

**Tugas Besar 2 IF 2123 Aljabar Linear Dan Geometri**

# **Aplikasi Nilai Eigen dan EigenFace pada Pengenalan Wajah (Face Recognition)**

**Semester I Tahun 2022/2023**



**Kelompok Keos Recognition:**

**Rachel Gabriela Chen      13521044**

**Muhammad Fadhil Amri      13521066**

**Akbar Maulana Ridho      13521093**

**PROGRAM STUDI TEKNIK INFORMATIKA  
SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA  
INSTITUT TEKNOLOGI BANDUNG**

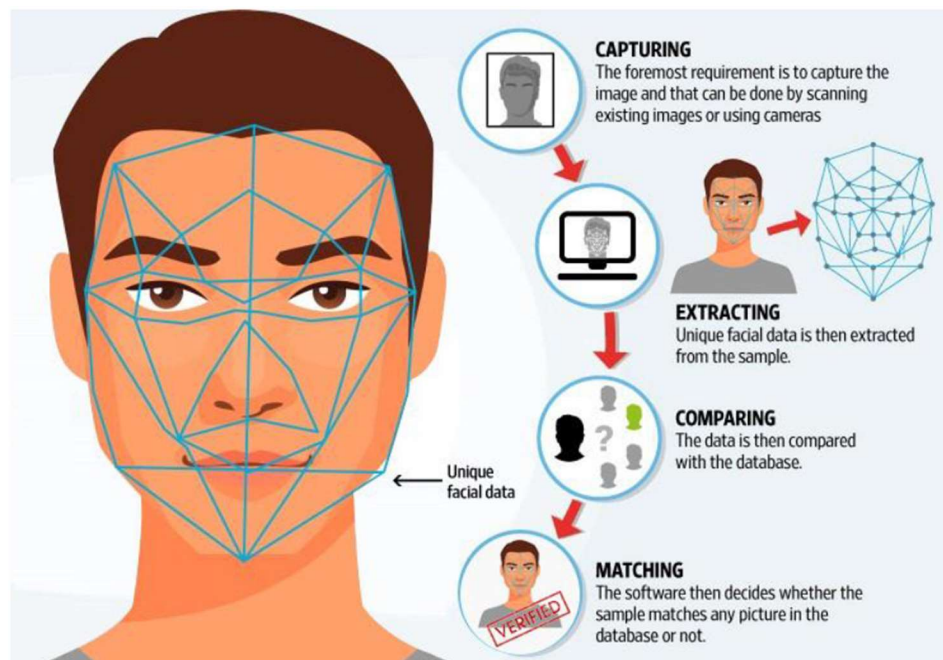
**2022**

## DAFTAR ISI

DAFTAR ISI.....	2
BAB I : DESKRIPSI MASALAH.....	3
BAB II: Teori Singkat.....	4
A. Perkalian Matriks .....	4
B. Nilai Eigen dan Vektor Eigen .....	5
BAB III : Impelementasi Pustaka .....	9
A. Lib .....	9
B. Main.....	12
BAB IV : Eksperimen.....	13
A. Eigenvalue dan Eigenvector.....	13
B. Data Test.....	13
BAB V : Kesimpulan, Saran dan Refleksi.....	24

## BAB I : DESKRIPSI MASALAH

Pengenalan wajah (Face Recognition) adalah teknologi biometrik yang bisa dipakai untuk mengidentifikasi wajah seseorang untuk berbagai kepentingan khususnya keamanan. Program pengenalan wajah melibatkan kumpulan citra wajah yang sudah disimpan pada database lalu berdasarkan kumpulan citra wajah tersebut, program dapat mempelajari bentuk wajah lalu mencocokkan antara kumpulan citra wajah yang sudah dipelajari dengan citra yang akan diidentifikasi. Alur proses sebuah sistem pengenalan wajah diperlihatkan pada Gambar 1.



Gambar 1. Mekanisme Face Recognition

Terdapat berbagai teknik untuk memeriksa citra wajah dari kumpulan citra yang sudah diketahui seperti jarak Euclidean dan cosine similarity, principal component analysis (PCA), serta Eigenface. Pada Tugas ini, akan dibuat sebuah program pengenalan wajah menggunakan Eigenface.

## BAB II: Teori Singkat

### A. Perkalian Matriks

Perkalian matriks adalah operasi pada dua buah matriks A dan B dimana jumlah kolom A harus sama dengan jumlah baris B. Jika A adalah matriks berukuran  $m \times n$  dan B adalah matriks berukuran  $n \times p$ , dengan elemen-elemen sebagai berikut:

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1p} \\ b_{21} & b_{22} & \cdots & b_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{np} \end{pmatrix}$$

Gambar 2. Matriks A Ukuran  $m \times n$  dan Matriks B ukuran  $n \times p$

Hasil perkalian kedua matriks tersebut dituliskan sebagai  $\mathbf{C} = \mathbf{AB}$  dimana C adalah sebuah matriks berukuran  $m \times p$ . Maka, matriks C:

$$\mathbf{C} = \begin{pmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{m1} & c_{m2} & \cdots & c_{mp} \end{pmatrix}$$

Gambar 3. Matriks C Hasil Perkalian Matriks A x Matriks B

, dimana setiap entri pada matriks C didefinisikan sebagai:

$$c_{ij} = a_{i1}b_{1j} + \cdots + a_{in}b_{nj} = \sum_{k=1}^n a_{ik}b_{kj},$$

Gambar 4. Entri Matriks C

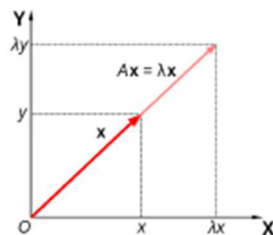
## B. Nilai Eigen dan Vektor Eigen

Jika  $A$  adalah matriks  $n \times n$ , maka vektor tidak-nol  $\mathbf{x}$  di  $\mathbb{R}^n$  disebut vektor eigen dari  $A$  jika  $A\mathbf{x}$  sama dengan perkalian suatu skalar  $\lambda$  dengan  $\mathbf{x}$ , yaitu

$$A\mathbf{x} = \lambda\mathbf{x}$$

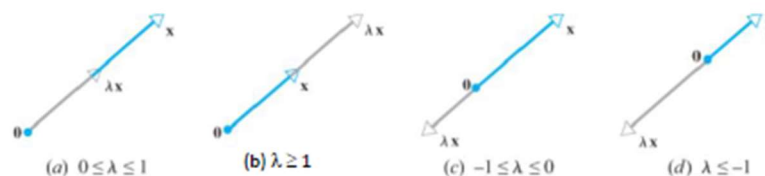
Skalar  $\lambda$  disebut nilai eigen dari  $A$ , dan  $\mathbf{x}$  dinamakan vektor eigen yang berkoresponden dengan  $\lambda$ . Nilai eigen menyatakan nilai karakteristik dari sebuah matriks yang berukuran  $n \times n$ .

Vektor eigen  $\mathbf{x}$  menyatakan matriks kolom yang apabila dikalikan dengan sebuah matriks  $n \times n$  menghasilkan vektor lain yang merupakan kelipatan vektor itu sendiri.



Gambar 5. Vektor Eigen

Dengan kata lain, operasi  $A\mathbf{x} = \lambda\mathbf{x}$  menyebabkan vektor  $\mathbf{x}$  menyusut atau memanjang dengan faktor  $\lambda$  dengan arah yang sama jika  $\lambda$  positif dan arah berkebalikan jika  $\lambda$  negatif.



Gambar 6. Arah Vektor Eigen berdasarkan Nilai Eigen

Tugas Besar 2 IF 2123 Aljabar Linear Dan Geometri  
Aplikasi Nilai Eigen dan EigenFace pada Pengenalan Wajah (Face Recognition)

Vektor eigen dan nilai eigen dari sebuah matriks  $A$  berukuran  $n \times n$  dihitung dengan cara sebagai berikut:

$$\begin{aligned} A\mathbf{x} &= \lambda\mathbf{x} \\ I A\mathbf{x} &= \lambda I\mathbf{x} \\ A\mathbf{x} &= \lambda I\mathbf{x} \\ (\lambda I - A)\mathbf{x} &= 0 \end{aligned}$$

$\mathbf{x} = 0$  adalah solusi trivial dari  $(\lambda I - A)\mathbf{x} = 0$ . Agar persamaan tersebut memiliki solusi tidak-nol, maka  $\det(\lambda I - A) = 0$ . Persamaan  $\det(\lambda I - A) = 0$  disebut persamaan karakteristik dari matriks  $A$  dan akar-akar persamaan tersebut, yaitu  $\lambda$ , dinamakan akar-akar karakteristik atau nilai-nilai eigen. Setelah nilai-nilai eigen didapatkan, nilai-nilai tersebut disubstitusikan kembali ke persamaan  $(\lambda I - A)\mathbf{x} = 0$ , sehingga didapatkan vektor-vektor eigen. Berikut adalah contoh pencarian nilai eigen dan vektor eigen:

Diberikan matriks  $A = \begin{bmatrix} 3 & 0 \\ 8 & -1 \end{bmatrix}$

$$\lambda I - A = \lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} 3 & 0 \\ 8 & -1 \end{bmatrix} = \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} - \begin{bmatrix} 3 & 0 \\ 8 & -1 \end{bmatrix} = \begin{bmatrix} \lambda - 3 & 0 \\ -8 & \lambda + 1 \end{bmatrix}$$

$$\det(\lambda I - A) = 0 \rightarrow \begin{vmatrix} \lambda - 3 & 0 \\ -8 & \lambda + 1 \end{vmatrix} = 0 \rightarrow (\lambda - 3)(\lambda + 1) = 0 \rightarrow \text{persamaan karakteristik}$$

$$\rightarrow \lambda_1 = 3 \text{ dan } \lambda_2 = -1$$

Jadi, nilai-nilai eigen dari matriks  $A$  adalah  $\lambda = 3$  dan  $\lambda = -1$ .

Gambar 7. Contoh Perhitungan Nilai Eigen

$$\text{Untuk } \lambda = 3 \rightarrow \begin{bmatrix} 0 & 0 \\ -8 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \rightarrow -8x_1 + 4x_2 = 0 \rightarrow 8x_1 = 4x_2 \rightarrow x_1 = \frac{1}{2}x_2$$

$$\rightarrow \text{Solusi: } x_1 = \frac{1}{2}t, x_2 = t, t \in \mathbf{R}$$

$$\text{Vektor eigen: } \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \frac{1}{2}t \\ t \end{bmatrix} = t \begin{bmatrix} \frac{1}{2} \\ 1 \end{bmatrix} \rightarrow \text{membentuk ruang eigen (eigenspace)}$$

$$\text{Jadi, } \begin{bmatrix} \frac{1}{2} \\ 1 \end{bmatrix} \text{ adalah basis untuk ruang eigen dengan } \lambda = 3$$

$$\text{Ruang eigen ditulis sebagai } E(3) = \{ \mathbf{x} = t \begin{bmatrix} \frac{1}{2} \\ 1 \end{bmatrix}, t \in \mathbf{R} \}$$

Gambar 8. Contoh Perhitungan Vektor Eigen

Selain itu, terdapat beberapa algoritma pencarian nilai dan vektor eigen, yaitu:

Tugas Besar 2 IF 2123 Aljabar Linear Dan Geometri  
Aplikasi Nilai Eigen dan EigenFace pada Pengenalan Wajah (Face Recognition)

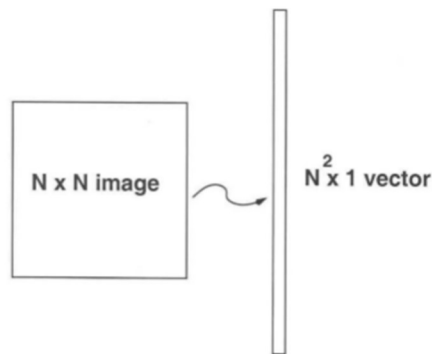
1. Lanczos algorithm
2. Power iteration
3. Inverse iteration
4. Rayleigh quotient iteration
5. Laguerre iteration
6. QR algorithm
7. Jacobi eigenvalue algorithm
8. Folded spectrum method

Dalam tugas besar ini, penulis menggunakan QR algorithm untuk perhitungan nilai eigen dan vektor eigen.

### C. Eigenface

Eigenface adalah sebutan untuk metode pengenalan wajah yang berdasarkan pada Principle Component Analysis (PCA). Eigenface dihasilkan dari sekumpulan citra digital dari wajah manusia yang kemudian dinormalisasikan. Adapun, langkah-langkah dalam algoritma Eigenface, yaitu:

1. Terdapat sejumlah  $m$  foto wajah manusia dengan dimensi  $N \times N$ . Foto ini dikonversi menjadi vektor  $\Gamma_i$  berukuran  $N \times 1$  sedemikian sehingga:



Gambar 9. Pemrosesan Foto Menjadi Vektor

$$S = \{\Gamma_1, \Gamma_2, \dots, \Gamma_m\}$$

2. Hitung rata-rata vektor wajah ( $\psi$ )

Tugas Besar 2 IF 2123 Aljabar Linear Dan Geometri  
Aplikasi Nilai Eigen dan EigenFace pada Pengenalan Wajah (Face Recognition)

$$\psi = \frac{1}{m} \sum_{i=1}^m x_i$$

3. Normalisasi vektor wajah menjadi  $\Phi$

$$\Phi_i = \Gamma_i - \psi$$

4. Gabungan setiap vektor wajah yang telah dinormalisasi sehingga didapatkan matriks A berukuran  $N^2 \times M$

$$A = \{\Phi_1, \Phi_2, \dots, \Phi_m\}$$

5. Cari matriks kovarian sehingga didapatkan sebuah matriks C berukuran  $M \times M$

$$C = AA^T$$

6. Hitung nilai eigen dan vektor eigen dari matriks kovarian tersebut dan gabungkan sebagai matriks C'

$$u_i = Av_i$$

7. Pilih K vektor eigen terbaik



### BAB III : Impelementasi Pustaka

Program dibuat dengan bahasa pemrograman *python*, dan memanfaatkan tkinter untuk GUI, library numpy untuk operasi matriks, dan library OpenCV untuk pemrosesan gambar. Secara garis besar, program terbagi menjadi: lib (memuat fungsi dan prosedur untuk pemrosesan), dan main.py (GUI program utama).

#### A. Lib

Lib memuat modul-modul yang digunakan untuk pemrosesan gambar. Modul-modul tersebut antara lain:

##### 1. processing.py

Fungsi/Prosedur	Keterangan	Penjelasan Algoritma
mean_image	Mendapatkan mean face $\psi$ dari vektor wajah	Menggunakan library numpy
normalize_image	Menormalisasi semua vektor wajah	$\Phi_i = \Gamma_i - \psi$
calculate_covariance	Menghitung matriks kovarian. Matriks kovarian memiliki ukuran MxM dengan M adalah banyaknya data.	$C = AA^T$
sort_image_by_eigenvalue	Mengurutkan eigenvalue dari yang terbesar. Hasil pengurutan juga dilakukan pada eigenvector, normalized_image, dan path.	
get_k_value	Menentukan nilai k, yakni banyaknya eigenface yang akan dipakai	Eigenvalue terurut akan dijumlahkan (eigenvalue negatif tidak dimutlakkan dan ikut dijumlahkan). Lalu, akan dicari nilai k, yang

Tugas Besar 2 IF 2123 Aljabar Linear Dan Geometri  
Aplikasi Nilai Eigen dan EigenFace pada Pengenalan Wajah (Face Recognition)

		memenuhi cumulative sum k eigenvalue pertama yang bernilai 95% dari total eigenvalue.
build_eigenfaces	Membuat eigenfaces dari eigenvector dan normalized_images	Membuat k eigenface dengan cara mengalikan eigenvector dan normalized image yang berkaitan.
calculate_weight	Menghitung weight setiap normalized image	Weight adalah hasil dari dot product antara eigenface dan normalized image. Setiap gambarnya, akan ada k weight karena sebelumnya kita memilih untuk menggunakan k buah eigenface.

## 2. Matrix.py

Fungsi/Prosedur	Keterangan	Penjelasan Algoritma
Frobenious_form	Menghitung normal matriks	Akar dari jumlah kuadrat matriks
Qr_factorization_householder	Algoritma dekomposisi QR eksplisit menggunakan metode householder	
Householder_vectorized	Menghitung vector householder	
Hessenberg_reduction	Mereduksi matriks kovarian menjadi matriks Hessenberg dengan metode householder	

Tugas Besar 2 IF 2123 Aljabar Linear Dan Geometri  
Aplikasi Nilai Eigen dan EigenFace pada Pengenalan Wajah (Face Recognition)

Qr_algorithm	Melakukan algoritma QR, untuk menghitung eigenvalue dan eigenvector	Matriks input dikonversi terlebih dahulu menjadi matriks hessenberg. Setiap loop, matriks tersebut dikalikan dengan dekomposisi Q transpose dari kiri dan Q dari kanan. Setelah puluhan loop, diagonal matriks akan konvergen menuju nilai eigenvalue.
--------------	---	--

### 3. Utils

Fungsi/Prosedur	Keterangan	Penjelasan Algoritma
Load_image	Memuat gambar dari path menjadi numpy array	Gambar dibaca dalam mode grayscale, lalu ukurannya diubah menjadi 256x256. Setelah itu, gambar diflatten sehingga menghasilkan array berukuran $1 \times 256^2$ .
Get_files	Membuat daftar gambar dengan ekstensi jpg atau png yang berada di dalam suatu folder	
Batch_load	Memuat semua gambar yang berada di dalam folder. Menghasilkan array of flatten image	

### 4. matching.py

Fungsi/Prosedur	Keterangan	Penjelasan Algoritma
-----------------	------------	----------------------

Tugas Besar 2 IF 2123 Aljabar Linear Dan Geometri  
Aplikasi Nilai Eigen dan EigenFace pada Pengenalan Wajah (Face Recognition)

match	Mencari closest image dari test image pada dataset lalu mengembalikan path serta kecocokan gambar tersebut	Menggunakan library numpy. Melakukan load test image menjadi matriks lalu menormalisasinya. Setelah itu menentukan eigenface dari test image. Terakhir, dilakukan perbandingan Euclidean distance dari test image dengan dataset serta menentukan kecocokan terbesar.
-------	--	---

5. camera.py

Fungsi/Prosedur	Keterangan	Penjelasan Algoritma
capture_image_from_camera	Menangkap gambar melalui kamera.	Menggunakan library numpy dan cv2. Gambar diambil memanfaatkan fungsi video capture lalu melakukan preprocessing gambar yang ditangkap (crop gambar dan membuat gambar di tengah)
Process_captured_image	Memproses gambar yang telah ditangkap	Gambar yang telah ditangkap diubah menjadi grayscale dan di-resize menjadi berukuran 256x256

## B. Main

File main berisi inisiasi dan implementasi GUI menggunakan TKinter. Selain itu, file ini juga mengimplementasikan modul lib dengan GUI sehingga program bisa terintegrasi dengan baik.

## BAB IV : Eksperimen

### A. Eigenvalue dan Eigenvector

Proses menghitung eigenvalue dan eigenvector matriks berukuran  $2 \times 2$  atau  $3 \times 3$  menggunakan tangan dengan menghitung eigenvalue dan eigenvector matriks berukuran  $100 \times 100$  lebih secara komputasional memiliki kompleksitas yang jauh berbeda. Sudah banyak orang yang meneliti bagaimana algoritma terbaik untuk menyelesaikan masalah ini, sehingga opsi yang tersedia cukup banyak. Kami memilih Algoritma QR eksplisit dengan dekomposisi QR menggunakan metode Householder untuk menghitung eigenvalue dan eigenvector. Sebenarnya masih ada algoritma lain yang lebih efisien, seperti Algoritma QR implisit dan Algoritma Divide and Conquer, tetapi setelah belasan jam mencoba mencari referensi dan mencoba memahami algoritma tersebut, kami harus puas dengan performa implementasi saat ini.

Berikut adalah performa implementasi algoritma QR dalam menghadapi ukuran matriks yang berbeda-beda. Jumlah loop yang dipakai dalam algoritma QR adalah 50. Terdapat pula performa implementasi LAPACK (library yang dipakai numpy untuk menghitung eigen) sebagai perbandingan.

Ukuran matriks	Waktu eksekusi (Implementasi kami)	Waktu Eksekusi (Numpy)
100x100	0,8 detik	0,1 detik
200x200	4,8 detik	0,1 detik
500x500	55,7 detik	0,2 detik
1000x1000	9 menit 48 detik	0,8 detik

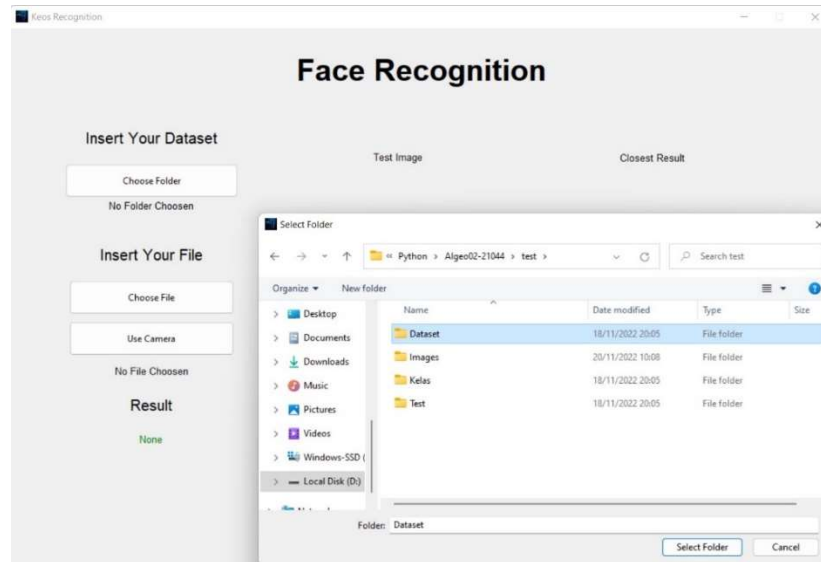
Terlihat bahwa implementasi yang kami buat masih memiliki efisiensi yang sangat kurang bila dibandingkan dengan implementasi yang digunakan oleh LAPACK. Setidaknya, implementasi ini masih cukup untuk use case dengan dataset berukuran 200 hingga 300 gambar.

### B. Data Test

# Tugas Besar 2 IF 2123 Aljabar Linear Dan Geometri

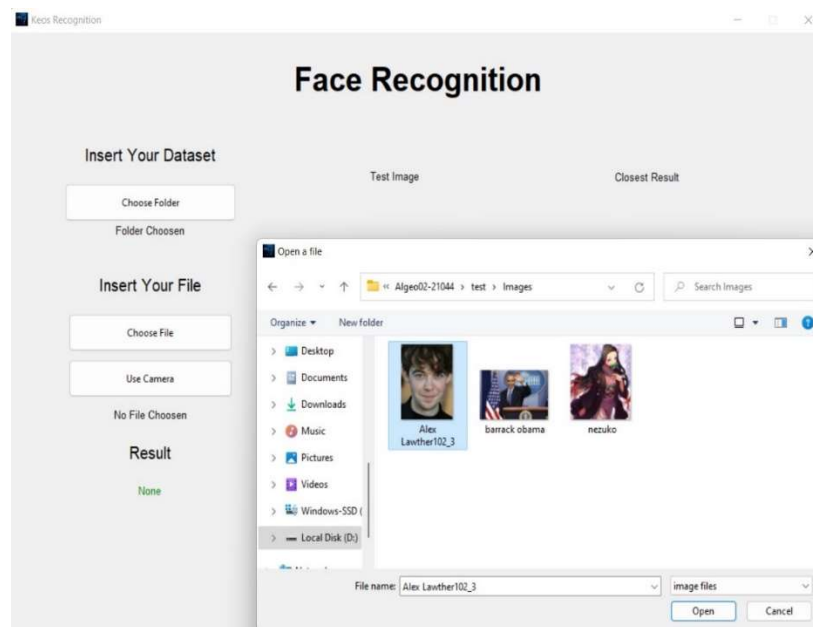
## Aplikasi Nilai Eigen dan EigenFace pada Pengenalan Wajah (Face Recognition)

### 1. Memilih Folder Dataset



Gambar 10. Pemilihan Dataset

### 2. Memilih test image dari file yang sudah ada



Gambar 11. Pemilihan Test Image dari File pada Local

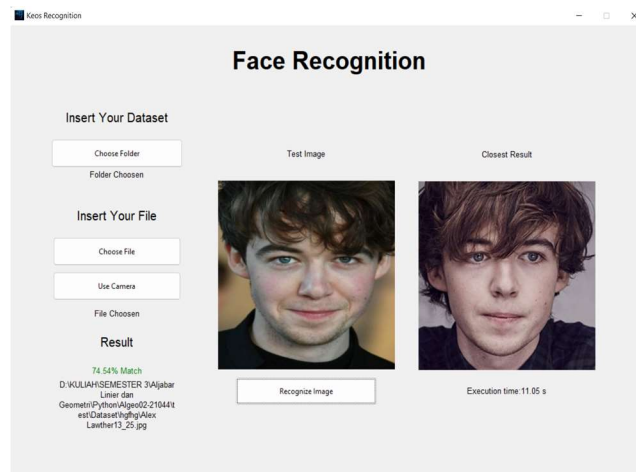
## Tugas Besar 2 IF 2123 Aljabar Linear Dan Geometri Aplikasi Nilai Eigen dan EigenFace pada Pengenalan Wajah (Face Recognition)

### 3. Recognize image berhasil (terdapat image yang mirip pada dataset)

Untuk semua data test, digunakan sebuah dataset yang terdiri atas 117 foto wajah 1:1 dengan ukuran berbeda-beda.

**Data Test 1 : Alex Lawther**

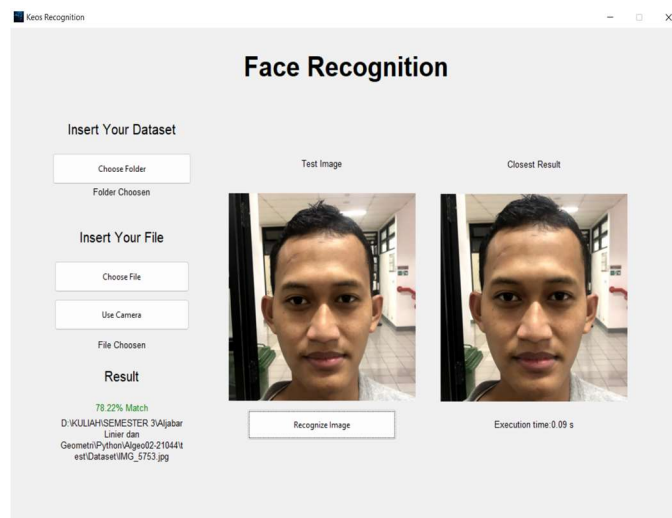
**Hasil : Berhasil**



Gambar 12. Hasil Face Recognition dengan Data Test Foto Alex Lawther

**Data Test 2 : Mutuwally Nawwar**

**Hasil : Berhasil**



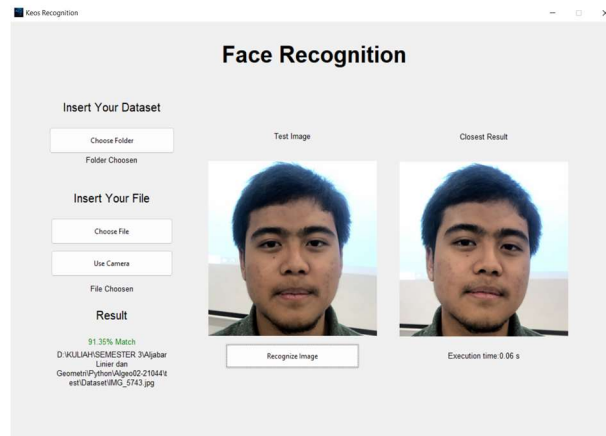
Gambar 13. Hasil Test dengan Foto Mutuwally Nawwar

## Tugas Besar 2 IF 2123 Aljabar Linear Dan Geometri

### Aplikasi Nilai Eigen dan EigenFace pada Pengenalan Wajah (Face Recognition)

**Data Test 3 : Fatih Nararya**

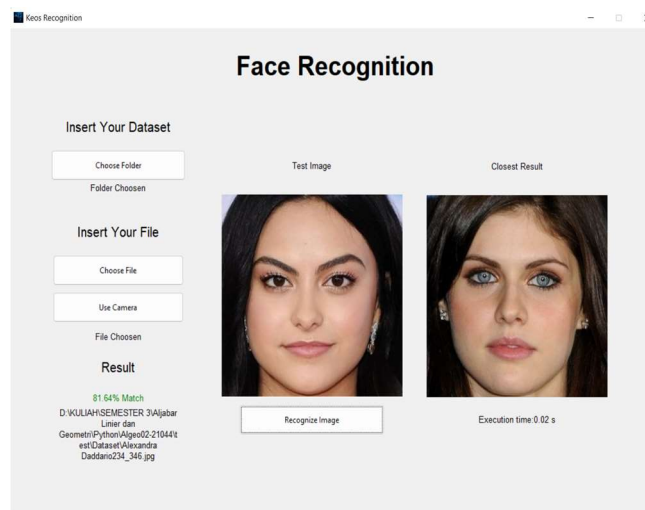
**Hasil : Berhasil**



*Gambar 14. Hasil Test dengan Foto Fatih Nararya*

**Data Test 4 : Camilla Mendes**

**Hasil : Gagal (Wajah yang dikenali bukan wajah test image)**



*Gambar 15. Hasil Test dengan Foto Camilla Mendes (Gagal)*



Tugas Besar 2 IF 2123 Aljabar Linear Dan Geometri  
Aplikasi Nilai Eigen dan EigenFace pada Pengenalan Wajah (Face Recognition)

**Data Test 5 : Sehun**

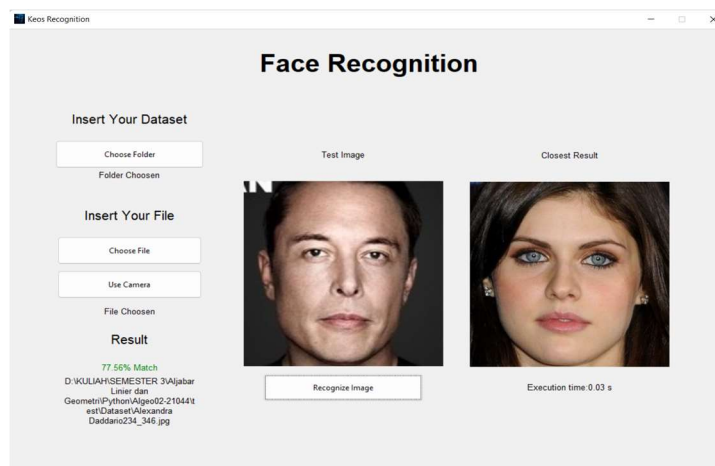
**Hasil : Gagal (Wajah yang dikenali bukan wajah test image)**



*Gambar 16. Hasil Test dengan Foto Sehun (Gagal)*

**Data Test 6 : Elon Musk**

**Hasil : Gagal (Wajah yang dikenali bukan wajah test image)**

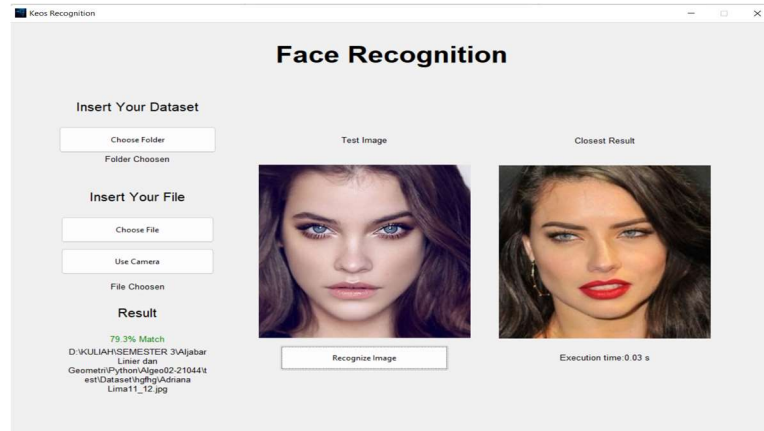


*Gambar 17. Hasil Test dengan Foto Elon Musk (Gagal)*

Tugas Besar 2 IF 2123 Aljabar Linear Dan Geometri  
Aplikasi Nilai Eigen dan EigenFace pada Pengenalan Wajah (Face Recognition)

**Data Test 7 : Barbara Palvin**

**Hasil : Gagal (Wajah yang dikenali bukan wajah test image)**



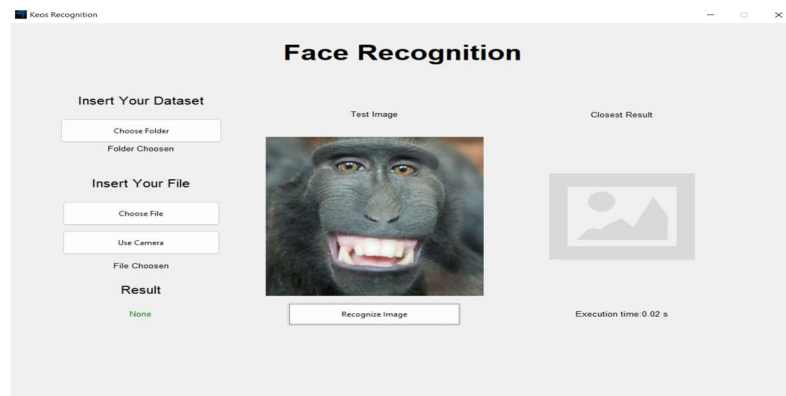
*Gambar 18. Hasil Test dengan Foto Barbara Palvin (Gagal)*

4. Recognize image gagal (tidak terdapat image yang mirip pada dataset)

Untuk semua data test, digunakan sebuah dataset yang terdiri atas 117 foto wajah 1:1 dengan ukuran berbeda-beda.

**Data Test 8 : Monyet**

**Hasil : Berhasil (Tidak dikenali karena tidak ada wajah pada data test yang mirip dengan test image)**

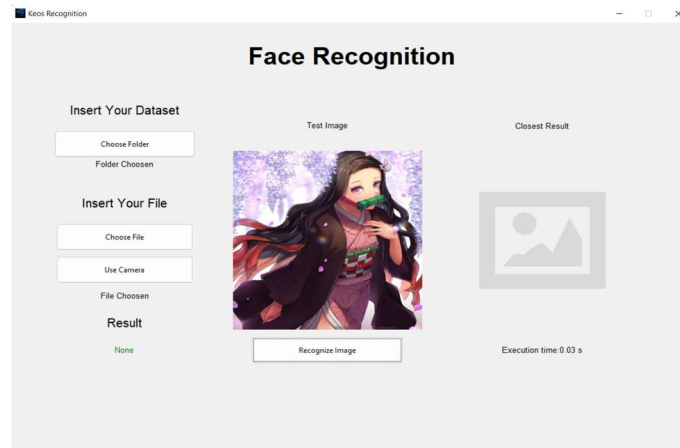


*Gambar 19. Hasil Test dengan Foto Monyet*

Tugas Besar 2 IF 2123 Aljabar Linear Dan Geometri  
Aplikasi Nilai Eigen dan EigenFace pada Pengenalan Wajah (Face Recognition)

**Data Test 9 : Foto Waifu**

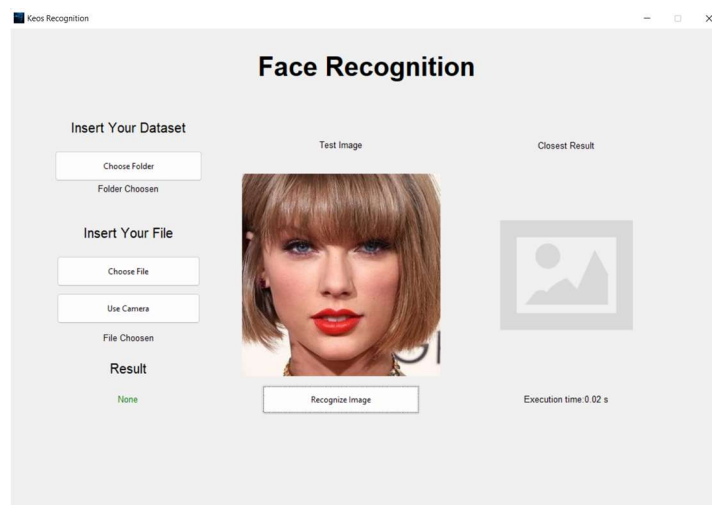
**Hasil : Berhasil (Tidak dikenali karena tidak ada wajah pada data test yang mirip dengan test image)**



*Gambar 20. Hasil Test dengan Foto Tanpa Wajah*

**Data Test 10 : Taylor Swidft**

**Hasil : Berhasil (Tidak dikenali karena tidak ada wajah pada data test yang mirip dengan test image)**



*Gambar 21. Hasil Test dengan Foto Taylor Swift*

Tugas Besar 2 IF 2123 Aljabar Linear Dan Geometri  
Aplikasi Nilai Eigen dan EigenFace pada Pengenalan Wajah (Face Recognition)

**Data Test 11 : Barack Obama**

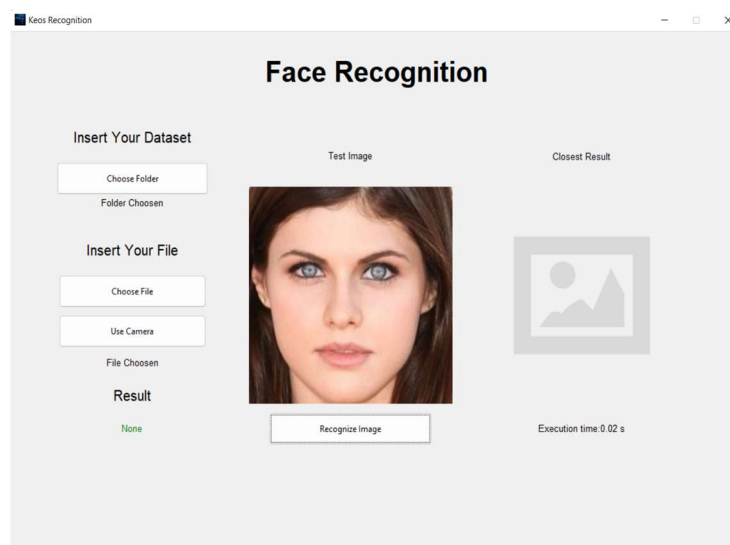
**Hasil : Gagal (Test image ada pada dataset, tetapi tidak kenali)**



*Gambar 22. Hasil Test dengan Foto Barack Obama*

**Data Test 12 : Alexandra Daddario**

**Hasil : Gagal (Test image ada pada dataset, tetapi tidak kenali)**

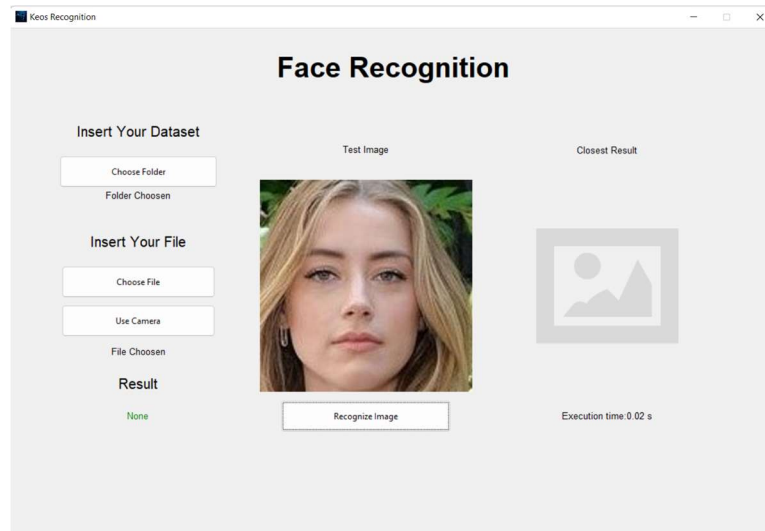


*Gambar 23. Hasil Test dengan Foto Alexandra Daddario*

Tugas Besar 2 IF 2123 Aljabar Linear Dan Geometri  
Aplikasi Nilai Eigen dan EigenFace pada Pengenalan Wajah (Face Recognition)

**Data Test 13 : Amber Heard**

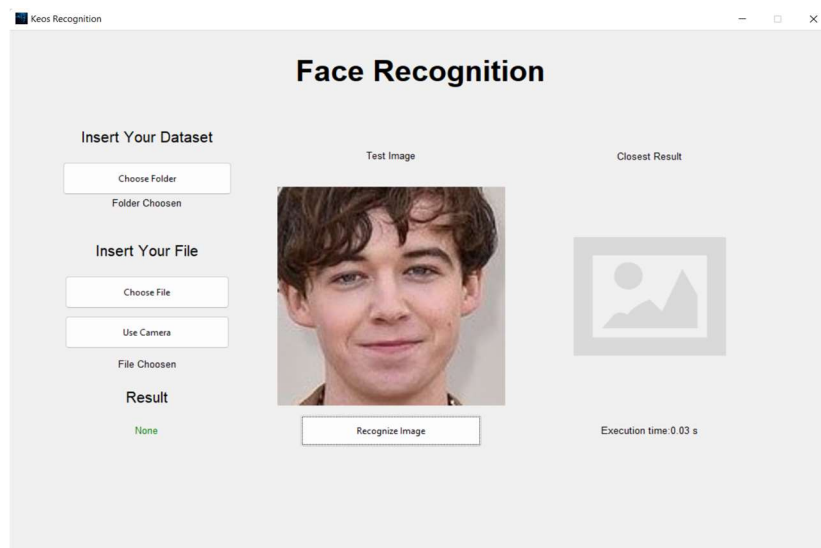
**Hasil : Gagal (Test image ada pada dataset, tetapi tidak kenali)**



*Gambar 24. Hasil Test dengan Foto Amber Heard*

**Data Test 14 : Alex Lawther**

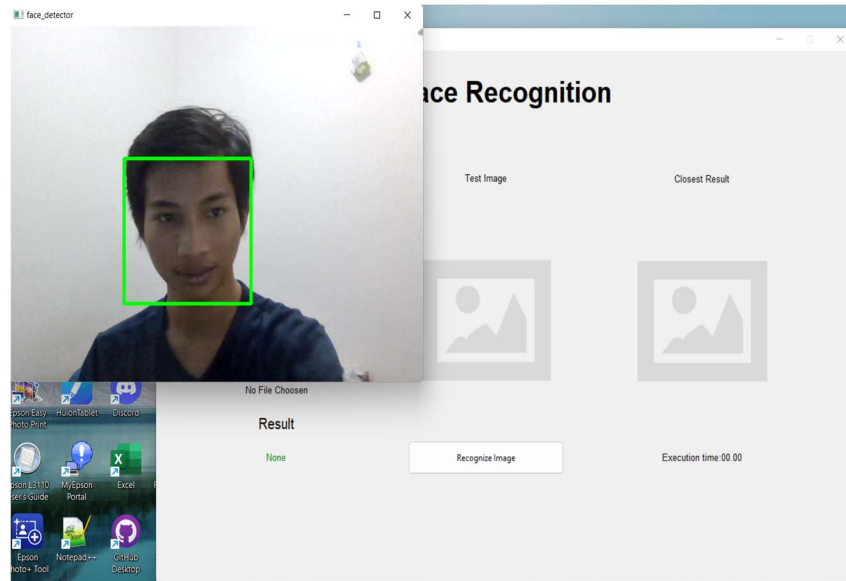
**Hasil : Gagal (Test image ada pada dataset, tetapi tidak kenali)**



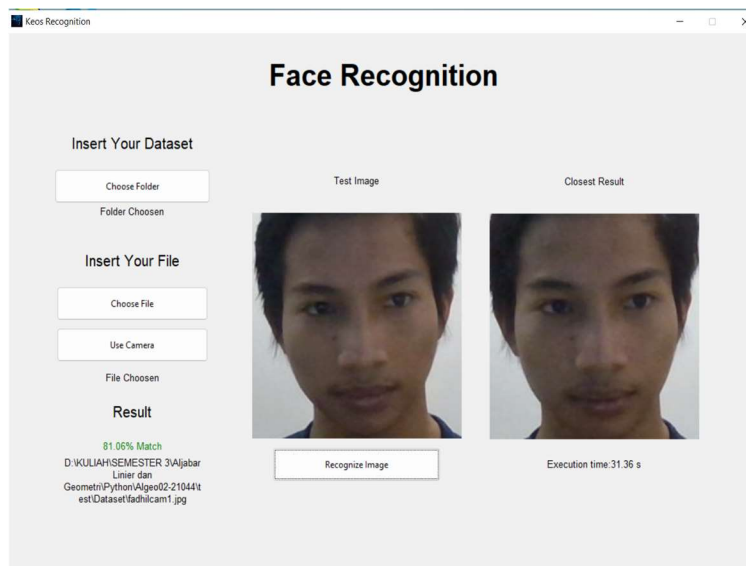
*Gambar 25. Hasil Test dengan Foto Alex Lawther*

Tugas Besar 2 IF 2123 Aljabar Linear Dan Geometri  
Aplikasi Nilai Eigen dan EigenFace pada Pengenalan Wajah (Face Recognition)

5. Test Image Menggunakan Kamera



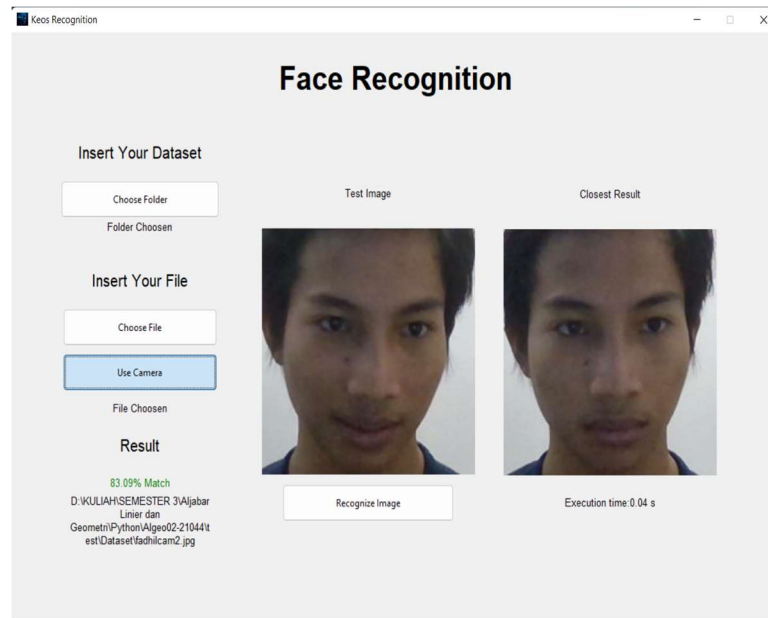
Gambar 16. Proses Pengambilan Foto dengan Kamera



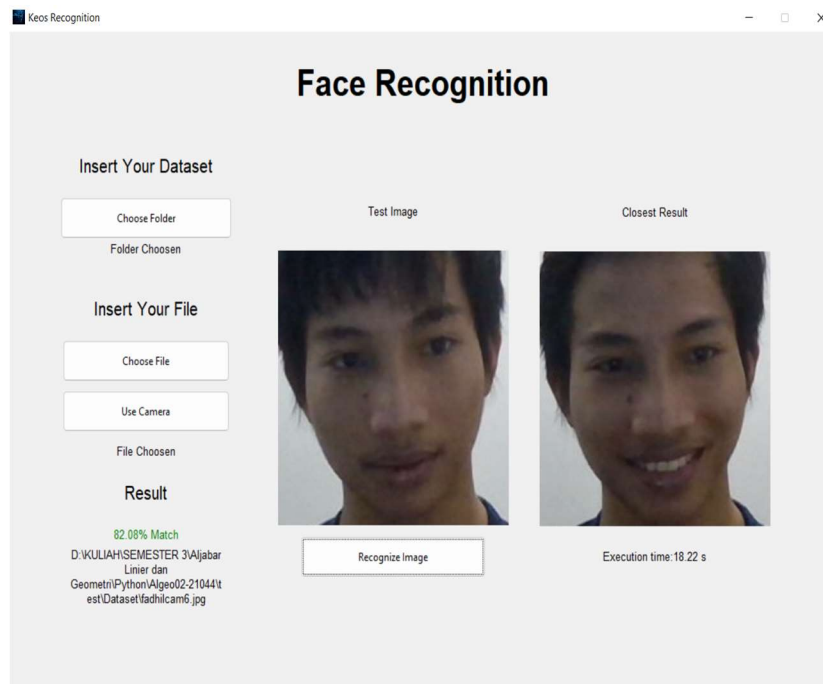
Gambar 17. Hasil Test dengan Foto dari Kamera 1

## Tugas Besar 2 IF 2123 Aljabar Linear Dan Geometri

### Aplikasi Nilai Eigen dan EigenFace pada Pengenalan Wajah (Face Recognition)



Gambar 18. Hasil Test dengan Foto dari Kamera 2



Gambar 19. Hasil Test dengan Kamera 3

## **BAB V : Kesimpulan, Saran dan Refleksi**

### **A. Kesimpulan**

Selama proses implementasi perhitungan eigen value dan eigenvector secara komputasional, kami menyadari bahwa metode perhitungan kedua nilai tersebut sangat sulit, terlebih lagi bila ingin mengejar efisiensi algoritma agar perhitungan bisa berjalan dengan cepat. Selain itu, metode pengenalan wajah menggunakan eigenface ini memiliki banyak keterbatasan karena sangat bergantung pada kualitas dataset dan test image yang digunakan. Hasil eksperimen pada Bab 4 menunjukkan bahwa algoritma ini bekerja dengan baik pada test image yang diambil pada pose wajah, pencahayaan dan kualitas kamera yang sama dengan foto pada data test. Sedangkan, algoritma ini kerap gagal dengan pose wajah dan pencahayaan yang berbeda. Artinya, Keakuratan proses matching dipengaruhi oleh pencahayaan dan orientasi wajah pada test yang digunakan. Hemat kami, metode pengenalan wajah dengan eigenface memiliki akurasi dan *reliability* yang kurang sehingga tidak cocok digunakan sebagai standar industri untuk pengenalan wajah.

### **B. Saran**

Sebagai manusia, kami menyadari bahwa hasil pengerjaan kami tidak sempurna, baik terkait optimasi performa program, estetika tampilan program, kerapihan laporan, atau pun hal lainnya. Oleh karena itu, kami sangat terbuka untuk menerima segala masukan, perbaikan, dan pengembangan lebih lanjut terhadap karya kami.

### **C. Refleksi**

Kami bersyukur dapat menyelesaikan tugas besar ini dengan tepat waktu dan juga bisa mengerjakan bagian bonus. Selain itu, tugas besar ini telah menjadi sarana bagi kami untuk mengeksplorasi dunia pemrograman dan memperdalam konsep yang telah diajarkan. Terlebih lagi, proses implementasi perhitungan nilai eigen dan vektor eigen merupakan hal yang sangat menantang dan menguras pikiran.

Masih banyak hal yang ingin kami gali lebih lanjut terkait tugas besar ini, mulai dari optimasi performa algoritma untuk menghitung nilai dan vektor eigen, hingga bereksperimen mengenai



Tugas Besar 2 IF 2123 Aljabar Linear Dan Geometri  
Aplikasi Nilai Eigen dan EigenFace pada Pengenalan Wajah (Face Recognition)

sejauh mana metode pengenalan wajah menggunakan eigenface ini bisa diimplementasikan, serta apa saja batasan yang dimiliki metode ini. Kami berharap kami bisa menggali permasalahan ini lebih lanjut di masa yang akan datang.

### DAFTAR PUSTAKA

Anton, H., Rorres, C., & Kaul, A. 2010. Elementary Linear Algebra: Applications Version (11th ed.). Wiley.

Arbenz, Peter. “Numerical Methods for Solving Large Scale Eigenvalue Problems, chapter 4”. Diakses 8 November 2022, <https://people.inf.ethz.ch/arbenz/ewp/Lnotes/chapter4.pdf>.

Denis, Ben. “Eigenvalues and Eigenvector”. Diakses 8 November 2022, dari <https://towardsdatascience.com/eigenvalues-and-eigenvectors-89483fb56d56>.

Flom, Peter. “Euclidean distance score and similarity”. Diakses 16 November 2022, dari <https://stats.stackexchange.com/questions/53068/euclidean-distance-score-and-similarity>

Henrik. “B2AC – Ellipse fitting in Python and C”. Diakses 8 November 2022, dari [https://github.com/hbldh/b2ac/blob/master/b2ac/matrix/matrix\\_algorithms.py](https://github.com/hbldh/b2ac/blob/master/b2ac/matrix/matrix_algorithms.py)

Lambers, Jim. “Section 6.3: The QR Algorithm”. Diakses 8 November 2022, dari <https://www.math.usm.edu/lambers/mat610/class0329.pdf>

Munir, Rinaldi. 2022. “Sistem persamaan linear (Bagian 2: Metode Eliminasi Gauss Jordan)”. Diakses 30 September 2022, dari <https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2022-2023/algeo22-23.htm>

Munir, Rinaldi. 2022. “Determinan (Bagian 1: menghitung determinan dengan reduksi baris)”. Diakses 30 September 2022, dari <https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2022-2023/algeo22-23.htm>

Tugas Besar 2 IF 2123 Aljabar Linear Dan Geometri  
Aplikasi Nilai Eigen dan EigenFace pada Pengenalan Wajah (Face Recognition)

Munir, Rinaldi. 2022. "Determinan (Bagian 2: menghitung determinan dengan ekspansi kofaktor)". Diakses 30 September 2022, dari

<https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2022-2023/algeo22-23.htm>

Mikulski, Bartosz. "PCA – How to choose the number of components?". Diakses 11 November 2022, <https://www.mikulskibartosz.name/pca-how-to-choose-the-number-of-components/>

Miguel, Juan. "Face Recognition Using Eigenfaces". Diakses 14 November 2022, <http://laid.delanover.com/explanation-face-recognition-using-eigenfaces/>.

Numpy Documentation, "numpy.linalg.norm". Diakses 8 November 2022, dari <https://numpy.org/doc/stable/reference/generated/numpy.linalg.norm.html>

Tel on Stackoverflow. "How can you implement Householder based QR decomposition in Python?". Diakses 8 November 2022, dari <https://stackoverflow.com/questions/53489237/how-can-you-implement-householder-based-qr-decomposition-in-python>

Weisstein, Eric W. "Gaussian Elimination." From MathWorld--A Wolfram Web Resource. Diakses 30 September 2022, dari <https://mathworld.wolfram.com/GaussianElimination.html>

Weisstein, Eric W. "Identity Matrix." From MathWorld--A Wolfram Web Resource. Diakses 30 September 2022, dari <https://mathworld.wolfram.com/IdentityMatrix.html>

Python Tutorial. 2022. "Tkinter". Diakses 9 November 2022, dari <https://www.pythontutorial.net/tkinter/>

Real Python. 2022. "Python GUI Programming With Tkinter". Diakses 9 November 2022, dari <https://realpython.com/python-gui-tkinter/>

Geeks for Geeks. 2021. "Python Tkinter Tutorial". Diakses 9 November 2022, dari <https://www.geeksforgeeks.org/python-tkinter-tutorial/>

Tugas Besar 2 IF 2123 Aljabar Linear Dan Geometri  
Aplikasi Nilai Eigen dan EigenFace pada Pengenalan Wajah (Face Recognition)

#### REPOSITORY

Link repository dari Tugas Besar 2 IF 2123 Aljabar Linear Dan Geometri kelompok Keos Recognition adalah sebagai berikut.

<https://github.com/Mehmed13/Algeo02-21044.git>

#### VIDEO PRESENTASI

Pranala video presentasi Tugas Besar 2 IF-2123 Aljabar Linear dan Geometri kelompok kami adalah sebagai berikut.

<https://youtu.be/yE6j27FbCWw>