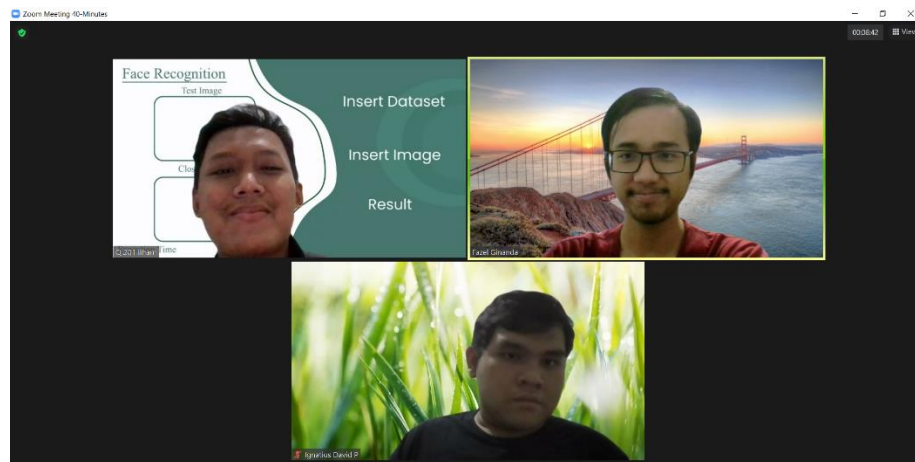


LAPORAN TUGAS BESAR 2  
IF2123 Aljabar Linier dan Geometri  
Aplikasi Nilai Eigen dan EigenFace pada Pengenalan Wajah (*Face Recognition*)



Disusun oleh:  
Ignatius David Partogi (13518014)  
Ilham Akbar (13521068)  
Fazel Ginanda (13521098)

Semester I Tahun 2022/2023

## 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.

Pada tugas besar ini, penyusun ditugaskan untuk membuat program pengenalan wajah dalam Bahasa Python berbasis GUI dengan spesifikasi sebagai berikut:

1. Program menerima input folder dataset dan sebuah gambar citra wajah.
2. Basis data wajah dapat diunduh secara mandiri melalui <https://www.kaggle.com/datasets/hereisburak/pins-face-recognition>.
3. Program menampilkan gambar citra wajah yang dipilih oleh pengguna.
4. Program melakukan pencocokan wajah dengan koleksi wajah yang ada di folder yang telah dipilih. Metrik untuk pengukuran kemiripan menggunakan eigenface + jarak euclidean.
5. Program menampilkan 1 hasil pencocokan pada dataset yang paling dekat dengan gambar input atau memberikan pesan jika tidak didapatkan hasil yang sesuai.
6. Program menghitung jarak euclidean dan nilai eigen & vektor eigen yang ditulis sendiri. Tidak boleh menggunakan fungsi yang sudah tersedia di dalam library atau Bahasa Python.

## BAB II

### TEORI SINGKAT

### A. Perkalian Matriks

Syarat perkalian dua buah matriks adalah jumlah kolom pada matriks pertama sama dengan jumlah baris pada matriks kedua. Misalkan A adalah sebuah matriks berukuran  $m \times r$  dan B adalah sebuah matriks berukuran  $r \times n$ . Misalkan hasil kali matriks A dengan matriks B adalah matriks C, maka ukuran matriks C adalah  $m \times n$ . Entri dari matriks C adalah sebagai berikut.

$$C = A \times B = [c_{ij}], c_{ij} = a_{i1}b_{1j} + a_{i2}b_{2j} + \dots + a_{in}b_{nj}$$

Contoh perkalian matriks adalah sebagai berikut.

$$A = \begin{bmatrix} 1 & 2 & 4 \\ 2 & 6 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 4 & 1 & 4 & 3 \\ 0 & -1 & 3 & 1 \\ 2 & 7 & 5 & 2 \end{bmatrix}$$

Hasil kali dari matriks A dan B adalah matriks AB sebagai berikut

$$\begin{bmatrix} 12 & 27 & 30 & 13 \\ 8 & -4 & 26 & 12 \end{bmatrix}$$

### B. Nilai Eigen dan Vektor Eigen

Nilai eigen adalah nilai karakteristik dari suatu matriks bujursangkar berukuran  $n \times n$ . Jika A adalah matriks berukuran  $n \times n$ , maka vektor tidak nol  $x$  di  $R^n$  disebut vektor eigen dari A jika  $Ax$  sama dengan perkalian suatu skalar  $\lambda$  dengan  $x$ .

$$Ax = \lambda x$$

Skalar  $\lambda$  disebut nilai eigen dari A dan  $x$  merupakan vektor eigen yang berkoresponden dengan  $\lambda$ . Vektor eigen  $x$  menyatakan matriks kolom yang apabila dikalikan dengan sebuah matriks  $n \times n$  menghasilkan vektor lain yang merupakan kelipatan dari vektor tersebut. Dapat disimpulkan, operasi  $Ax = \lambda x$  menghasilkan vektor  $x$  yang menyusut atau memanjang dengan faktor  $\lambda$ .

### C. Eigenface

Eigenface adalah metode pengenalan wajah berbasis nilai eigen dan vektor eigen yang digunakan untuk persoalan-persoalan dalam *computer vision*. Metode ini dikembangkan oleh Sirovich dan Kirby. Metode ini juga digunakan oleh Matthew Turk dan Alex Pentland dalam klasifikasi wajah. Vektor eigen dalam metode eigenface diturunkan dari matriks kovarian dari sejumlah citra wajah

latih (training image). Bentuk wajah direpresentasikan secara matematis sebagai kombinasi linier dari *eigenheads* yaitu set himpunan berisi variasi wajah.

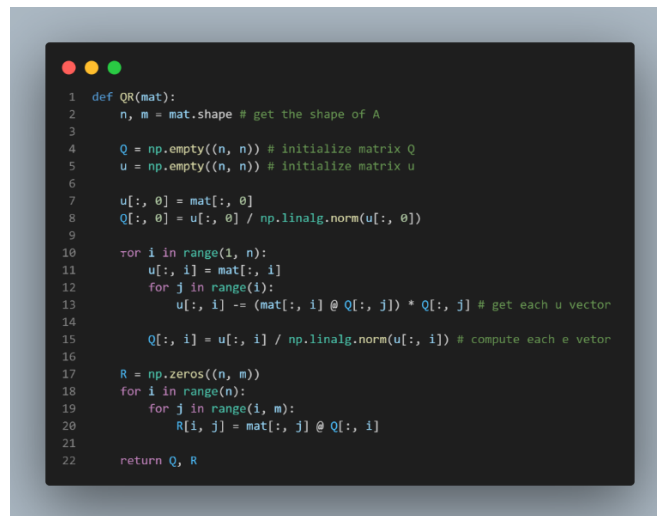
### BAB III

#### IMPLEMENTASI PROGRAM

Pada pengerjaan tugas besar ini, terdapat lima modul program di dalam folder src dan satu file program di dalam folder GUI. Program-program yang terdapat di dalam folder src adalah program untuk memproses citra wajah dan menjalankan algoritma eigenface. Sementara itu, program yang ada di folder GUI bertujuan untuk memproses interaksi pengguna dengan program secara grafis.

Program yang terdapat di dalam folder src adalah main.py, eigen.py, pict.py, matrix.py, dan mat\_operator.py. Pada folder GUI, terdapat file GUI.py. Dalam implementasi program ini, digunakan library pemrosesan gambar, yaitu OpenCV dan PIL. Sementara itu, untuk melakukan operasi matematika, digunakan library math dan numpy.

Pada program eigen.py, terdapat algoritma untuk mencari nilai eigen dari suatu matriks bujursangkar. Agar nilai eigen bisa dihitung, matriks didekomposisi terlebih dahulu dengan metode QR.



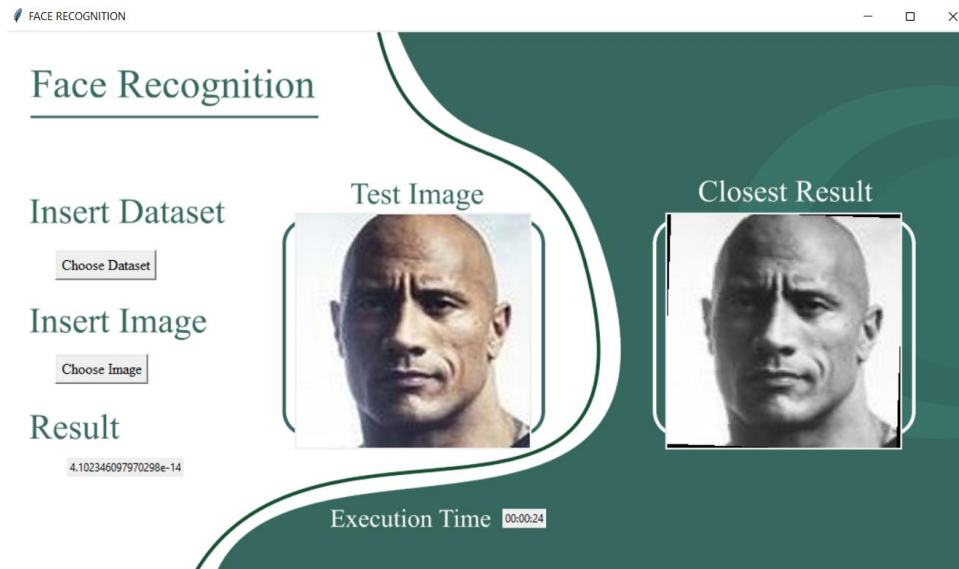
```
1 def QR(mat):
2     n, m = mat.shape # get the shape of A
3
4     Q = np.empty((n, n)) # initialize matrix Q
5     u = np.empty((n, n)) # initialize matrix u
6
7     u[:, 0] = mat[:, 0]
8     Q[:, 0] = u[:, 0] / np.linalg.norm(u[:, 0])
9
10    for i in range(1, n):
11        u[:, i] = mat[:, i]
12        for j in range(i):
13            u[:, i] -= (mat[:, i] @ Q[:, j]) * Q[:, j] # get each u vector
14
15        Q[:, i] = u[:, i] / np.linalg.norm(u[:, i]) # compute each e vetor
16
17    R = np.zeros((n, m))
18    for i in range(n):
19        for j in range(i, m):
20            R[i, j] = mat[:, j] @ Q[:, i]
21
22    return Q, R
```

Untuk mencari nilai eigen dan vektor eigen, penyusun membuat fungsi eigen\_Val dan eigen\_Vec yang juga terdapat pada file eigen.py.

```
1 def eigen_Val(mat):
2     pQ = np.eye(mat.shape[0])
3     X = np.copy(mat)
4     for i in range(1000):
5         Q, R = QR(X)
6         pQ = pQ @ Q
7         X = R @ Q
8     return np.diag(X)
9
10 def eigen_Vec(mat):
11     pQ = np.eye(mat.shape[0])
12     X = np.copy(mat)
13     for i in range(1000):
14         Q, R = QR(X)
15         pQ = pQ @ Q
16         X = R @ Q
17     return pQ
```

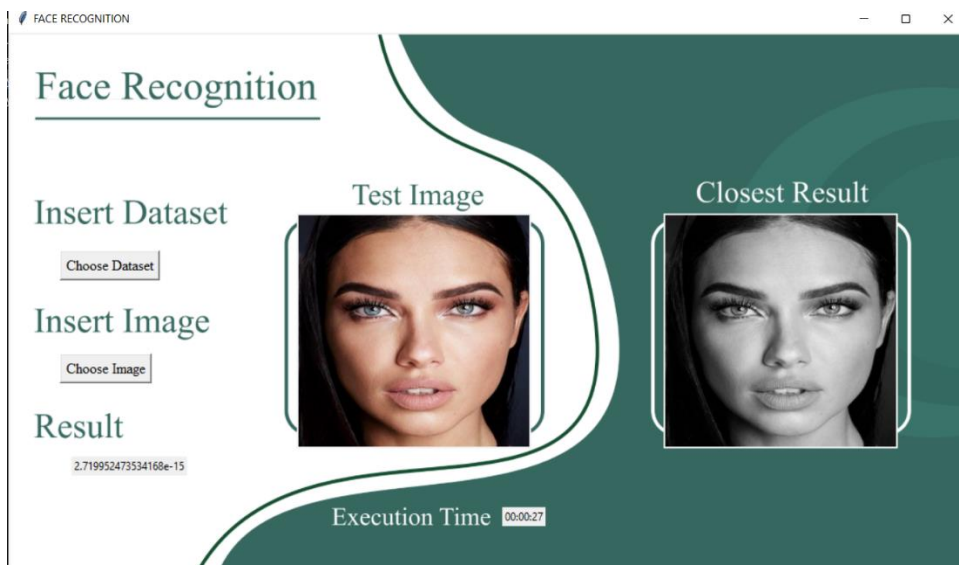
## BAB IV EKSPERIMEN

### A. Foto input testface terdapat pada database



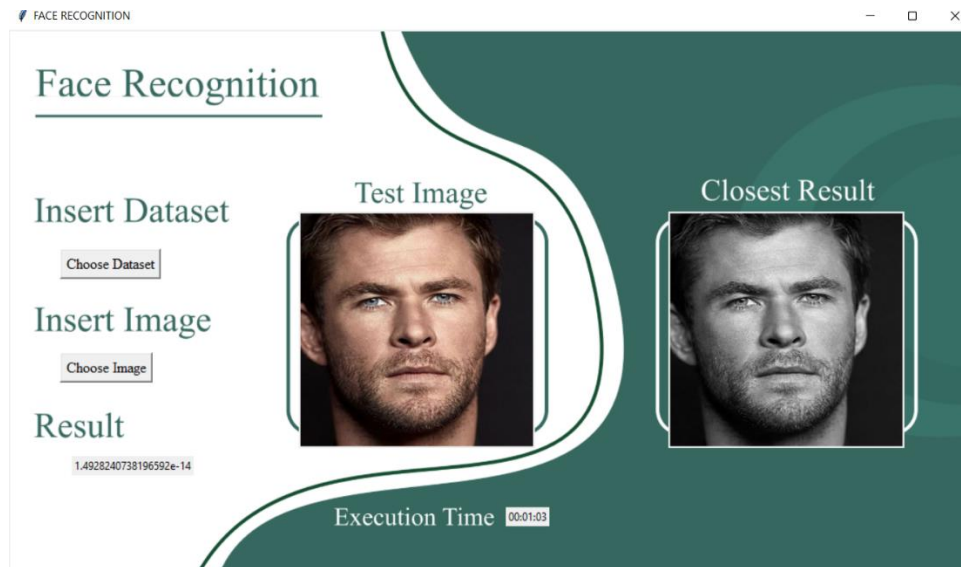
Gambar 4.1.1 Dwayne Johnson

Hasil eksekusi tersebut memperlihatkan testface yaitu image Dwayne Johnson dan gambar yang paling mirip dengan testface yang ditemukan pada database. Program memberikan output yang benar, yaitu gambar yang mirip disertai dengan waktu eksekusi program selama 24 detik.



Gambar 4.1.2 Adriana Lima

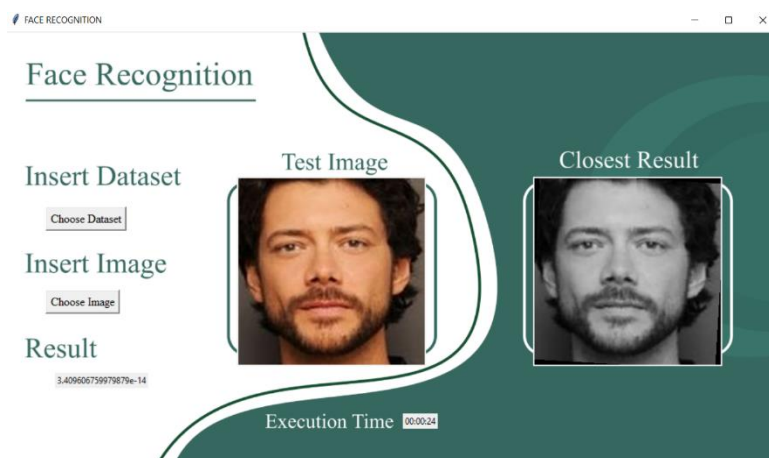
Hasil eksekusi tersebut memperlihatkan testface yaitu image Adriana Lima dan gambar yang paling mirip dengan testface yang ditemukan pada database. Program memberikan output yang benar, yaitu gambar yang mirip disertai dengan waktu eksekusi program selama 27 detik.



Gambar 4.1.3 Chris Hemsworth

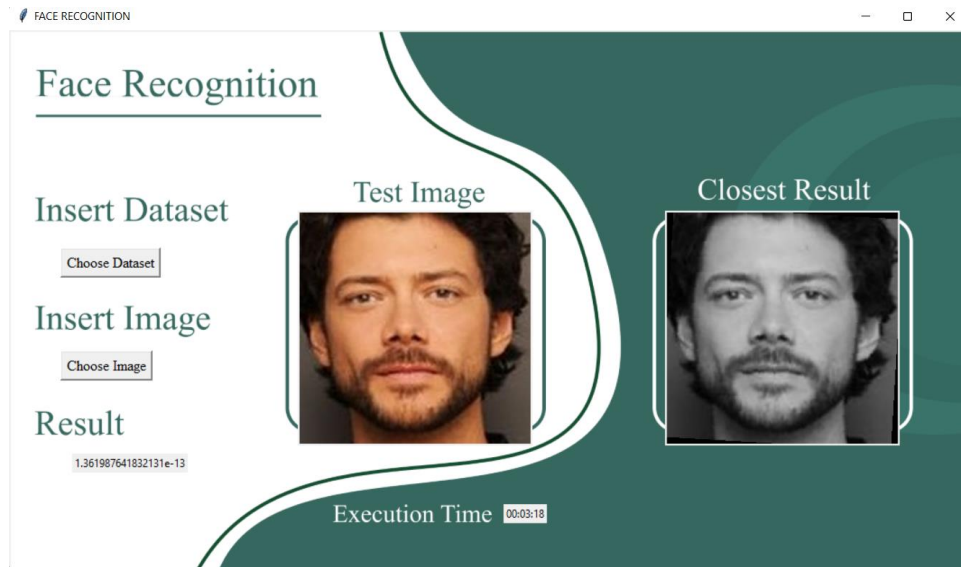
Hasil eksekusi tersebut memperlihatkan testface yaitu image Chris Hemsworth dan gambar yang paling mirip dengan testface yang ditemukan pada database. Program memberikan output yang benar, yaitu gambar yang mirip disertai dengan waktu eksekusi program selama 1 menit 3 detik.

## B. Perubahan Jumlah Gambar pada Database

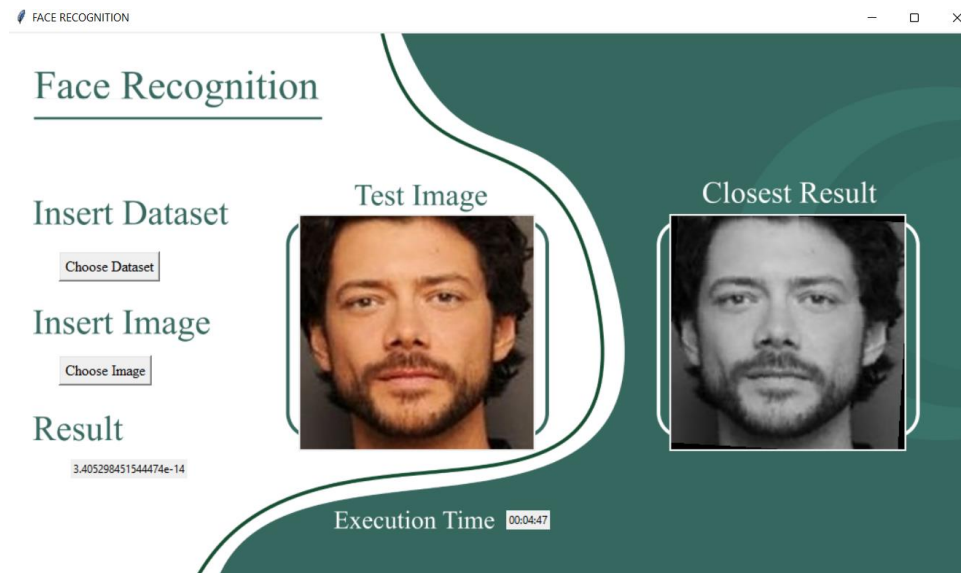


Gambar 4.2.1 Alvaro





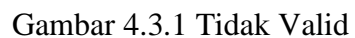
Gambar 4.2.2 Alvaro

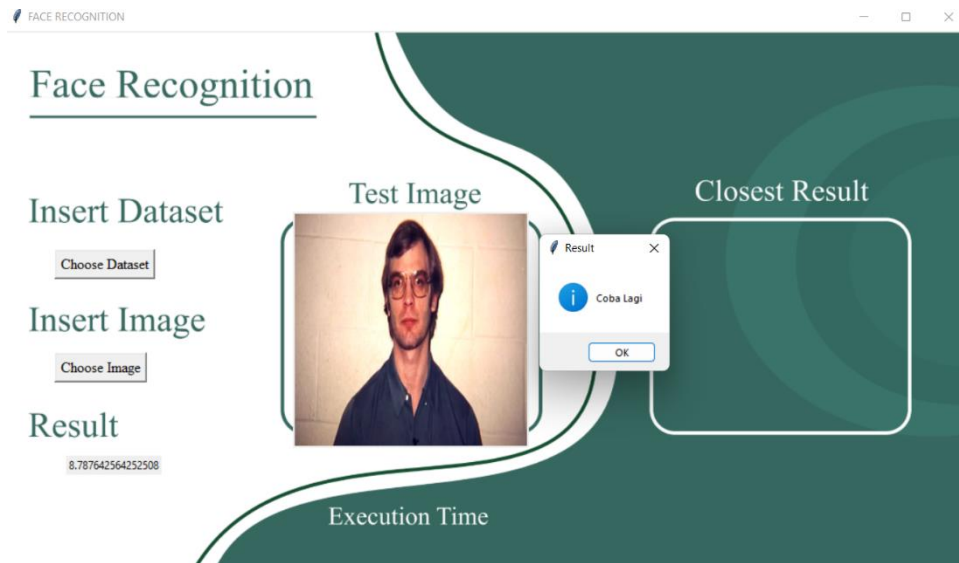


Gambar 4.2.3 Alvaro

Percobaan pertama dilakukan dengan memasukkan foto Alvaro. Ketiga input tersebut berhasil memberikan keluaran dari set data, berupa foto yang mengandung wajah yang tepat. Pada percobaan ini, dilakukan pengujian pengaruh perubahan jumlah gambar pada database terhadap waktu eksekusi program. Pada percobaan pertama, dengan jumlah gambar sebanyak 44 gambar pada database, program dieksekusi dalam waktu 24 detik. Pada percobaan kedua dengan jumlah gambar sebanyak 20 gambar pada database, program dieksekusi dalam waktu 3 menit 18 detik. Pada percobaan ketiga, dengan jumlah gambar sebanyak 10 gambar pada database, program

### C. Input Tidak Valid





Gambar 4.3.2 Jeffrey Dahmer

Percobaan selanjutnya adalah memasukkan foto Jeffrey Dahmer, serta sembarang gambar. Input tersebut ternyata tidak bisa memberikan keluaran, karena tidak terdapat foto pada database yang lumayan mendekati input.

## BAB V

### SIMPULAN, SARAN, DAN REFLEKSI

#### A. Simpulan

Pada tugas besar ini, penyusun telah membuat program pengenalan wajah dalam Bahasa Python berbasis GUI. Metode yang digunakan dalam pengenalan wajah adalah eigenface. Metode tersebut merupakan aplikasi dari materi nilai eigen dan vektor eigen yang sudah dipelajari di kelas. Pengguna dapat menjalankan program ini untuk menentukan apakah ada gambar wajah pada folder basis data yang cocok dengan gambar wajah input.

#### B. Saran

Pada pengerjaan tugas besar, diperlukan kesadaran untuk melakukan eksplorasi dan pembelajaran mandiri dari setiap anggota kelompok. Selain itu, setiap anggota kelompok sebaiknya mampu untuk mengatur dirinya sendiri dan mampu bekerja sama dengan anggota kelompok lainnya. Hal ini mutlak dibutuhkan sebab dalam kondisi perkuliahan yang padat dan ada tugas besar-tugas besar lainnya, penyusun harus bisa mengatur waktu yang dialokasikan untuk pengerjaan semua tugas maupun kegiatan akademik lainnya.

#### C. Refleksi

Selama pengerjaan tugas besar ini, penyusun mendapatkan wawasan dan pengetahuan baru terutama tentang pemrosesan gambar melalui penggunaan library OpenCV dan PIL. Penyusun juga mengalami kendala dalam mengerjakan tugas besar ini, terutama penghitungan nilai eigen dan vektor eigen. Penyusun menyadari bahwa kerumitan yang dirasakan dalam mengerjakan latihan soal di kelas juga berlanjut di dalam implementasi program di dalam tugas besar ini. Akan tetapi, segala kendala yang dihadapi penyusun dalam pengerjaan tugas besar ini membuat penulis menjadi semakin memahami konsep dari materi yang diajarkan di kelas dan dapat merasakan langsung aplikasi dari materi tersebut pada kehidupan sehari-hari.

## DAFTAR REFERENSI

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

<https://en.m.wikipedia.org/wiki/Eigenface>

<https://medium.com/machine-learning-world/feature-extraction-and-similar-image-search-with-opencv-for-newbies-3c59796bf774>

<https://www.geeksforgeeks.org/ml-face-recognition-using-eigenfaces-pca-algorithm/>

“Nilai Eigen dan Vektor Eigen Bagian 1” by Rinaldi Munir

<https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2020-2021/Algeo-18->

Nilai-Eigen-dan-Vektor-Eigen-Bagian1.pdf “Face Recognition Using Eigenfaces (PCA Algorithm)” <https://www.geeksforgeeks.org/ml-face-recognition-using-eigenfaces-pca-algorithm/>

“Pengenalan Wajah Menggunakan Algoritma Eigenface dan Euclidean Distance

<https://www.neliti.com/publications/135138/pengenalan-wajah-menggunakan-algoritma->

eigenface-dan-euclidean-distance “Python GUI Programming With Tkinter”

<https://realpython.com/python-gui-tkinter/>

“Create UI in Python-Tkinter” <https://www.tutorialsteacher.com/python/create-gui-using-tkinter-python>

“How to build a GUI with PyQt” <https://blog.logrocket.com/how-to-build-gui-pyqt/>

## LAMPIRAN

Link github: <https://github.com/Ilhamgzzlr/Algeo02-18014>

Link video demo: [Program Pengenalan Wajah dengan Metode Eigenface](#)

