Dokumentácia k projektu SUR

Autor: Jakub Magda <xmagda03>

1. Úvod

Cieľom projektu je vytvoriť systém pre identifikáciu 31 osôb na základe kombinácie hlasových nahrávok a obrázkov tváre. Systém využíva dva nezávislé modely – pre audio a obrazovú modálnosť – a kombinuje ich výsledky pomocou váženej fúzie. Hlavným prínosom je zvýšenie presnosti oproti používaniu jednej modality. Táto dokumentácia podrobne vysvetľuje implementáciu, trénovanie, vyhodnotenie a spôsob spustenia systému.

---

2. Požiadavky na softvér a inštalácia

Na správne fungovanie systému je potrebný aspoň **Python 3.8** alebo novší a nasledujúce knižnice:

- numpy, scikit-learn – pre strojové učenie a lineárnu algebru,
- librosa, python_speech_features – extrakcia audio features,
- opencv-python – spracovanie obrazu,
- imbalanced-learn – vyváženie tried (SMOTE),
- scikit-image, scipy – augmentácia a transformácie obrazu.

Inštalácia sa vykoná pomocou príkazu:

```
pip install numpy scikit-learn librosa opencv-python python-speech-features imbalanced-learn tqdm scikit-image scipy
```

3. Popis riešenia

3.1 Spracovanie audia

Extrakcia features

Pre audio signál sa extrahuje široké spektrum príznakov:

- **MFCC (Mel-Frequency Cepstral Coefficients)** s 40 koeficientmi, doplnené o delta a delta-delta príznaky pre zachytenie dynamických zmien.
- **Štatistiky spektra** (centroid, šírka pásma, kontrast) a časové charakteristiky (počet prechodov cez nulu, tempo).
- **Chroma features** pre tonalitu a harmonické vlastnosti.

Vo fáze trénovania sa aplikuje **augmentácia dát** pridávaním šumu (±5 % amplitúdy), modifikáciou výšky tónu (±1 poltón) a časovým natiahnutím (±10 %). Tieto techniky zvyšujú robustnosť modelu voči šumu a variabilite v nahrávacich podmienkach.

Klasifikátor
Ako klasifikátor bol zvolený **SVM (Support Vector Machine)** s jadrom RBF. Experimentálne sa overilo, že SVM poskytuje lepšiu separáciu tried ako GMM (Gaussian Mixture Models), najmä v kombinácii s vysokodimenzionálnymi príznakmi. Parametre C=10 a gamma='scale' boli zvolené na základe validácie: vyššia hodnota C redukuje underfitting, zatiaľ čo gamma='scale' automaticky prispôsobuje váhu príznakov.

Pre redukciu dimenzie sa použilo **PCA** (zachovanie 95 % variability dát) a na vyváženie tried **SMOTE** (Synthetic Minority Oversampling Technique), ktorý synteticky generuje vzorky pre menej početné triedy.

3.2 Spracovanie obrazu

Predspracovanie
Obrázky sa normalizujú na veľkosť **80×80 pixelov** pre jednotný vstup do modelu. Nasleduje zvýšenie kontrastu pomocou **CLAHE (Contrast Limited Adaptive Histogram Equalization)** a redukcia šumu Gaussovým filtrom.

Extrakcia features

- **HOG (Histogram of Oriented Gradients)** s parametrami: 9 orientácií, 8×8 pixelov na bunku, 2×2 bunky na blok. HOG zachytáva tvarové vlastnosti tváre.
- **LBP (Local Binary Patterns)** s 8 bodmi a polomerom 1 px. LBP popisuje textúru pokožky a detaily ako vrásky alebo očné okolí.

Augmentácia
Pre zvýšenie generalizácie sa aplikuje:

- **Rotácia** (±10°), horizontálne preklopenie,
- Zmena jasu (±20 %) a kontrastu,
- Náhodné orezanie a natiahnutie na pôvodnú veľkosť.

Klasifikátor
Aj pre obraz bol použitý **SVM s jadrom RBF** (C=10). Výber bol motivovaný schopnosťou SVM pracovať s vysokodimenzionálnymi príznakmi (kombinácia HOG a LBP má cez 1 000 dimenzií) a odolnosťou voči pretrénovaniu.

3.3 Kombinovaný model (fúzia)

Výstupy oboch modelov sa kombinujú pomocou **váženého súčtu log-pravdepodobností**. Predvolené váhy sú 0.5:0.5, ale optimalizáciou na validačnej množine sa zistilo, že váhy 0.6 pre audio a 0.4 pre obraz poskytujú vyššiu presnosť. Dôvodom je, že audio model dosahuje stabilnejšie výsledky pri variáciách v osvetlení alebo póze tváre.

4. Trénovanie a vyhodnotenie

4.1 Trénovacie postupy

Trénovanie prebiehalo na poskytnutých dátach. Keďže počet dát na triedu bol malý a počet tried veľký, použil som augmentáciu dát. Rozloženie dát v datasetoch `dev` a `train` som nijak nemenil. Počas trénovania som skúšal rôzne trénovacie algoritmy ako som už spomínal. Rovnako som aj menil jednotlivé parametre klasifikátorov a hľadal som najoptimálnejšie riešenie aby som sa zároveň vyhol overfittingu. Audio model pôvodne fungoval s GMM klasifikátorom ale SVM sa ukázal ako lepší.

- **Audio model** sa trénuje príkazom:

```
python audio_model.py --mode train --data_path cesta_k_datam
--output_dir models/audio_svm --use_augmentation
```

  Augmentácia zvýši počet trénovacích vzoriek 4-násobne.

- **Obrazový model** sa spustí príkazom:

```
python image_model.py --mode train --data_path cesta_k_datam --
output_dir models/image_svm --augment
```

  Augmentácia generuje 10 variantov pre každý obrázok. Počet variantov môže byť aj nastaviteľný príkazom

4.2 Metriky úspešnosti

- **Audio model** dosiahol presnosť **83 %** na validačnej množine. Kľúčovým faktorom bola kombinácia MFCC s delta príznakmi a augmentácia.
- **Obrazový model** dosiahol presnosť **78 %**. Nižšia hodnota oproti audiu je spôsobená variabilitou v pózach a osvetlení.
- **Kombinovaný model** dosiahol presnosť **93 %** pri váhach 0.7:0.3, čo potvrdzuje synergický efekt fúzie.

4.3 Kritické rozhodnutia

- **Výnimka orezávania tichých pasáží**: Pokusy s orezávaním ticha v audio signáli viedli k strate kontextových informácií (napr. pauzy medzi slovami), čo znížilo presnosť. Preto bola táto technika vynechaná.
- **Voľba SVM namiesto neurónových sietí**: Kvôli obmedzeniam zadania (zákaz použitia predtrénovaných modelov) boli preferované klasické metódy. Vyskúšal som aj trénovať modelpomocou neurónových sietí ale ani augmentácia dát nepomohla a presnosť modelu bola len niečo okolo 4% čo je prakticky presnosť hádania.

---

5. Návod na použitie

5.1 Spustenie na testovacích dátach

- Pre audio:

```
python audio_model.py --mode test --test_dir eval/ --output_file
audio_results.txt
```

- Pre obraz:

```
python image_model.py --mode test --test_dir eval/ --output_file
image_results.txt
```

- Pre fúziu:

```
python fusion.py --test_dir eval/ --output_file fusion_results.txt
--weights "0.6,0.4"
```

- Pre natrénovanie a evaluáciu všetkých modulov je aj skript **`run_fusion.py`** , ktorý natrénuje všetky modely a evaluačne vyhodnotí:

```
python run_fusion.py --test_dir eval/  --weights 0.5,0.5
```

6. Záver

Systém úspešne kombinuje audio a obrazové dáta s presnosťou nad 90 %. Hlavnými výhodami sú robustnosť voči šumu a variabilite osvetlenia. Medzi obmedzenia patrí závislosť od kvality vstupných dát – napríklad veľké prekrytie tváre v obraze môže znížiť presnosť.