

Nume: Toader Sergiu-Cristian

Grupa: 315CB

Cerinta 1:

Se citeste imaginea si se retine intr-o matrice. Folosind functia Octave svd, se realizeaza descompunerea valorilor singulare pe matricea A. Se determina matricele U_k , S_k , V_k , dupa care, prin inmultirea lor se determina matricea A_k .

Cerinta 3:

Se citeste imaginea si se retine in matricea A. Se calculeaza medie pe fiecare vector din matrice si se actualizeaza fiecare vector ca fiind diferenta dintre el insusi si vectorul μ . Se calculeaza matricea Z si apoi se aplica descompunerea pe ea. Se determina matricea W care contine primii k vectori proprii din V. Se calculeaza proiectia lui A in spatiul componentelor principale. In final, se aproximeaza matricea A_k ca fiind $W*Y+\mu$.

Cerinta 4:

Se citeste imaginea si se retine in matricea A. Se calculeaza medie pe fiecare vector din matrice si se actualizeaza fiecare vector ca fiind diferenta dintre el insusi si vectorul μ . Se calculeaza matricea de covarianta Z si apoi se aplica descompunerea folosind functia Octave eig. Se determina matricea W care contine primii k vectori proprii din V. Se calculeaza proiectia lui A in spatiul componentelor principale. In final, se aproximeaza matricea A_k ca fiind $W*Y+\mu$.

Cerinta 6:

(In functia eigenface_core): Fiecare imagine este citita intr-o matrice, matricea respectiva fiind convertita intr-un vector. Cele 10 coloane din matricea A reprezinta cei 10 vectori determinati de valorile din matricea imaginilor. Se calculeaza media vectorilor din A si se actualizeaza vectorii, scazandu-se din ei media. Se calculeaza vectorii si valorile proprii ale matricei $A'*A$ folosind eig.

Vectorii pentru care valorile proprii sunt mai mici decat 1 sunt eliminati, dupa care se calculeaza spatiul fetelor si proiectia fiecarei imagini in spatiul fetelor.

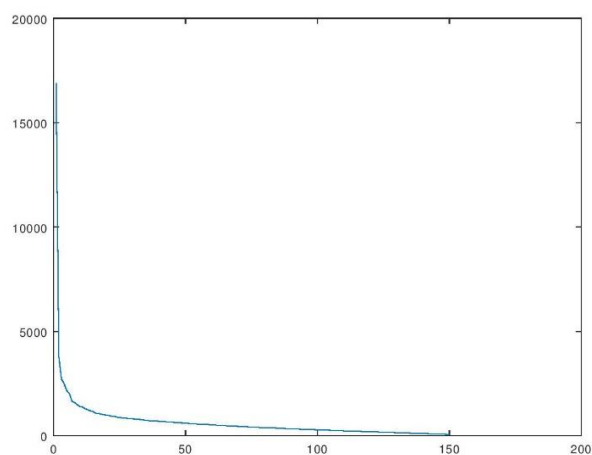
(In functia `face_recognition`): Se citește imaginea de test, se transformă în vector și se actualizează vectorul scăzându-se din el media obținută în funcția `eigenface_core`. Se calculează proiectia imaginii de test în spațiul fetelor, după care se calculează distanța minimă dintre imaginea de test și celelalte imagini folosind funcția `norm`, determinându-se astfel imaginea cea mai apropiată de cea de test:

Grafice cerinta 2 si 5:

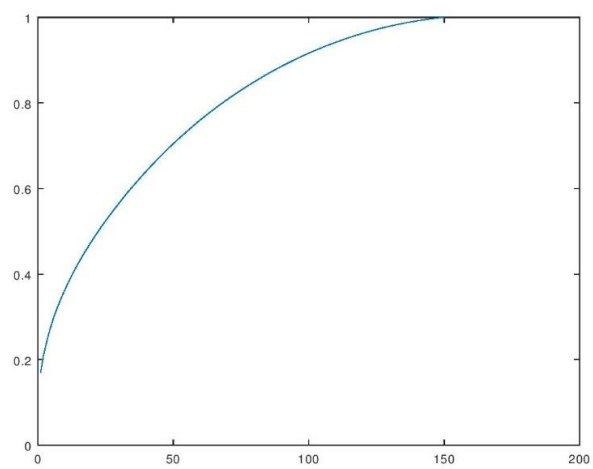
Observatie: k reprezinta un vector de valori (functia realizeaza graficele de mai jos cu apeluri de forma `task2("./in/images/image1.gif", 1:200)`;

Cerinta 2, imagine 2, $k = 1 : 150$

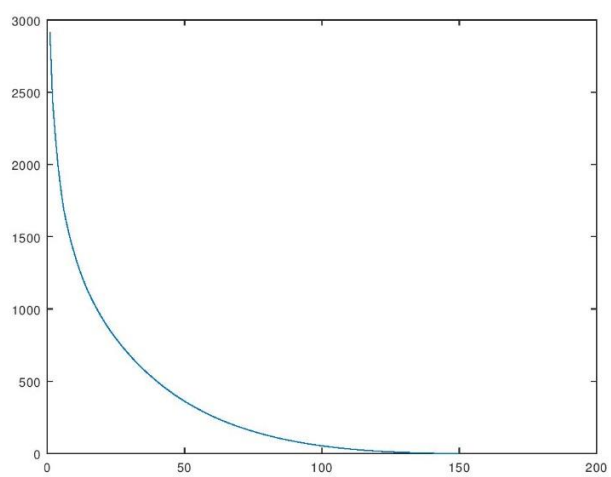
Grafic 1:



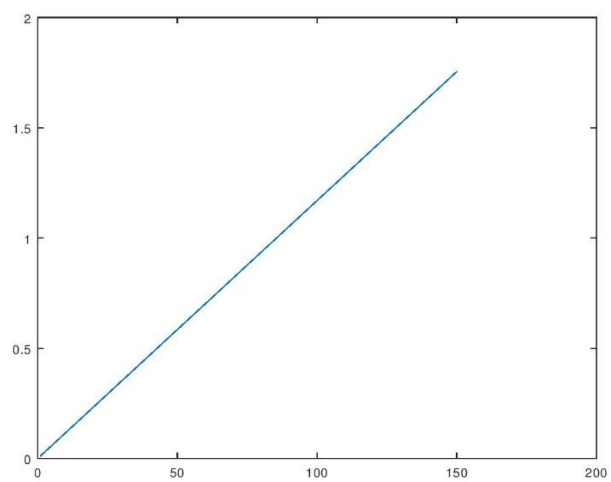
Grafic 2:



Grafic 3:

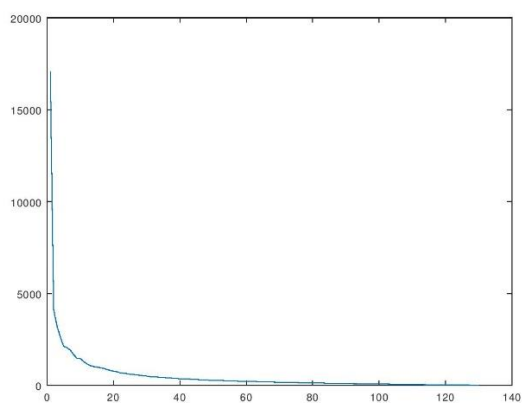


Grafic 4:

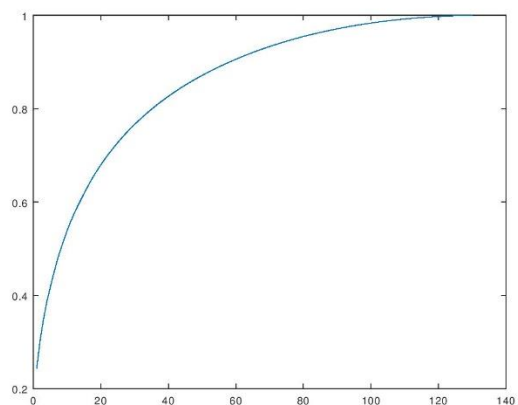


Cerinta 2, imagine 3, k = 1 : 130

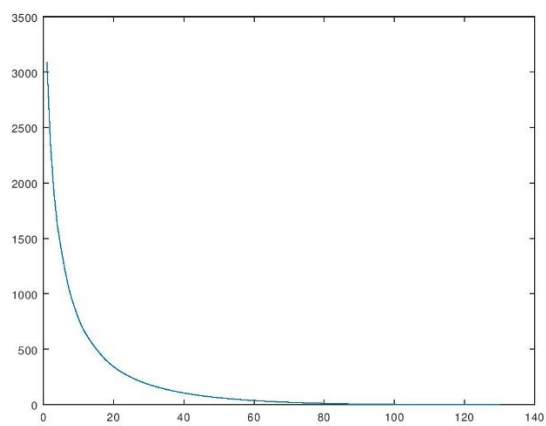
Grafic 1:



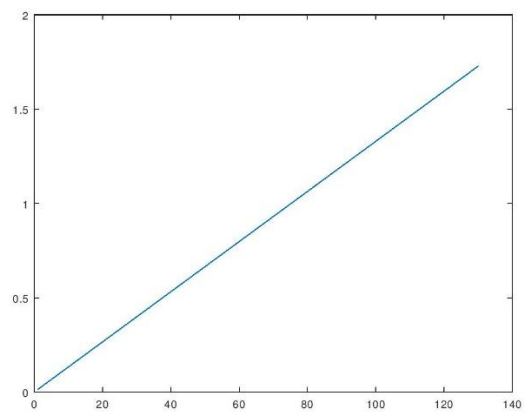
Grafic 2:



Grafic 3:

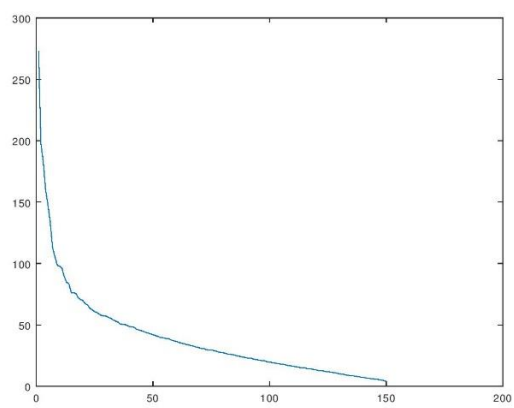


Grafic 4:

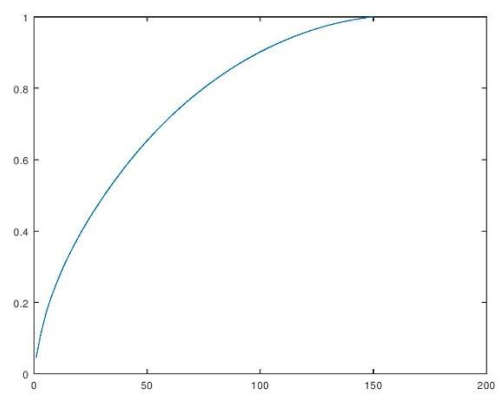


Cerinta 5, imagine 2, k = 1 : 150

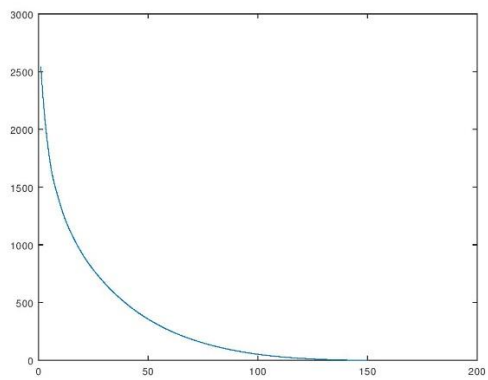
Grafic 1:



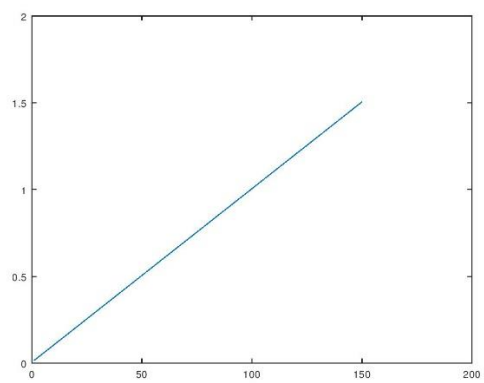
Grafic 2:



Grafic 3:

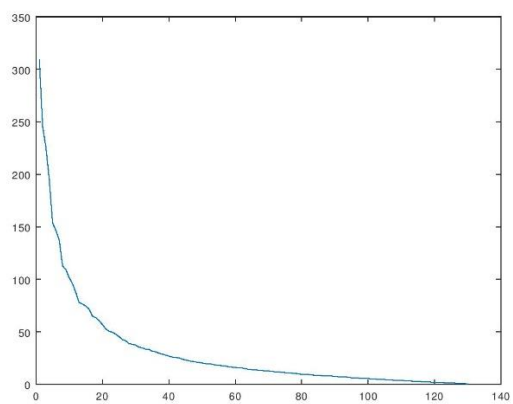


Grafic 4:

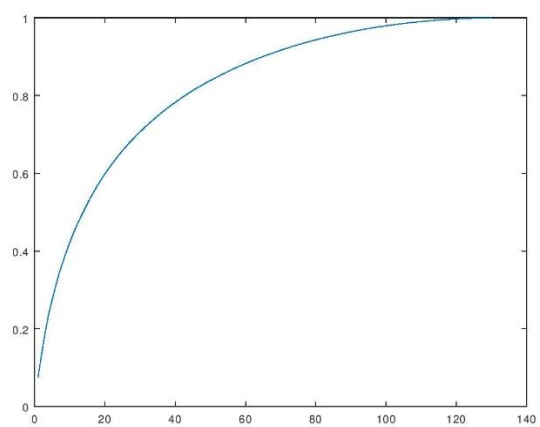


Cerinta 5, imagine 3, k = 1 : 130

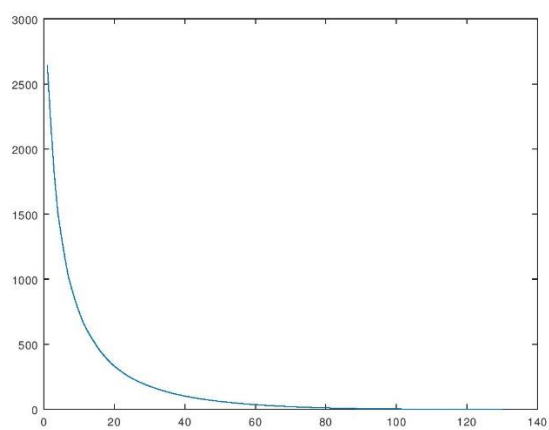
Grafic 1:



Grafic 2:



Grafic 3:



Grafic 4:

