

Wyszukiwanie osób na podstawie portretu pamięciowego

Jakub Grzana 241530
Małgorzata Pietras 235794

Projekt zespołowy
03.06.2020

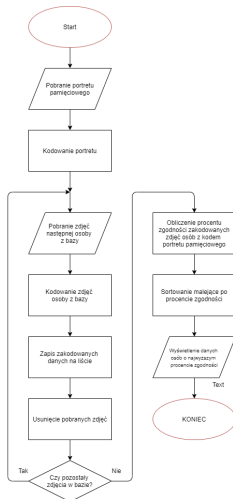
Plan

- 1 Opis projektu
- 2 Rozpoznawanie twarzy
 - Biblioteka face-recognition
 - Program encodings.py
 - Program calculations.py
- 3 Komunikacja Klient-Serwer
 - Aplikacja kliencka
 - Aplikacja serwera
 - Struktura bazy danych
 - Przykład działania oprogramowania

Opis projektu

Celem projektu było stworzenie oprogramowania rozpoznającego osobę na podstawie danych wejściowych w formie zdjęcia lub portretu pamięciowego. Stworzono do tego celu aplikację umożliwiającą komunikację Klient-Serwer. Klient wysyła do Serwera zdjęcie poszukiwanej osoby i odbiera wyniki obliczeń. Wyniki te przedstawione są jako lista osób o największej procentowej zgodności.

Diagram blokowy systemu



Plan

- 1 Opis projektu
- 2 Rozpoznawanie twarzy
 - Biblioteka face-recognition
 - Program encodings.py
 - Program calculations.py
- 3 Komunikacja Klient-Serwer
 - Aplikacja kliencka
 - Aplikacja serwera
 - Struktura bazy danych
 - Przykład działania oprogramowania

face-recognition

- Kod napisany Pythonie
- Wykorzystuje uczenie maszynowe
- Skuteczność rozpoznawania to 99.38%
- Umożliwia rozpoznawanie wielu twarzy
- Poprawnie odnajduje oczy, usta, nos i brwi
- Jest nawet w stanie dorysować makijaż

Rozpoznawanie twarzy



Input



Output

Odnajdywanie oczu, ust, nosa i brwi



Input



Output

Sztuczny makijaż



Input



Output

Plan

- 1 Opis projektu
- 2 Rozpoznawanie twarzy
 - Biblioteka face-recognition
 - Program encodings.py
 - Program calculations.py
- 3 Komunikacja Klient-Serwer
 - Aplikacja kliencka
 - Aplikacja serwera
 - Struktura bazy danych
 - Przykład działania oprogramowania

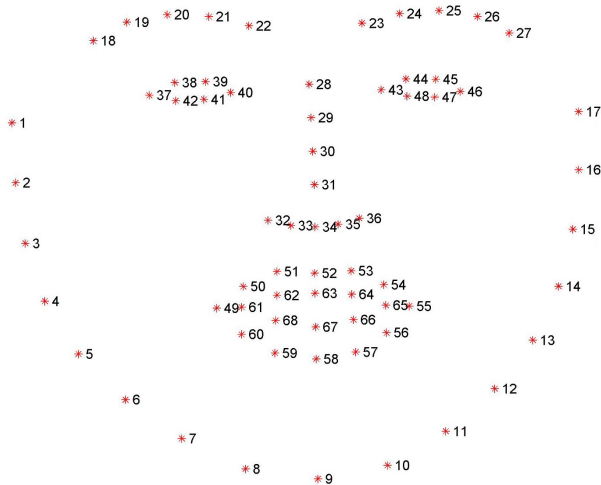
Program encodings.py

Program tworzący plik z kodowaniem twarzy znajdujących się w bazie danych. Na twarzy odnajdywanych jest 68 charakterystycznych punktów.

Wykorzystane funkcje:

- `face_recognition.face_locations`
- `face_recognition.face_encodings`

Kodowanie twarzy



Plan

- 1 Opis projektu
- 2 Rozpoznawanie twarzy
 - Biblioteka face-recognition
 - Program encodings.py
 - Program calculations.py
- 3 Komunikacja Klient-Serwer
 - Aplikacja kliencka
 - Aplikacja serwera
 - Struktura bazy danych
 - Przykład działania oprogramowania

Program calculations.py

Program obliczający podobieństwo danej osoby z bazy danych do portretu pamięciowego. Odległości pomiędzy odpowiadającymi sobie punktami wyznaczone są za pomocą metryki euklidesowej. Wynik zwracany jest w postaci liczby z przedziału 0-1. Mniejsza wartość oznacza większe podobieństwo. Wiarygodne wyniki powinny być mniejsze od 0,6.

Wykorzystane funkcje:

- `face_recognition.face_locations`
- `face_recognition.face_encodings`
- `face_recognition.face_distance`

Program calculations.py

```
|4  
0.5464801142643373  
nicolas_cage  
0.5497709425715045  
nicolas_cage  
0.6085810781959743  
nicolas_cage  
0.6624370941773301  
nicolas_cage  
0.7342964630688374  
harrison_ford  
0.7696486356107545  
harrison_ford  
0.7872786177748673  
harrison_ford  
0.84429130847974  
eddie_murphy  
0.8461649578705479  
eddie_murphy  
0.8791909325111716  
eddie_murphy  
0.8794203683579142  
angelina_jolie  
0.9151688864174464  
angelina_jolie  
0.9376839179730859  
angelina_jolie  
0.9598438280092176  
angelina_jolie
```

Plan

- 1 Opis projektu
- 2 Rozpoznawanie twarzy
 - Biblioteka face-recognition
 - Program encodings.py
 - Program calculations.py
- 3 **Komunikacja Klient-Serwer**
 - **Aplikacja kliencka**
 - Aplikacja serwera
 - Struktura bazy danych
 - Przykład działania oprogramowania

Aplikacja kliencka

- Napisana z użyciem SFMLa
- Umożliwia wybór IP serwera oraz pliku graficznego do przesłania
- Aplikacja nawiązuje połączenie na określonych w kodzie portach, przesyła plik, odbiera odpowiedź i wyświetla ją na ekranie

Plan

- 1 Opis projektu
- 2 Rozpoznawanie twarzy
 - Biblioteka face-recognition
 - Program encodings.py
 - Program calculations.py
- 3 **Komunikacja Klient-Serwer**
 - Aplikacja kliencka
 - **Aplikacja serwera**
 - Struktura bazy danych
 - Przykład działania oprogramowania

Aplikacja serwera

- Napisana z użyciem SFMLa i wątków Pthread
- Klient łączy się na uniwersalnym porcie (określonym w kodzie). Zostaje mu przydzielony wątek oraz indywidualny port
- Serwer wysyła ten port do klienta, po czym zrywa połączenie
- Następnie klient łączy się z serwerem na przydzielonym porcie
- Klient wysyła portret pamięciowy do serwera. Python wykonuje obliczenia, następnie serwer robi listę najlepszego dopasowania i wysyła ją oraz pliki info.txt do klienta

Plan

- 1 Opis projektu
- 2 Rozpoznawanie twarzy
 - Biblioteka face-recognition
 - Program encodings.py
 - Program calculations.py
- 3 Komunikacja Klient-Serwer**
 - Aplikacja kliencka
 - Aplikacja serwera
 - Struktura bazy danych**
 - Przykład działania oprogramowania

Struktura bazy danych

- Osobny katalog dla każdej osoby
- W katalogach znajdują się:
 - zdjęcia w formacie .jpg
 - pliki info.txt, zawierające:
 - imię i nazwisko
 - numer PESEL
 - inne istotne informacje

Plan

- 1 Opis projektu
- 2 Rozpoznawanie twarzy
 - Biblioteka face-recognition
 - Program encodings.py
 - Program calculations.py
- 3 Komunikacja Klient-Serwer
 - Aplikacja kliencka
 - Aplikacja serwera
 - Struktura bazy danych
 - Przykład działania oprogramowania

Baza danych

Branch: master ▾ projekt_zespolowy / SerwerKlient / Serwer / BazaDanych /

Create new fileUpload filesFind fileHistory

 AzethMeron Ostateczna wersja view

Latest commit 82bb412 11 days ago

..

 angelina_jolie Ostateczna wersja 11 days ago

 eddie_murphy Ostateczna wersja 11 days ago

 harrison_ford Ostateczna wersja 11 days ago

 nicolas_cage Ostateczna wersja 11 days ago

 info.txt Komunikacja ukonczona, załączek współpracy z PY last month



info.txt



nc1.jpg



nc2.jpg



nc3.jpg




nc4.jpg

Działanie programów

```
azethmeron@AMPenLinux:~/Dokumenty/Studia/Semestr 6/Projekt Zespoło  
wy/PZSerwer/Klient$ ./Klient 192.168.43.86 test.jpg  
Błąd połączenia z portem: 32001  
Probuje ponownie  
Połączenie z portem: 32001 udane  
Wyniki. Im liczba mniejsza tym lepiej  
: 0.586229 Nicolas Cage 1234567  
: 0.763741 Harrison Ford 1234567  
azethmeron@AMPenLinux:~/Dokumenty/Studia/Semestr 6/Projekt Zespoło  
wy/PZSerwer/Klient$
```

test.jpg

Plik Edycja Widok Przejdź Pomoc



```
azethmeron@AMPenLinux:~/Dokumenty/Studia/Semestr 6/Projekt Zespoło  
wy/PZSerwer/Serwer$ ./Serwer  
Pobieranie adresu IP...  
Pobrano adres IP.  
Adres IP serwera (globalny): 31.60.230.57  
Adres IP serwera (lokalny): 192.168.43.86  
Port do łączenia się z programem: 32000  
Oczekiwanie na klienta...  
Przybył klient. Przydzielono port: 32001  
Oczekiwanie na klienta...  
PWR W4 Projekt zespółowy 2020  
M.Pietras, J.Grzana  
Program obliczający podobieństwo twarzy  
0.0 - całkowite podobieństwo  
100.0 - brak podobieństwa  
Na podstawie programów P.Ryana
```