

Politechnika Gdańska
Wydział Elektrotechniki i Automatyki
Katedra Elektrotechniki, Systemów Sterowania i Informatyki

Monitorowanie i diagnostyka w systemach sterowania

Rozpoznawanie twarzy: Metoda Eigenfaces oraz sieci neuronowe

Zadania do ćwiczeń laboratoryjnych – zajęcia nr 8 - 9

Opracowanie:

dr inż. Michał Grochowski

Zadanie

Celem laboratorium jest ekstrakcja i zaimplementowanie algorytmu rozpoznawania twarzy Eigenfaces, z klasyfikatorem neuronowym.

Należy:

1. Zaimplementować algorytm rozpoznawania twarzy Eigenfaces.
2. W ramach algorytmu wykonaj również rekonstrukcję zdjęć przy pomocy wybranych twarzy własnych.
3. Jako klasyfikatora należy użyć metody najbliższego sąsiada (odległość Euclidesowa) a następnie sieci neuronowych.
4. Stworzyć własną bazę danych zdjęć przy pomocy stanowiska laboratoryjnego. Początkowo można skorzystać z bazy zdjęć umieszczonej w katalogu:
Pulpit/SIW_laboratorium/Eigenfaces/_Dane.
5. Przetestować jakość działania algorytmu dla:
 - różnych ilości twarzy własnych;
 - zdjęć wykonywanych w różnych warunkach oświetleniowych;
 - zdjęć „bliskich” i „odległych” od zdjęć uczących (ale pochodzących od tych samych osób);
 - klasyfikatora Euclidesowego oraz neuronowego.

Uwagi:

Program umożliwiający tworzenie bazy zdjęć Database.exe znajduje się w:
Pulpit/SIW_laboratorium/Eigenfaces/Testowanie.

Przydatne komendy Matlaba:

- wczytywanie obrazu: `imread`;
- zapisywanie macierzy w postaci kolumny: `A(:)`;
- „formatowanie” macierzy do dowolnych rozmiarów: `reshape`;
- konwersja obrazu do skali szarości: `rgb2gray`;
- Pobranie przykładowego zdjęcia w celu pobrania jego wymiarów:
 - `[Height Width Deep] = size(imread('Dane/osoba_1.png'));`
 - Wyświetlanie zdjęcia: `colormap(gray); imagesc(przykladowe_zdjecie);`
- Zachowanie proporcji wyświetlanego zdjęcia: `axis image`;
- Odległość dwóch wektorów: `pdist`;
- Sieć neuronowa, (oprócz wielu innych) np.: `newff`

Baza zdjęć oraz program do jej ich gromadzenia pochodzą z: Błaszowski P. Wykrywanie, rozpoznawanie i śledzenie ruchomych obiektów poprzez niezależną platformę monitorującą, przy wykorzystaniu sztucznych sieci neuronowych. Praca magisterska, Politechnika Gdańska 2013. Promotor dr inż. Michał Grochowski.

Przykładowy „szkielet” programu:

```
% ===== Ustawianie parametrów =====
% Liczba twarzy uczacych:
LiczbaTwarzyUczacych=xxx;
% Liczba osób:
LiczbaOsob = xxx;
% Liczba twarzy testowych:
LiczbaTwarzyttestowych = xxx;
% Liczba EigenFaces:
LiczbaEigenFaces = xxx;
% Imiona twarzy uczacych:
ImionaUczace = {'Rysiek', 'Czesiek', 'Zofia', 'Heniek',..., };

% Przypisanie zdjęć osobom
for i=1: LiczbaTwarzyUczacych
    Osoby(1,i)=floor((i-1)* LiczbaOsob/LiczbaTwarzyUczacych)+1;
end
% Pobranie przykładowego zdjęcia w celu pobrania wymiarów
[Height Width Deep] = size(imread('Dane/osoba_1.png'));

% wczytywanie kolejnych zdjęć uczących

for i=1: LiczbaTwarzyUczacych
    %pobranie zdjęcia z pliku
    x = imread(sprintf('Uczace/uczace_%d.png',i));
    %skalowanie do odcieni szarosci w double (można użyć rgb2gray)
    LearnFaces{i}=double(((0.299*x(:, :, 1)+0.587*x(:, :, 2)+0.114*x(:, :, 3))/3))/255;
end

% ===== Wyznaczanie modelu Eigen Faces =====
% Algorytm właściwy
% Wykresy
% Rekonstrukcja przykładowego zdjęcia

% ===== Testowanie zdjęć =====
% Klasyfikator Euclidesowy
% Klasyfikator Neuronowy
Budowa sieci
Uczenie
Testowanie

% ===== Statystyki jakości rozpoznawania twarzy ===== %
Klasyfikator Euclidesowy % Klasyfikator Neuronowy
```