

## **BAB III**

### **PELAKSANAAN KERJA MAGANG**

#### **3.1 Kedudukan Dan Koordinasi.**

Selama proses kerja magang berlangsung berperan sebagai programmer. Yang bertugas untuk melakukan pengujian terhadap dataset citra wajah yang dibutuhkan dalam penelitian. Tugas dan penulis bekerja dengan asisten penelitian lainnya yang diawasi dan dibantu oleh Bapak Adhi Kusnadi sebagai kepala proyek penelitian.

#### **3.2 Tugas Dan Uraian Kerja Magang**

##### **3.2.1 Tugas Yang Dilakukan**

Selama 360 jam penulis melakukan kerja magang untuk melakukan pengujian dataset gambar wajah, berikut tugas-tugas yang dikerjakan:

1. Menganalisa hasil penelitian sebelumnya

Mempelajari paper penelitian sebelumnya dari berbagai macam penelitian yang berhubungan dengan *Face Recognition* dan metode yang sedang dicari. Proses analisa dilakukan bersama tim penelitian dan kemudian didiskusikan bersama ketua penelitian.

2. Mencari dataset gambar wajah yang akan digunakan pada penelitian

Dataset gambar wajah yang digunakan menggunakan dataset gambar wajah *ORL Headpose Database*

3. Mencari metode yang tepat untuk diaplikasikan pada penelitian

Metode yang akan digunakan pada penelitian ini menggunakan metode yang dapat diuji coba dengan pengenalan wajah pada frekuensi rendah. Pada proses yang dilakukan akan dilakukan bersama tim penelitian dan akan didiskusikan kepada ketua penelitian untuk mendapatkan metode yang tepat.

4. Melakukan pengujian pada dataset gambar wajah

Pengujian dataset *ORL Headpose* gambar wajah akan diproses dengan metode yang telah ditentukan oleh tim penelitian dan ketua penelitian. Setelah metode tersebut telah ditentukan proses akan dilakukan dengan menggunakan library matlab dan penggunaan aplikasi matlab yang sesuai dengan metode penelitian pengenalan wajah. Setelah metode berhasil diaplikasikan pada dataset. Akan dijalankan lah proses fungsi DCT low frekuensi, fungsi *Eigenfaces* dan proses fungsi *Face Recognition* untuk pengenalan wajah.

5. Melakukan perhitungan nilai akurasi

Tahap setelah pengujian telah berhasil akan dilakukan proses perhitungan akurasi.

## 6. Menulis laporan penelitian berupa paper

Tugas terakhir pada praktik kerja magang adalah menuliskan laporan hasil dari penelitian yang telah dilakukan pada kerja magang yang dilakukan dan menulis paper yang menggunakan bahasa Inggris.

Dalam proses penelitian dibutuhkan alat-alat pendukung yaitu perangkat keras (*Hardware*) dan perangkat lunak (*Software*). Berikut adalah spesifikasi dari *Hardware* dan *Software* yang digunakan untuk melaksanakan penelitian terhadap pengenalan wajah.

### 1. *Software*

- a. Operating System Windows 10 64-Bit
- b. Matlab R2018a
- c. Microsoft Word 2016

### 2. *Hardware*

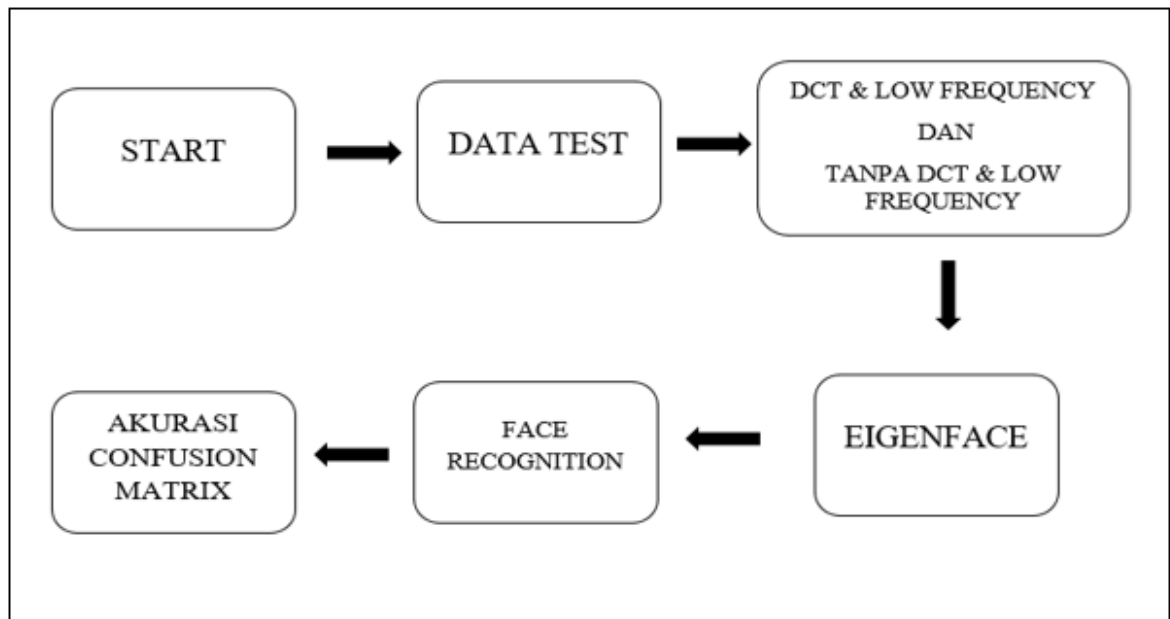
- a. Processor Intel(R) Pentium(R) G3250 Cpu @ 3.20 Ghz
- b. Ram 8.00 Gb
- c. Hard Disk Drive (Hdd) 2tb

*Tabel 1.1 Rincian Tugas Yang Dilakukan Pada Saat Kerja Magang*

| Minggu Ke- | Kegiatan   |
|------------|--|
| 1          | <ul style="list-style-type: none"> <li>- Bahas Topik Penelitian Yang Akan Dilakukan</li> <li>- Cari Informasi Dan Pelajari Mengenai Fitur Wajah</li> <li>- Pelajari Hasil Dan Metode Dari Penelitian Terdahulu</li> </ul>                |
| 2          | <ul style="list-style-type: none"> <li>- Install Aplikasi Matlab 2018a</li> <li>- Pelajari Pemakaian Aplikasi Matlab</li> <li>- Pelajari Metode Face Recognition Eigenfaces</li> </ul>   |
| 3          | <ul style="list-style-type: none"> <li>- Mengambil Sampel Dataset Orl Headpose Gambar Wajah</li> <li>- Membuat Ui Pada Matlab 2018a</li> </ul>   |
| 4          | <ul style="list-style-type: none"> <li>- Pengujian Gambar Wajah Menggunakan Dct Low</li> <li>- Mengubah Gambar Wajah Menjadi Grayscale</li> <li>- Pengujian Fungsi Eigenfaces Pada Gambar Wajah</li> <li>- Revisi Source Code</li> </ul> |
| 5          | <ul style="list-style-type: none"> <li>- Pengujian Pembuatan Deteksi Wajah Face Recognition</li> <li>- Pengujian Dataset Pada Face Recognition</li> <li>- Revisi Source Code</li> </ul>  |
| 6          | <ul style="list-style-type: none"> <li>- Hitung Akurasi Yang Di Dapat Pada Aplikasi</li> <li>- Melaporkan Hasil Source Code</li> <li>- Revisi Source Code Final</li> </ul>   |
| 7          | <ul style="list-style-type: none"> <li>- Penulisan Laporan Berupa Paper Yang Akan Di Submit Di International</li> </ul>  |
| 8          | <ul style="list-style-type: none"> <li>- Revisi Paper Yang Di Tulis</li> <li>- Menyelesaikan Paper</li> <li>- Melakukan Pengumpulan Final Revisi Paper</li> </ul>  |

### **3.2.2 Uraian Kerja Magang**

Pengujian Metode Pengenalan Wajah Dilakukan Dalam Beberapa Tahapan Atau Alur Pada Metode Ini. Alur Metode Yang Di Kerjakan Sebagai Berikut Pada Gambar 3.1



Gambar 3. 1 Alur Metode Pengujian Pengenalan Wajah

### A. Data Set

Dataset yang digunakan pada aplikasi ini di akses dari *ORL Headpose Database* (Khan, 2014) dapat dilihat pada Gambar 3.2.

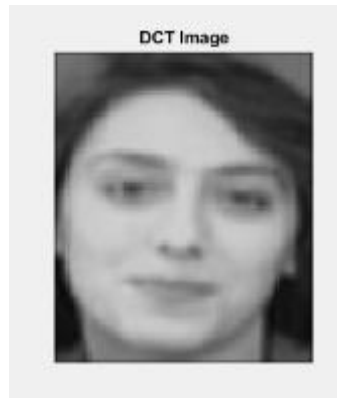


Gambar 3. 2 Dataset yang Digunakan

Dari *dataset* tersebut akan di bagi 2 menjadi *datatest* dan *datatrain*. Pada data test adalah gambar wajah yang akan diuji pada aplikasi, sedangkan data *Training* dimana data tersebut akan dijadikan target ketika data yang uji di lakukan proses *Face Recognition* akan melakukan pencocokan gambar wajah yang sama. Dataset yang dimiliki dibagi menjadi *datatest* menjadi 20 gambar wajah dan data *Training* menjadi 80 gambar wajah. Pada *dataset* yang diuji akan ada proses yang diuji dimana proses tersebut dimulai dari penampilan gambar wajah *Datatest Image*, dilanjutkan proses *DCT & Low Frequency*, akan ada proses *Eigenface* dan lanjut ke *Face Recognition*. Hasil akhir akan dilakukan proses perhitungan akurasi *Confusion Matrix*

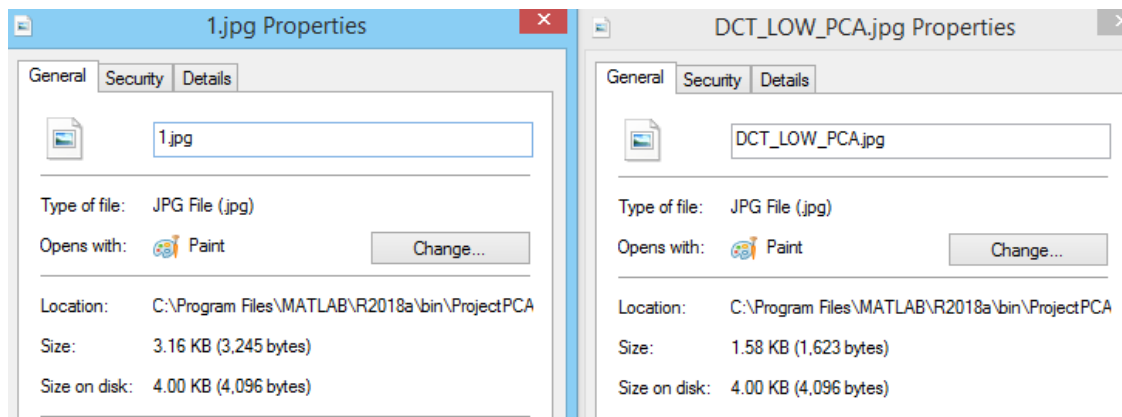
## **B. Dct & Low Frequency**

Pengujian pada *DCT (Discrete Cosine Transform) & Low Frequency* digunakan untuk menghilangkan fitur tidak penting dengan menghilangkan *Low* frekuensi. Pada *Discrete Cosine Transform* menggunakan sebuah skema menghilangkan fitur tidak penting, dimana gambar sebelumnya di transformasikan menjadi bentuk *Array* yang masih berada domain spasial ke domain *DCT*.



Gambar 3. 3 Hasil Pengaplikasian *DCT & Low Frequency*

Kemudian akan dimasukan parameter untuk mengganti koefisien menjadi *Low* frekuensi dengan nilai (0) dimana nilai tersebut untuk menghilangkan *Low Frequency* dari gambar, parameter tersebut digunakan untuk menentukan banyak *Low Frequency* yang akan di hilangkan. Setelah itu komponen yang telah dihilangkan pada gambar, gambar tersebut akan kembali ke *Domain Spasial*. Pada frekuensi koefisien DCT yang lebih rendah muncul pada kiri atas dari sebuah *Matriks* DCT. Pengujian tersebut terjadinya proses penghilangan fitur yang tidak penting dengan menghilang kan frekuensi rendah, sebelum terjadinya proses menghilangkan fitur pada gambar wajah tersebut akan diubah menjadi *Grayscale* setelah itu gambar wajah akan dikompres *File* gambar wajah tersebut akan menjadi lebih rendah sehingga hasil yang terjadi pada dataset gambar wajah bertipe *JPEG* tersebut dapat dilihat pada gambar 3.4 pada hasil tersebut awal gambar sebelum terjadi *DCT & Low Frequency* adalah 3.16 kb dan setelah di *DCT Low & Frequency* adalah 1.58 kb.



Gambar 3. 4 Hasil *DCT & Low Frequency*

### C. Eigenfaces

Metode *Eigenfaces* adalah algoritma pengenalan wajah yang berdasarkan pada *PCA (Principle Component Analysis)*. Dalam metode algoritma *Eigenface*, *Decoding* dilakukan dalam menghitung sebuah *Vektor* yaitu *Eigenvector* dan setelah itu direpresentasikan dalam sebuah *Matriks*. *Eigenvector* inilah yang dikemudian dinyatakan sebagai karakteristik sebuah wajah. Setiap wajah akan direpresentasikan dalam kombinasi linear *Eigenface*. Pada perhitungan *Eigenface* dilakukan dengan beberapa tahap yaitu menyiapkan kumpulan dataset gambar wajah *ORL database headpose* yang akan membuat sebuah himpunan *Matriks* yang ada di *Database*, kemudian mencari nilai tengah lalu setelah diperoleh akan dicari *Matriks Kovarian* dari kumpulan *Matriks*, dan menghitung nilai *Eigen* dan *Vektor Eigen*. Diakhiri dengan indentifikasi melalui kecocokan berdasarkan *Vektor* tersebut (Norhikmah 2018). Pengujian beberapa tahapan dengan menggunakan *DCT* dan tanpa *DCT*. Pada pengujian menggunakan *DCT* gambar yang akan diuji



akan melalui proses *DCT* setelah itu akan dilanjutkan ke proses *Eigenface* dan akan langsung ke proses *Face Recognition* Sedangkan pengujian tanpa *DCT* tidak jauh beda dengan menggunakan *DCT* akan tetapi dengan tanpa *DCT* gambar yang akan diuji dengan *Face Recognition* langsung dari gambar yang ditest tanpa melalui proses *DCT*, proses yang dilakukan gambar *Datatest* dimasukan lalu gambar tersebut tidak akan melakukan proses penghilangan bagian koefisien *Low Frequency* di daerah yang merupakan bagian dari *Low Frequency* sehingga langsung ke bagian proses *Eigenface* untuk menentukan proses lanjutan ke bagian *Face Recognition*.

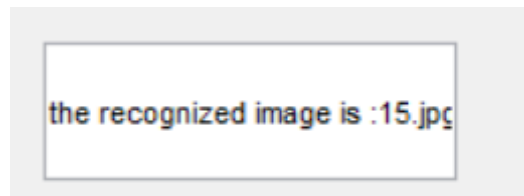
Pada proses ini terdapat pengujian menggunakan fitur *DCT* maupun tanpa *DCT*. Pengujian dimulai dari penggunaan fitur *DCT* 20:80 pada proses ini 20 *Datatest* dan 80 *Datatrain* pada pengujian tersebut mendapatkan hasil 85% dikarenakan terdapat 3 gambar yang salah dari 20 *Datatest* yang diuji. Sedangkan 30:70 proses ini menggunakan 30 *Datatest* dan 70 *Datatrain* sehingga pada pengujian tersebut mendapatkan hasil 83.33% dikarenakan terdapat 5 gambar yang salah dari 30 *Datatest* yang diuji. Selanjutnya proses tanpa menggunakan *DCT* tidak jauh beda dengan hasil dari menggunakan fitur *DCT* dikarenakan penggunaan fitur *DCT* maupun tanpa *DCT* tidak berpengaruh pada metode *Eigenfaces* sehingga hasil pengujian pada 20:80 dan 30:70 mendapatkan hasil yang sama yaitu akurasi yang didapat pada 20:80 adalah 85% dan 30:70 adalah 83.33%.

#### D. Face Recognition

Pada proses pengujian pengenalan citra wajah dilakukan pada gambar wajah yang sedang diuji dapat dilihat pada Gambar 3.5 dan Gambar 3.6.



Gambar 3. 5 Hasil Pengaplikasian *Face Recognition*



Gambar 3. 6 Hasil *File Face Recognition*

Pada hasil tersebut menampilkan gambar yang sama pada *Datatest* yang sedang diuji. Pada tahapan ini hasil dari *Datatest* yang diuji lalu ke tahap *DCT & Low Frequency* kemudian ke fungsi *Eigenface* dan tahap akhir hasil *Face Recognition* yang bisa dilihat gambar tersebut menunjukkan gambar yang sama dan menunjukkan *File* yang ada di *Data Training* data gambar ke berapa sehingga gambar tersebut sama.

Pada saat proses *Face Recognition* beberapa hasil dari *Face Recognition* masih ada hasil yang tidak sesuai dimana ada beberapa faktor yang dapat hasil tersebut tidak sesuai dengan hasil yang ditest, salah satu faktor gambar wajah dari pose gambar wajah.

#### **E. Perhitungan Akurasi**

Setelah proses pada pengujian pengenalan wajah telah selesai dilakukan , tahap akhir adalah menghitung akurasi. Adapun rumus dari *Accuracy* sebagai berikut

$$Accuracy = \frac{Jumlah\ file\ benar}{Jumlah\ seluruh\ data} \times 100 \quad (3.1)$$

Hasil pada perhitungan akurasi yang di dapat mendapatkan hasil akurasi yang cukup baik dan tinggi dari akurasi tersebut. Pada hasil akurasi dapat dilihat pada Tabel 1.2.

Tabel 1.2 Tabel Akurasi

| Percobaan DCT dan Tanpa DCT | Hasil Akurasi |
|-----------------------------|---------------|
| DCT dan tanpa DCT 80:20     | 85%           |
| DCT dan tanpa DCT 70:30     | 83.33%        |

Pada proses perhitungan metode yang digunakan cukup berhasil meningkatkan nilai akurasi pada pengenalan wajah. Pada penelitian sebelumnya mendapatkan nilai 43% dan sedangkan pada penelitian yang sedang diuji mendapatkan nilai 85% pada 80:20 dan percobaan 70:30 mendapatkan hasil 83.33% pada percobaan menggunakan *DCT* maupun tanpa *DCT*.

### **3.2.3 Kendala Yang Ditemukan**

Selama proses magang kendala yang ditemukan adalah kurangnya pengalaman dibidang fungsi *DCT & Low Frequency*, fungsi *Eigenfaces*, dan fungsi *Face Recognition* dan pengetahuan tentang metode yang digunakan. Kurangnya pengetahuan dan pengalaman dalam menggunakan aplikasi matlab sehingga pada percobaan dibutuhkan waktu lebih untuk mempelajari dan memperdalam aplikasi tersebut. Penggunaan *Library* yang digunakan dibutuhkan pembelajaran lebih kegunaan library tersebut dan penggunaan *Library* tersebut sehingga butuh waktu lebih. Dalam pembuatan aplikasi banyak sekali kesalahan pemahaman sehingga programmer selalu mengganti *Source Code* pada fitur yang diinginkan ketua proyek penelitian.

### **3.2.4 Solusi Atas Kendala Yang Ditemukan**

Solusi atas kendala yang ditemukan adalah dengan mencari tahu dan mempelajari sendiri dengan beberapa media yang sudah ada seperti buku, internet, penelitian sebelumnya, ketua penelitian dan beberapa teori yang ada. Serta saling berbagi pengetahuan yang telah ditemukan dengan tim penelitian mau pun dengan ketua penelitian. Untuk bagian kesalahan pemahaman dilakukan perbaikan diri ketika ketua proyek penelitian memberi tugas lebih fokus, membuat note yang sesuai dengan ketua proyek penelitian, membuat rekaman pada zoom meet yang dapat didengar berulang kali sehingga pada saat ketika ketua proyek memberikan tugas proyek dapat dikerjakan fitur-fitur yang diinginkan ketua proyek supaya dapat sesuai dengan ketua penelitian inginkan, tidak ada kesalahan, dan tidak mengulangi kesalahan yang sama.