# 3. ANALISIS DAN PERANCANGAN

Pada bab ini akan membahas mengenai rancangan sistem aplikasi dan metode yang digunakan dalam penelitian ini. Bab ini berisikan spesifikasi *software* dan *hardware,* data penelitian, analisa sistem, dan perancangan antarmuka sistem.

## 3.1 Spesifikasi *Hardware* dan *Software*

Berikut ini adalah beberapa *hardware* dan *software* yang digunakan untuk membuat aplikasi ini adalah sebagai berikut :

1. *Processor* : Intel Core i3 2328M
2. Sistem operasi : Windows 7 Ultimate 64-bit
3. RAM : 4 GB
4. *Hardisk* : 1024 GB
5. *Software* : Matlab 2016b (9.1.0.441655)

## 3.2 Pengumpulan Data

Data yang digunakan pada penelitian ini adalah citra daun tanaman obat. Citra yang dikumpulkan hanya 3 jenis daun untuk penelitian yaitu daun jambu biji, daun kersen, dan daun sirih. Pengumpulan data latih dilakukan secara manual yaitu mengunduh citra daun tanaman obat dari pencarian di *Google Search Image*. Seluruh data yang digunakan berjumlah 160 citra, dimana pembagian citra ditunjukan pada Tabel 3.1.

Tabel 3.1 Pembagian Data Latih Dan Data Uji

|  |  |  |
| --- | --- | --- |
| No | Dataset | Jumlah Data |
| 1  2 | Data Pelatihan  Data Pengujian | 160  15 |

Citra yang telah dikumpulkan dibagi menjadi dua bagian yaitu *training* dataset yang akan digunakan untuk memberikan pengetahuan tentang jenis daun tanaman obat pada sistem dan *testing* dataset atau data pengujian untuk mengetahui hasil dari proses identifikasi. Jenis citra daun tanaman obat yang digunakan dalam data latih terdapat pada tabel 3.2 dengan jumlah 160 citra.

Tabel 3.2 Jumlah Citra Tanaman Obat Pada Data Latih

|  |  |  |
| --- | --- | --- |
| No | Nama Daun | Jumlah Data |
| 1  2  3 | Jambu Biji  Kersen  Sirih | 60  40  60 |

Selanjutnya pada tabel 3.3 terdapat jumlah citra daun tanaman obat untuk pengujian sebanyak 30 citra dengan format *joint photographic experts group* (.jpg/.jpeg). Citra yang digunakan berasal dari pengambilan gambar menggunakan kamera *smartphone* dengan resolusi 13 *megapixel*. *Background* atau latar yang digunakan berwarna putih dengan kondisi pencahayaan terang. Pengambilan gambar dilakukan pada daun bagian depan. Sudut pengambilan gambar sama untuk semua daun yaitu dari atas.

Tabel 3.3 Jumlah Citra Tanaman Obat Pada Data Uji

|  |  |  |
| --- | --- | --- |
| No | Nama Daun | Jumlah Data |
| 1  2  3 | Jambu Biji  Kersen  Sirih | 10  10  10 |

## 3.3 **Analisa Sistem**

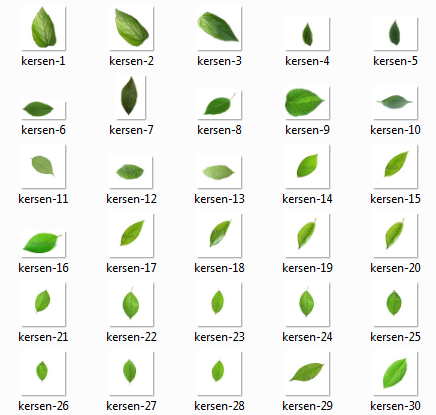
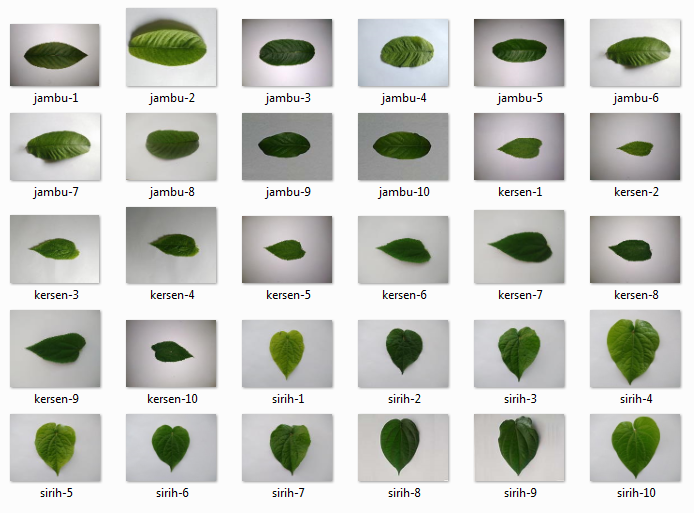
Identifikasi jenis penyakit kulit pada penelitian ini terdiri dari beberapa tahapan. Tahapan tersebut diawali dengan akuisisi citra jenis tanaman obat untuk digunakan sebagai citra pengujian dan pelatihan. Selanjutnya memasuki tahapan *preprocessing* yaitu mengubah resolusi citra (*resizing*), tranformasi warna citra, segmentasi citra untuk memperoleh nilai bentuk citra, mengubah citra kedalam citra keabuan dan melakukan proses ektraksi ciri (*feature extraction*) terhadap citra menggunakan metode *Gray Level Co-Occurrence Matrix* (GLCM) untuk memperoleh nilai tekstur citra, dan tahap identifikasi menggunakan metode *K-Nearest Neighbor* (KNN). Setelah seluruh tahap dilakukan maka aplikasi menghasilkan keluaran berupa informasi jenis tanaman obat. Adapun tahapan tersebut dapat dilihat dalam bentuk arsitektur umum pada gambar 3.1.



Gambar 3.1 Arsitektur Umum

### 3.3.1 Akuisisi Citra

Akuisisi Citra merupakan proses awal untuk mendapatkan data. Pada penelitian ini, akuisisi citra untuk data pengujian didapat dengan mengunduh gambar dari *google image*. Sedangkan untuk data pelatihan didapat dengan memotret daun yang diletakan pada kertas kosong. Pada penelitian ini, alat yang digunakan untuk memotret gambar adalah Xiaomi Redmi Note 4 dengan kamera 13 *megapixel*.

3.2 (a) Data Pelatihan (b) Data Pengujian

### 3.3.2 *Preprocessing*

Sebelum citra masukan diproses lebih lanjut, perlu dilakukan proses awal (*preprocessing*) terlebih dahulu dengan tujuan agar mendapakan hasil yang maksimal pada saat proses identifikasi. *Preprocessing* yang dilakukan pada citra adalah *resizing*. *Resizing* adalah proses merubah ukuran jumlah pixel suatu citra digital. *Resizing* dilakukan untuk memperoleh area fokus objek yaitu citra daun tanaman yang akan diidentifikasi dengan cara memanipulasi orientasi objek. Citra diubah ukurannya kedalam 200x250 piksel. Dengan besar piksel demikian, sistem akan akan lebih mudah memproses nilai ciri dari citra tersebut dan mempercepat proses komputasi.

Gambar 3.3 (a) Citra Input (b) Citra Hasil Resizing

### 3.3.3 Transformasi Warna

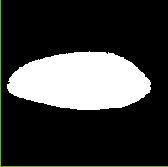
Transformasi warna dilakukan untuk menyederhanakan warna dari sebuah objek dengan merubah warna dari RGB menjadi CIELab. Ruang warna CIELab telah didefinisikan untuk menyeragamkan skala pada diagram *chromaticity* yang sebanding dengan persepsi mata manusia. Proses transformasi warna dilakukan dengan tujuan membuat hasil citra segmentasi sempurna.



Gambar 3.4 Citra Hasil Transformasi

### 3.3.4 Ekstraksi Fitur Bentuk

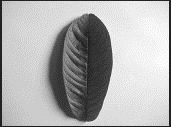
Setelah citra melalui transformasi warna kemudian citra daun akan disegmentasi dan melakukan operasi morfologi *filling holes* yaitu mengisi keseluruhan region menjadi 1. Hasil dari proses segmentasi adalah citra biner (hitam dan putih). Proses ini dilakukan untuk mendapatkan fitur bentuk dari citra daun yang akan diklasifikasi. Dari proses ini hanya akan diambil nilai *metric* dan *eccentricity* untuk dimasukan sebagai fitur bentuk dari citra daun tanaman.



Gambar 3.5 Citra Hasil Segmentasi

### 3.3.5 Ekstraksi Fitur Tekstur

Proses Ekstraksi Fitur dilakukan dengan terlebih dahulu mengubah citra dari RGB menjadi citra abu dengan memanfaatkan fungsi *rgb2gray* pada MATLAB. Seperti yang terlihat pada Gambar 3.6.



Gambar 3.6 Citra Hasil *Resizing*

Kemudian dari citra grayscale daun dilakukan proses ekstraksi fitur dengan menggunakan *Gray Level Co-Occurrence Matrix* (GLCM). Setiap data citra akan diambil 4 fitur utama dari proses ekstraksi, yaitu *contrast, correlation, energy* dan *homogeneity*. Seluruh fitur tersebut yang selanjutnya digunakan untuk proses klasifikasi. Pada Tabel 3.4 dapat dilihat contoh seluruh fitur yang dihasilkan setelah melalui proses segmentasi dan proses ekstraksi fitur dengan menggunakan metode GLCM.

Tabel 3.4 Ekstraksi Fitur Citra Daun

|  |  |  |
| --- | --- | --- |
| No | Fitur | Nilai |
| 1 | *Metric* | 0.68761 |
| 2 | *Eccentricity* | 0.9141 |
| 3 | *Contrast* | 0.17171 |
| 4 | *Correlation* | 0.94066 |
| 5 | *Energy* | 0.59682 |
| 6 | *Homogeneity* | 0.96565 |

### 3.3.6 *K-Nearest Neighbor* (KNN)

Tahap terakhir dalam pengenalan citra adalah klasifikasi. Proses klasifikasi yang dilakukan pada penetian ini adalah dengan menggunakan metode *K-Nearest Neighbor* (KNN). KNN bekerja dengan cara mengklasifikasikan suatu objek yang memiliki kemiripan paling dekat dengan objek lainnya. KNN memiliki atribut yang diinisalisasikan sebagai k, yaitu jumlah nilai tetangga yang dijadikan acuan pada klasifikasi KNN. Tahapan dari klasifikasi menggunakan metode KNN adalah sebagai berikut :

* Tahap 1: Penentuan nilai k

Penentuan nilai k yang digunakan dalam klasifikasi tidak memiliki aturan yang baku.

* Tahap 2: Perhitungan jarak antar data latih dan data uji

Teknik perhitungan jarak yang digunakan dalam metode klasifikasi KNN ini adalah *Euclidean Distance*. Jarak dihitung dari semua data latih ke data uji. Contoh perhitungan jarak *euclidean distance* dengan menggunakan metode KNN akan dijelaskan pada bab selanjutnya dengan menggunakan data *real*.

* Tahap 3: Pengurutan data hasil perhitungan

Jarak yang telah didapatkan pada tahap sebelumnya kemudian diurutkan dari jarak yang paling dekat sampai jarak yang paling jauh (*ascending*).

* Tahap 4: Klasifikasi berdasarkan nilai k

Tahap selanjutnya yaitu klasifikasi berdasarkan nilai k. Nilai k atau tetangga terdekat merupakan acuan dari klasifikasi KNN. Contoh dari penerapan k adalah jika nilai k = 3, ini berarti jumlah tetangga terdekat yang dihitung adalah sejumlah 3 tetangga terdekat. Dari 3 tetangga terdekat tersebut, diukur mana nilai yang paling mirip dengan objek yang diuji pada pengenalan pola, contoh penerapan K-NN dapat dilihat pada gambar 3.7.



Gambar 3.7 Ilustrasi Metode KNN

Gambar 3.7 menjelaskan bahwa terdapat 3 objek, objek pertama yaitu “kotak biru” yang diklasifikasikan sebagai kelas daun jambu biji, lalu objek kedua yaitu “bulat hitam” yang diklasifikasikan sebagai daun kersen, dan objek ketiga yaitu “segitiga merah” yang tidak diketahui termasuk dalam kelas yang mana. Gambar tersebut menampilkan jumlah nilai k = 3 dengan merepresentasikan 3 tertangga tertangga terdekat yang berada didalam lingkaran. Objek yang ada di dalam lingkaran dihitung mana yang paling banyak memiliki kemiripan dengan objek yang sudah diketahui. Gambar di atas memiliki 2 objek “bulat hitam” dan 1 “kotak biru” di dalam lingkaran tersebut, artinya bisa disimpulkan bahwa objek “segitiga merah” termasuk ke dalam kelas kedua yaitu daun kersen.

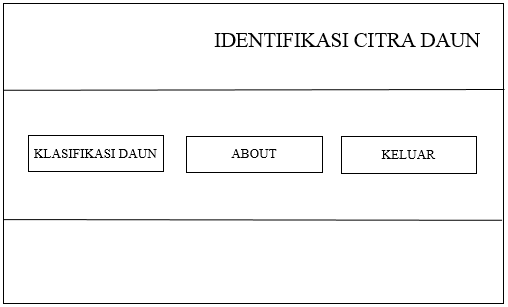
## 3.4 Perancangan Antarmuka Sistem

Perancangan antarmuka merupakan gambaran umum tentang tampilan halaman awal, tampilan halaman klasifikasi, tampilan halaman about, dan tampilan halaman informasi daun.

### 3.4.1 Rancangan Tampilan Halaman Awal

Pada tampilan halaman awal aplikasi terdapat judul penelitian dan logo di bagian atas. Pada form ini terdapat 3 tombol seperti yang ditunjukan pada gambar 3.8 Rincian tombol yang digunakan dalam aplikasi adalah sebagai berikut :

1. Tombol “klasifikasi” adalah tombol yang digunakan untuk membuka halaman klasifikasi.
2. Tombol about adalah tombol yang digunakan untuk membuka halaman about.
3. Tombol keluar adalah tombol yang digunakan untuk keluar aplikasi.

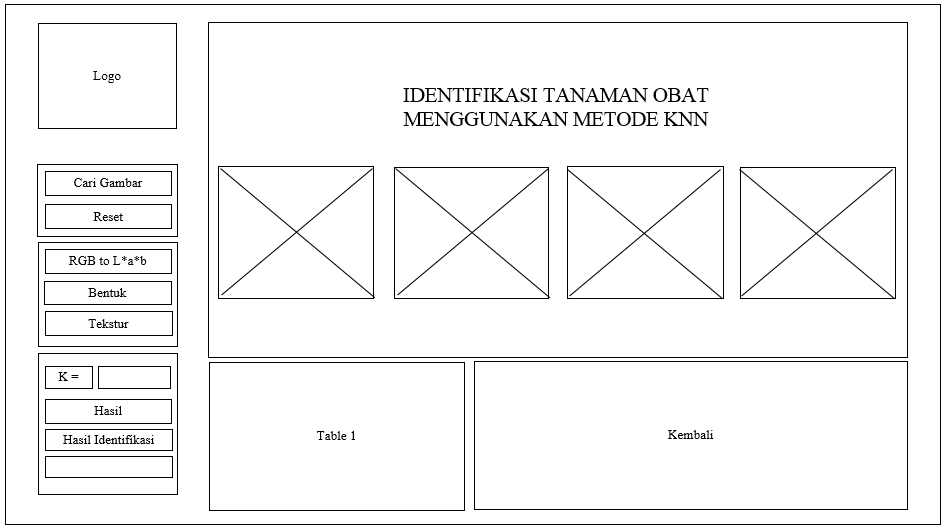


Gambar 3.8. Rancangan Tampilan Halaman Awal

### 3.4.2 Rancangan Tampilan Halaman Klasifikasi Daun

Halaman klasifikasi daun merupakan tampilan utama sistem untuk menguji aplikasi. Halaman ini terdiri beberapa tombol untuk memilih gambar data uji, transformasi warna, ekstraksi ciri, melakukan identifikasi, kembali ke halaman awal, dan menghapus semua data. Rincian tombol yang digunakan dalam aplikasi adalah sebagai berikut :

1. Tombol cari gambar digunakan untuk memilih dan membaca jenis daun dari suatu direktori serta melakukan proses *resizing*. Setelah citra dipilih, maka citra akan ditampilkan.
2. Tombol RGB *to* L\*a\*b digunakan untuk transformasi warna citra dari warna RGB menjadi CIELab dan hasil gambar transformasi warna akan ditampilkan.
3. Tombol bentuk digunakan untuk melakukan proses segmentasi dan mengambil nilai ciri bentuk dari citra yaitu *metric* dan *eccenricity*. Nilai ekstraksi ciri akan ditampilkan pada tabel bagian bawah aplikasi dan hasil gambar berupa citra biner akan ditampilkan.
4. Tombol tekstur digunakan untuk melakukan proses *grayscalling* dan mengambil nilai ciri tekstur dari citra yaitu *contrast*, *correlation*, *energy*, dan *homogeneity*. Nilai ekstraksi ciri akan ditampilkan pada tabel bagian bawah aplikasi dan hasil gambar berupa *grayscalling* akan ditampilkan.
5. Panel citra terdiri atas “Citra RGB” yang menampilkan hasil *resizing*, “Citra L\*a\*b” yang menampilkan citra hasil tranformasi warna, “Citra Biner” yang menampilkan hasil citra hitam putih, dan “Citra *Grayscale*” yang menampilkan citra hasil *grayscaling*.
6. Edit text nilai k digunakan sebagai tempat untuk user memasukan nilai k.
7. Tombol hasil digunakan untuk melakukan proses identifikasi citra yang telah dimasukan sebelumnya dan menampilkan hasil identifikasi pada kotak dialog serta menampilkan form berupa informasi dari citra yang telah diidentifikasi.



Gambar 3.9 Rancangan Tampilan Halaman Klasifikasi Daun

### 3.4.3 Rancangan Tampilan Halaman Informasi Daun

Halaman ini berfungsi untuk menampilkan informasi mengenai manfaat dari jenis daun yang telah diidentifikasi. Pada halaman ini terdapat tombol kembali yang digunakan untuk menutup halaman informasi dan menampilkan halaman klasifikasi daun.



Gambar 3.10 Rancangan Tampilan Halaman Informasi Daun

### 3.4.4 Rancangan Tampilan Halaman *About*

Halaman *about* berfungsi untuk menampilkan informasi mengenai sistem aplikasi yang dibuat dan identitas pembangun sistem aplikasi ini. Pada halaman ini terdapat tombol kembali yang digunakan untuk menutup halaman *about* dan membuka halaman awal aplikasi.

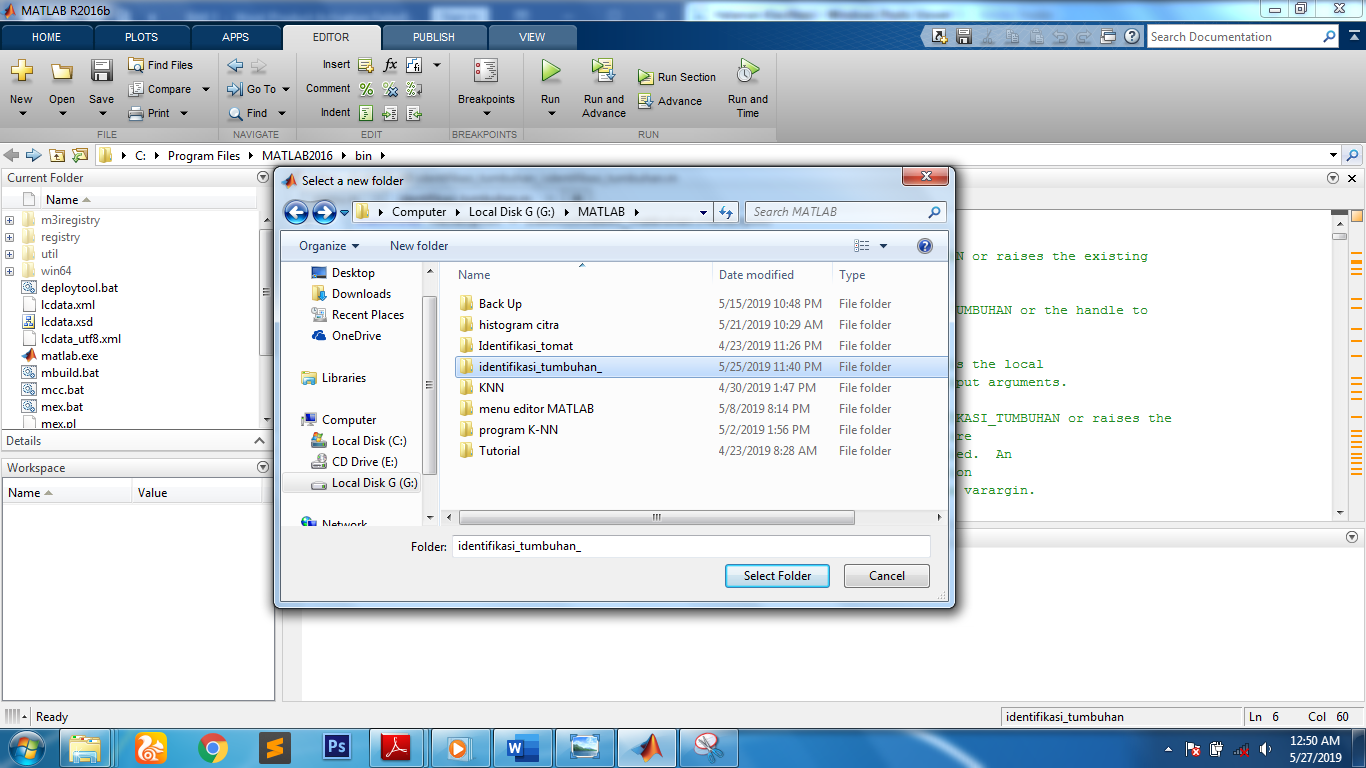


Gambar 3.11 Rancangan Tampilan Halaman *About*

## 3.5 Implementasi Sistem

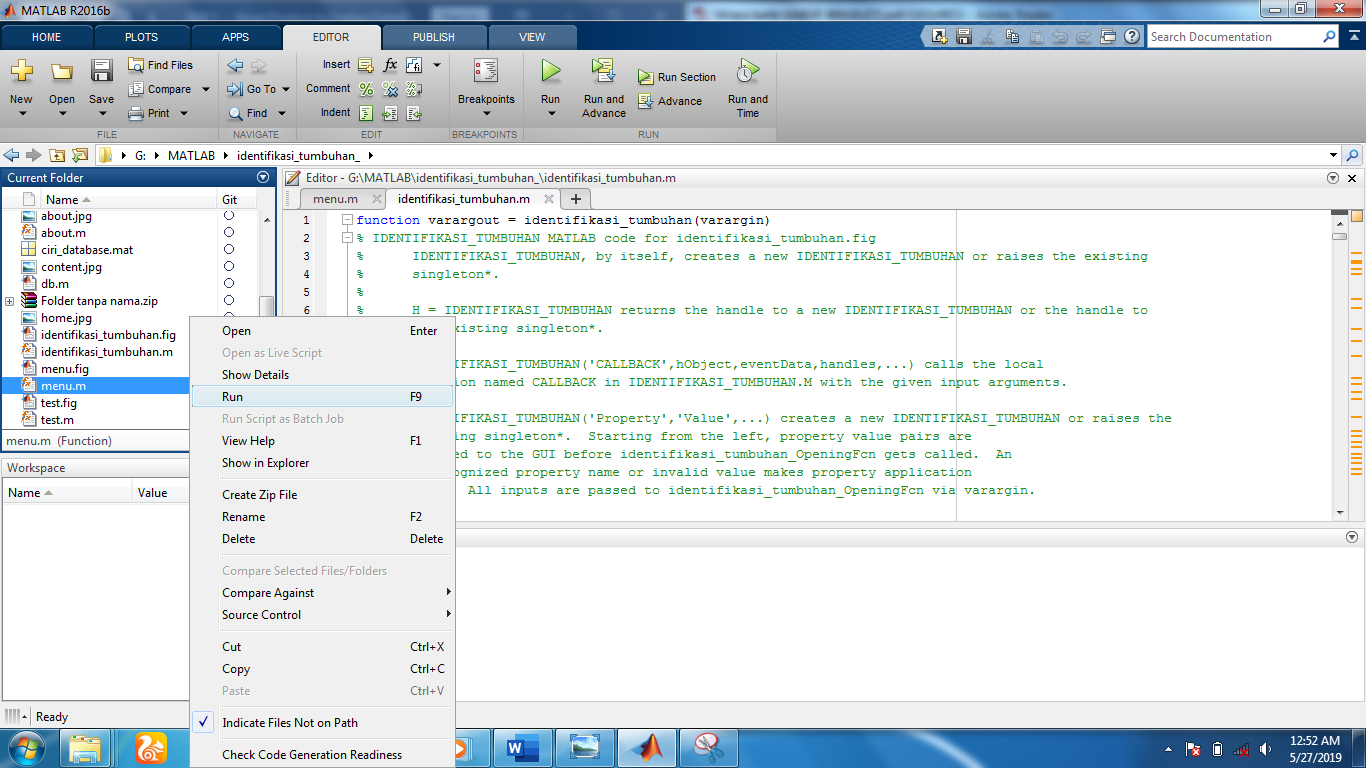
Berikut ini akan dijelaskan mengenai langkah atau alur kerja sistem secara umum untuk menjalankan sistem ini. Sistem ini akan dijalankan dengan menggunakan Matlab R2016b tetapi sebelumnya ada beberapa hal penting yang perlu diperhatikan, yaitu persiapan berkas (*file*) yang akan digunakan. Dalam Matlab untuk mengakses suatu *file* yang berkaitan, maka *file* *listing* program tersebut harus disimpan dalam satu *folder*. Setelah *folder* telah dipersiapkan langkah berikutnya adalah sebagai berikut :

1. Menjalankan program Matlab R2016b.
2. Setelah aplikasi Matlab R2016b diaktifkan, langkah berikutnya adalah mengubah direktori aktif “/work” pada Matlab R2016b ke dalam direktori yang telah dipersiapkan sebelumnya (dalam penelitian ini path yang digunakan adalah G:\MATLAB\identifikasi\_tumbuhan\_).



Gambar 3.12 Petunjuk Pengubahan Direktori Yang Ingin Diaktifkan

1. Dari perubahan tersebut dapat langsung klik kanan pada *file* yang akan dijalankan lalu pilih *Run*. Maka aplikasi Matlab R2016b akan menjalankan program sesuai filih yang dipilh.



Gambar 3.13 Petunjuk Menjalankan Program

### 3.5.1 *Form* Halaman Awal



Gambar 3.14 Form Halaman Awal

*Form* halaman awal merupakan *form* yang menyediakan menu-menu yang terdapat dalam aplikasi. Dalam *form* menu utama ini terdiri dari tiga macam tombol diantaranya klasifikasi daun, about dan keluar.

### 3.5.2 *Form* Halaman Klasifikasi Daun



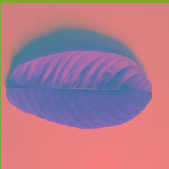
Gambar 3.15 *Form* Halaman Klasifikasi Daun

*Form* ini merupakan form utama dalam sistem. Pada *form* ini pengguna pada umumnya dapat melakukan pencarian data citra daun dengan menggunakan gambar. Dimana langkah awal yang harus dilakukan oleh pengguna pada umumnya adalah memasukkan gambar atau data citra yang ingin dicari dengan menekan tombol “Cari Gambar”, maka sistem akan membuka *form open file image* dan pengguna dipersilakan untuk memilih data citra atau gambar yang ingin dicari dalam direktori. Setelah data citra terpilih, maka axes 1 akan menampilkan gambar atau citra yang telah dipilih sebelumnya.



Gambar 3.16 Proses Menampilkan Citra RGB

Selanjutnya pengguna menekan tombol RGB to L\*a\*b untuk melakukan proses transformasi warna. Ketika tombol RGB to L\*a\*b telah dipilih maka axes 2 akan menampilkan proses transformasi warna.



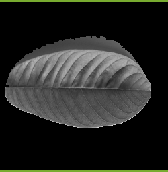
Gambar 3.17 Proses Transformasi Warna

Setelah melakukan proses transformasi warna, maka pengguna dipersilahkan untuk menekan tombol bentuk. Tombol ini berfungsi untuk melakukan proses ekstraksi fitur bentuk pada citra. Hasil dari proses ekstraksi fitur bentuk akan ditampilkan pada axes 3 dalam bentuk citra biner serta tabel 1 akan menampilkan nilai *metric* dan *eccentricity*.



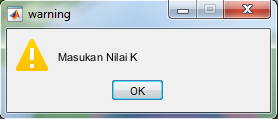
Gambar 3.18 Proses Ekstraksi Fitur Bentuk

Kemudian pengguna menekan tombol tekstur untuk melakukan proses ekstraksi fitur tekstur pada citra. Setelah proses ekstraksi teksur dijalankan maka hasil citra akan ditampilkan pada axes 4 dalam bentuk citra *grayscall* serta tabel 1 akan menampilkan nilai *contrast*, *correlation*, *energy* dan *homogeneity*.



Gambar 3.19 Proses Ekstraksi Fitur Tekstur

Setelah melakukan semua proses ekstraksi, maka langkah selanjutnya pengguna memasukan nilai k pada textbox 1 dan menekan tombol hasil. Apabila pengguna tidak memasukan nilai k, maka sistem akan menampilkan pesan kesalahan sebagai berikut.



Gambar 3.20 Pesan Kesalahan Jika Tidak Memasukan Nilai K

Tombol hasil berfungsi untuk melakukan proses klasifikasi dan akan menampilkan informasi berupa hasil identifikasi pada textbox 2.



Gambar 3.21 Form Halaman Informasi Daun

### 3.5.3 *Form* Halaman Informasi Daun



Gambar 3.22 Form Halaman Informasi Daun

*Form* ini berfungsi untuk menampilkan informasi mengenai manfaat dari jenis daun yang telah diidentifikasi. Pada *form* ini terdapat tombol kembali yang digunakan untuk menutup halaman *about* dan menampilkan halaman klasifikasi daun.

### 3.5.4 *Form* Halaman *About*



**Kembali**

Gambar 3.23 Form Halaman About

Form ini berfungsi untuk menampilkan informasi kepada pengguna mengenai program atau sistem yang digunakan dan juga menampilkan identitas pembangun sistem. Pada *form* ini terdapat tombol kembali yang digunakan untuk menutup halaman *about* dan membuka halaman awal aplikasi.