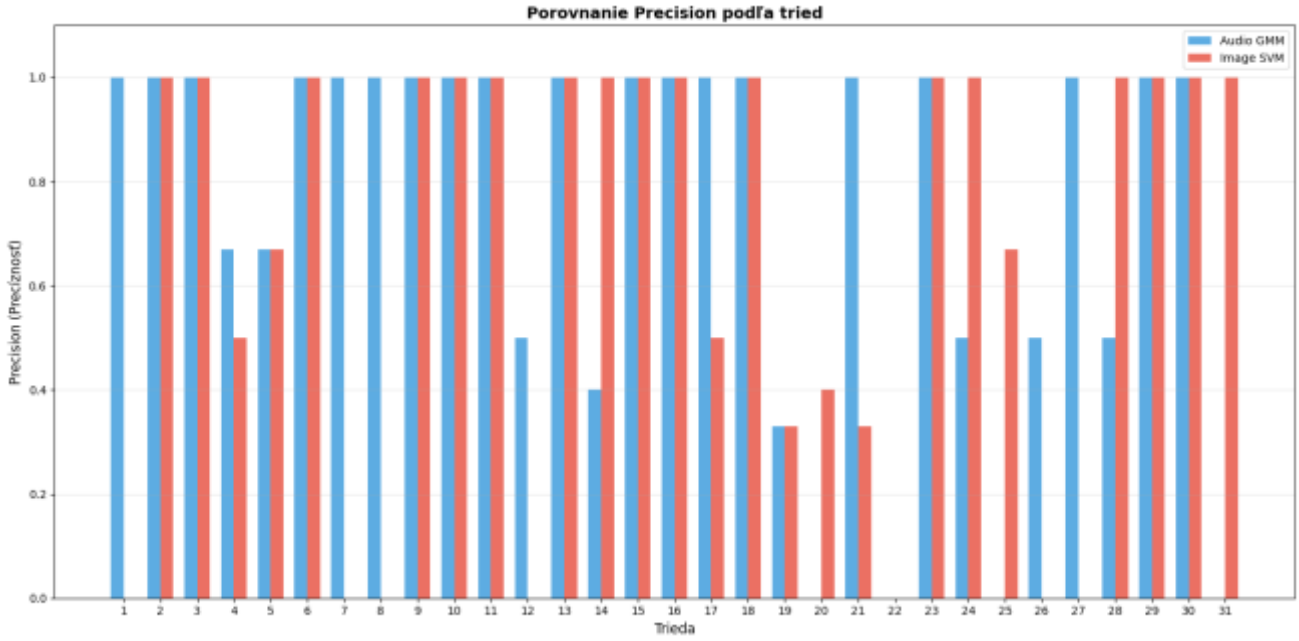



*Obrazok 2 Metriky Porovnanie*




*Obrazok 1Metriky porovnanie*

## 4.2 Porovnanie výkonu po triedach




*Obrazok 3 Porovnanie medzi triedami(precision)*

**Triedy kde Audio-GMM je lepšie(PrecisionImage<PrecisionAudio):**

- 1, 7, 8, 12, 26,21, 31
- Celkovo je lepší v 7 triedach.

**Triedy kde Image-SVM je lepšie (PrecisionImage>PrecisionAudio):**

- Výrazne lepšie: 20, 22, 25,4, 14, 19
- Celkovo je lepší v 6 triedach.

**Rovnaké výsledky (Precision = 1.00):**

- 2, 3, 6, 9, 10, 11, 13, 15, 16, 18, 27, 28, 29, 30
- Celkovo 14 tried.

**Trieda, kde ani jeden model není správny:**

- 22 (Precision = 0.00)
- Celkovo 1 trieda

# 5. Záver

Podobná presnosť- výkonnosť pri GMM+MFCC (audio) a SVM+HOG (obraz). Presnosť je presne 72.58%, čo ukazuje dobrú klasifikáciu, lebo ak by sme tipovali, mali by sme presnosť 3% (100/31). Vyššia precision GMM naznačuje, že robí menej falošných pozitívnych chýb v porovnaní so SVM na tomto datasete, tj. GMM je spolahlivejší.

V budúcnosti by som sa rád pozrel viac do hĺbky, prečo trieda 22 nie je ani pri jednom modeli správna. Zároveň vďaka tomu, že iba pri jednej triede nie je vôbec presný ani jeden model, môžeme implementovať kombináciu týchto dvoch modelov a predikovať ešte presnejšie klasifikácie osôb. Ďalej by som rád preskúmal vplyv postupu filtrovania ticha - či je efektívnejšie filtrovať tichšie segmenty priamo z extrahovaných MFCC príznakov, alebo eliminovať ticho z audio signálu ešte pred MFCC spracovaním a porovnať presnosti GMM modelov.