

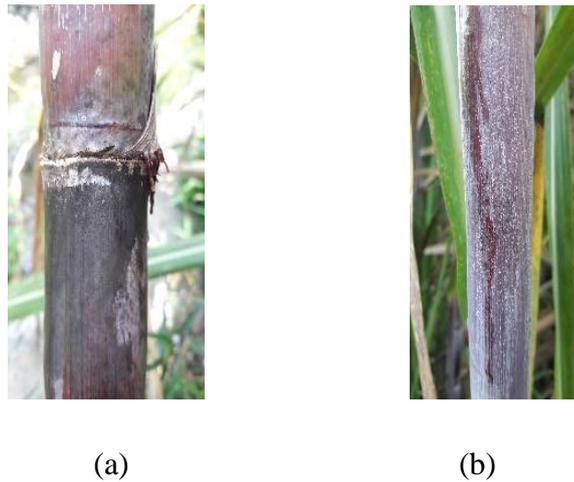
## **BAB III**

### **ANALISIS DAN PERANCANGAN SISTEM**

#### **3.1. Analisis Sistem**

Permasalahan yang ditimbulkan dalam pengelompokan kematangan citra tebu adalah penentuan kematangan batang citra tebu, mana citra tebu yang matang dan citra tebu mentah. Pada tahapan pengumpulan data dilakukan pengambilan citra secara langsung. Setelah itu citra akan diolah untuk menentukan pembeda citra tebu matang dan citra tebu mentah, dari beberapa citra tebu matang dan citra tebu mentah kemudian akan dijadikan sebagai gambar acuan dan disimpan sebagai bentuk database gambar. Tentunya jika kita ingin mengklasifikasikan citra tebu matang dengan citra tebu mentah berdasarkan tekstur batang kita harus bisa membedakan mana citra tebu matang dan citra tebu mentah, sehingga pada saat melakukan klasifikasi citra tebu matang dengan citra tebu mentah tidak terjadi kesalahan.

Dalam aplikasi ini, sistem akan dibagi dalam 2 tahapan, yakni tahapan yang pertama adalah pengambilan citra tebu dan yang ke dua adalah penapisan tekstur. Berikut adalah ciri-ciri yang menjadi dasar untuk membedakan citra tebu yang siap panen (tebu matang) dan citra tebu belum bisa dipanen (tebu mentah), adapun ciri dari citra tebu matang yaitu bertekstur kasar, berwarna merah tua kekusaman serta mempunyai akar pada batang tebu, sedangkan untuk ciri dari citra tebu mentah yaitu bertekstur halus, berwarna merah muda dan belum memiliki akar pada batangnya. Didalam sebuah perkebunan tebu, tentunya tidak hanya terdapat jenis tebu saja, tetapi didalamnya jelas ditumbuhi beberapa tumbuhan penyeimbang buat tanaman disekitarnya, misalnya rerumputan, pepohonan, bebatuan atau mungkin dalam sebuah perkebunan itu dekat dengan kawasan hutan, atau bisa jadi dalam sebuah petak perkebunan tersebut terjadi campuran tumbuhan antara citra tebu matang dan citra tebu mentah. Seperti pada gambar dibawa ini:



**Gambar 3.1** Pengklasifikasian citra tebu matang dengan citra tebu mentah

Keterangan:

- (a). Adalah contoh citra tebu matang
- (b). Adalah contoh citra tebu mentah

Pengklasifikasian citra tebu berfungsi untuk mengetahui yang termasuk citra tebu matang dan citra tebu mentah. Adapun dalam perencanaan dan perancangan pembuatan perangkat lunak memanfaatkan bahasa pemrograman MATLAB Versi 8.1.0.604 (R2013) sebagai perangkat lunak yang dapat membantu menyelesaikan masalah pada penelitian ini.

### 3.2 Hasil analisis

Hasil analisis yang dapat dilakukan dari sistem klasifikasi dapat membedakan tebu matang dengan tebu mentah. Pembuatan sistem klasifikasi kematangan tebu berdasarkan tekstur menggunakan metode KNN diperlukan data pembelajaran, data tersebut diperoleh dari *capturing* tebu untuk mendapatkan *citra* tebu, selanjutnya dilakukan *preprocessing* pada citra tebu, dan dari hasil *preprocessing* citra dilakukan ekstraksi ciri menggunakan *Gray Level Co-occurrence matrix* yang menghasilkan nilai fitur. Sebelum data citra di *preprocessing* secara manual di karenakan hasil citra yang di dapat tidak bisa di segmentasi secara total di perangkat lunak Matlab. Dari nilai fitur tersebut nantinya akan diolah dengan metode menggunakan KNN ( *K-Neares Neighbor* ). Hasil yang diperoleh dari perhitungan metode KNN ( *K-Neares Neighbor* ) berupa hasil klasifikasi citra tebu yang dapat membantu petani dalam membedakan tebu.

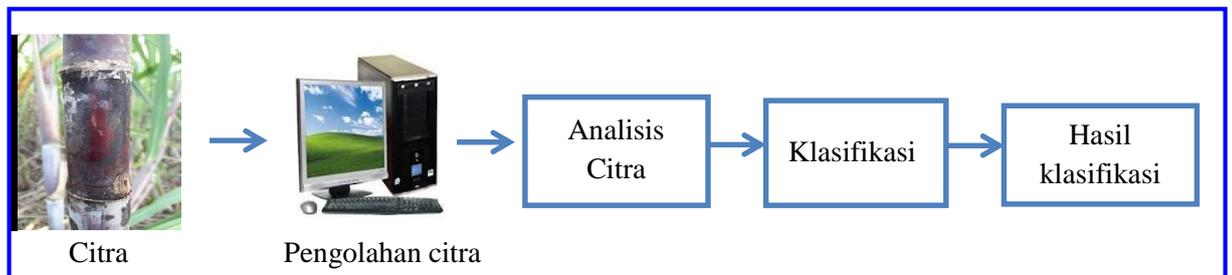
Berdasarkan uraian tersebut maka pada skripsi ini ingin membuat suatu aplikasi dengan judul “**Klasifikasi Kematangan Tebu Berdasarkan Tekstur Menggunakan Metode KNN**”.

### 3.3. Deskripsi Sistem

Deskripsi sistem dimaksudkan untuk memberikan gambaran secara umum tentang *software* yang dibuat dan juga akan dibuat dan juga *hardware* yang di butuhkan. Hal ini berguna untuk menunjang *software* yang akan dibuat, sehingga kebutuhan akan *software* tersebut dapat diketahui sebelumnya.

#### 3.3.1. Gambaran Umum Sistem

Didalam pembuatan suatu sistem, diperlukan adanya perancangan sistem. Perancangan sistem ini dimaksudkan untuk memberikan gambaran secara umum tentang bagaimana proses dimulai hingga mampu menyelesaikan permasalahan yang dibuat. Berikut adalah gambaran dari perancangan sistem tersebut:

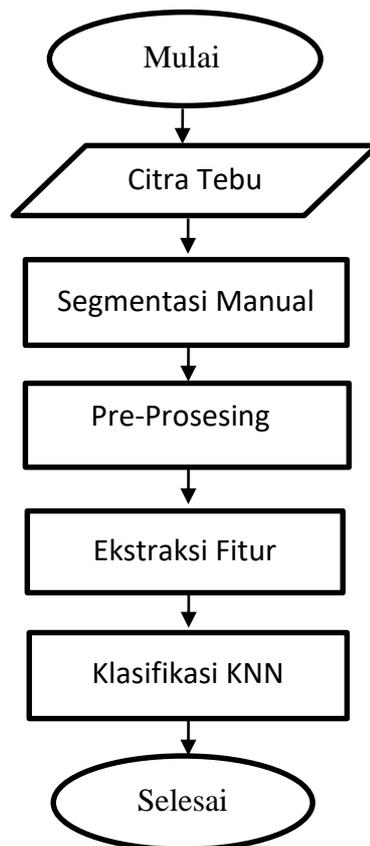


**Gambar 3.2** Perancangan Umum Sistem

Dari gambar 3.1 diatas menunjukkan sistem yang akan dibuat menggunakan kamera HP Samsung Galaxy J7 Prime dengan fitur kamera 16 MP, resolusi 1080x1920 pixels sebagai bahan untuk mengambil gambar sehingga bisa dilakukan pemrosesan data menggunakan proses pengolahan citra dalam hal ini memanfaatkan bahasa pemrograman MATLAB sebagai media pemrosesan data digital dan juga menggunakan sistem operasi *Microsoft Windows 7 Ultimate SPI 32-bit*. Kemudian dilakukan proses analisis citra untuk menghasilkan citra atau objek yang dapat diklasifikasi sesuai dengan syarat dan kondisi yang sudah ditetapkan sebelumnya.

### 3.4. Perancangan Sistem

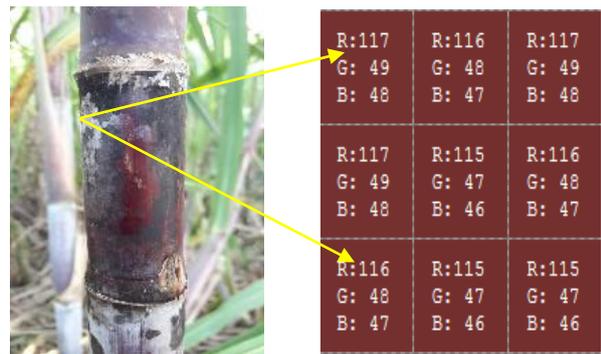
Fungsi dari *flowchart* ialah memberikan gambaran tentang program yang akan dibuat pada penelitian ini, pada bagian ini akan dijelaskan bagaimana proses pengolahan data yang berupa citra dapat diolah menggunakan proses pengolahan citra hingga dapat menghasilkan kemampuan mengidentifikasi suatu objek. Dapat dilihat pada gambar 3.3.



**Gambar 3.3** *flowchart* Perancangan Sistem

a. Citra Tebu

Citra Tebu yang digunakan dalam penelitian ini adalah data citra tebu yang terlihat pada gambar 3.4.



(a)

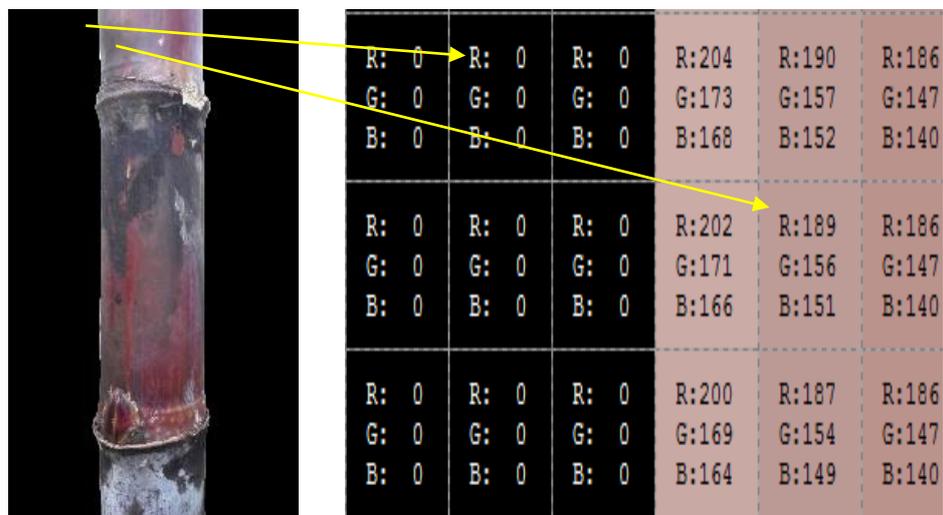
(b)

**Gambar 3.4.** (a). Citra Tebu, (b). Citra RGB Tebu

Dalam citra RGB 1 pixel mempunyai 3 kanal yaitu kanal RED, GREEN dan BLUE. Untuk memproses citra dengan ukuran 1152 x 2048 akan dihitung  $1152 \times 2048 \times 3 = 7.077.888$ , hal ini menjadikan komputasi menjadi sangat gerial maka dipermudah penyerderhanaan proses.

b. Segmentasi Manual

Citra RGB kemudian di segmentasi secara manual dengan *Adobe Photoshop CC 2018* untuk mempermudah gambar, dimana background dirubah menjadi hitam dengan  $R=0, G=0, B=0$ . Hasil segmentasi terlihat pada gambar 3.5.



(a)

(b)

**Gambar 3.5.** (a). Segmentasi, (b). Citra Segmentasi RGB Tebu

c. Premrosesan Data Awal (*Pre-pocessing*)

Hasil citra yang telah tersegmentasi kemudian dikonversi menjadi grayscale untuk mendapatkan citra gray (abu-abu). Dengan proses *grayscale* ini dapat mempermudah untuk memproses gambar lebih lanjut, karena citra gray hanya memiliki satu nilai kanal pada setiap pixelnya yang bernilai 0-255. Cara menghitung rata-rata setiap elemen warna, yaitu *Red*, *Green* dan *Blue*. Secara matematis, perhitungan untuk konversi citra warna menjadi keabuan dapat dirumuskan sebagai berikut

$$R \times 0,2989 + G \times 0,5870 + B \times 0,1140$$

R: 1	R: 1	R: 0	R:140	R:170	R:152
G: 1	G: 2	G: 2	G:144	G:176	G:159
B: 0	B: 0	B: 0	B:130	B:164	B:151
R: 0	R: 2	R: 1	R:136	R:164	R:145
G: 0	G: 3	G: 4	G:140	G:170	G:152
B: 0	B: 0	B: 0	B:126	B:158	B:144
R: 0	R: 3	R: 2	R:126	R:157	R:139
G: 0	G: 4	G: 5	G:129	G:163	G:146
B: 0	B: 0	B: 0	B:118	B:151	B:139
R: 0	R: 2	R: 0	R:111	R:152	R:139

0	1	1	1	141	173	156	140
0	0	2	3	137	167	149	139
0	0	3	4	127	160	143	139
0	0	2	1	112	155	143	139
0	2	2	1	96	153	147	141
0	4	2	1	84	152	149	142

(a). Citra RGB

(b). Citra Grayscale

Keterangan:

(a). Citra RGB memiliki 3 kanal dalam setiap pixelnya yaitu R (Red) G (Green) B (Blue) sehingga didapatkan bit dalam satu kanalnya  $((2^8)^3) = 16.777.216$

(b). Citra *gray* hanya memiliki satu kanal pada setiap *pixel*nya yang bernilai antara 0-255.

d. Binerisasi

Setelah itu dilanjutkan dengan binerisasi citra pada proses ini, citra akan dirubah menjadi dua macam intensitas saja, yaitu 0 dan 255, atau sering digunakan istilah 0 dan 1. Hasil citra binerisasi terlihat pada gambar 3.6.



**Gambar 3.6.** Gambar Citra Binerisasi

e. *Histogram Equalisation*

Untuk melakukan proses ini digunakan *threshold*. Nilai *threshold* digunakan untuk memisahkan antara latar belakang (hitam) dan objek (putih) pada citra. Setelah proses *thresholding* langkah berikutnya adalah memperbaiki citra dengan meratakan drajat keabuannya dengan menggunakan *Histogram Equalisation*. Langkah terakhir pada perbaikan citra ini adalah segmentasi, yaitu untuk memudahkan dalam menganalisa citra dengan mengalikan citra biner dengan citra *Histogram Equalisation* sehingga citra yang di dapat sesuai dengan apa yang dibutuhkan untuk proses ekstraksi ciri. Hasil histeq citra tebu terlihat pada gambar 3.7.



**Gambar 3.7.** *Histogram Equalisation* Citra Tebu

f. Proses penentuan acuan tekstur

Pada proses penentuan acuan tekstur pertama-tama, citra inputan (Citra RGB) yang di segmentasi manual akan dikonversi kedalam citra *gray*, citra *gray* sendiri merupakan citra digital yang hanya memiliki satu nilai kanal pada setiap pixelnya.

Proses pengolahan citra untuk mendapatkan nilai ciri pada proses analisis, pada proses ini sudah didapatkan segmentasi manual tebu, langkah berikutnya adalah *preprocessing* dan menganalisa citra dari ciri

tekstur yang tampak pada tebu, dengan menggunakan *Gray Level Co-occurrence matrix*.



Hasil analisis *Gray Level Co-occurrence matrix* diklasifikasikan dengan metode KNN.

Proses kemudian dilanjutkan dengan melakukan perhitungan metode *Co-occurrence matrix* yang menghasilkan nilai *ASM*, *CON*, *COR*, *VAR*, *IDM* dan *ENT*, setelah itu akan dilakukan ekstraksi nilai tekstur hasil ekstraksi ciri tekstur ini dapat dilihat pada Tabel 3.1.

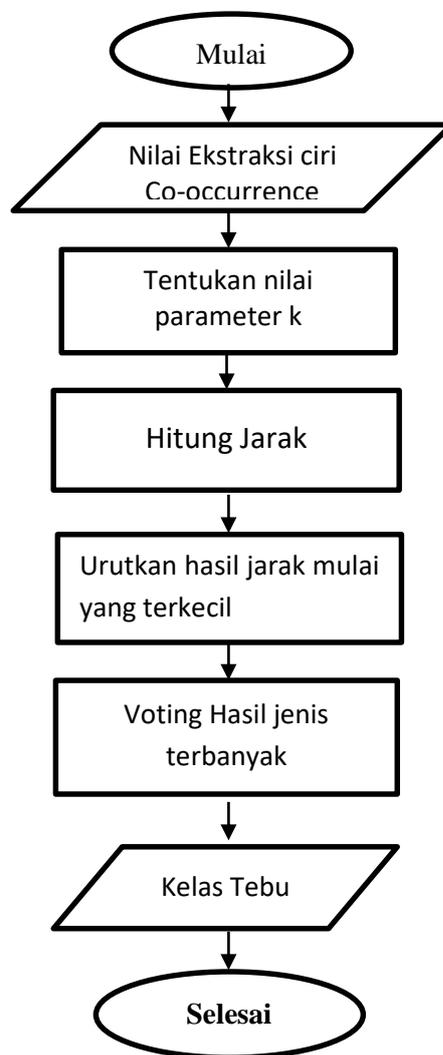
**Tabel 3.1.** Hasil Ekstraksi ciri sebagai data acuan tekstur

No	ASM	CON	COR	VAR	IDM	ENT
1.jpg	0,484192	79,575082	4767,067356	3430,480656	0,76244	4,692197
2.jpg	0,396719	44,895256	4777,686692	3154,961802	0,7664401	5,096919
3.jpg	0,471974	37,127387	3342,792262	2435,316924	0,801329	4,441729
4.jpg	0,447449	54,6674	5323,614748	3802,875362	0,757442	5,018006
5.jpg	0,397816	41,771516	4762,861561	3146,344343	0,764946	5,085509
71.jpg	0,431664	34,949669	8979,621701	5944,853908	0,731533	4,870189
72.jpg	0,440375	127,344146	8437,697387	5667,040743	0,711166	5,260115
73.jpg	0,50313	180,572407	6039,267548	4357,557715	0,742908	4,81116
74.jpg	0,502977	199,890711	6091,003378	4390,381213	0,742055	4,833228
75.jpg	0,521246	121,527456	6816,464758	4952,976449	0,766273	4,4077239

g. Proses klasifikasi menggunakan metode KNN

Proses pengelompokkan untuk mengetahui manakah termasuk tebu matang atau mentah, dilakukan menggunakan metode KNN. Setelah melalui *pre-processing* kemudian citra di ekstraksi menggunakan *Co-Occurrence Matrix* dan mendapatkan beberapa *variable* nilai (fitur-titur dari *Co-Occurrence Matrix* yang menghasilkan nilai *ASM* (*Anguler*

*Second Moment*), *Contrast*, *Corellation*, *Variance*, *IDM (Invers Different Moment)*, dan *Entropy*)) kemudian dilakukan pengelompokan menggunakan rumus dari KNN. Proses KNN dapat dilihat seperti pada gambar 3.8.



**Gambar 3.8.** Proses KNN Untuk Penentuan Kelas Kematangan Tebu

Pada proses klasifikasi tahapan dimulai dengan melakukan penginputan citra RGB, kemudian dilakukan *pre-processing* data. Setelah syarat dan atau kondisi terenuhi, proses dilanjutkan pada mengganti background citra jadi hitam, citra RGB dijadikan citra *rgb2gray*, setelah itu citra *rgb2gray* di konvers di ekstraksi ciri.

Proses kemudian beralih pada pendekatan menggunakan metode *Co-Occurrence Matrix* yang menghasilkan nilai *ASM (Angular Second Moment)*, *Contrast*, *Corellation*, *Variance*, *IDM (Invers Different Moment)* dan *Entropy*.

Proses selanjutnya pada pengelompokkan tebu menggunakan metode KNN. Dimana acuan datanya dari hasil ekstraksi ciri tekstur yang menggunakan konsep *Co-Occurrence Matrix*. Berikut Setelah diketahui nilai/hasil ekstraksi ciri itu, kemudian dicari jarak ecludiannya, setelah menghasilkan jarak ecludiannya, disorting berdasarkan jarak terdekat. Kemudian masuk proses KNN dimana KNN ini bekerja mencari jarak yang paling dekat dari pada data acuan dengan data uji dengan menggunakan voting terbanyak dari  $k$  yang telah ditentukan.

Berikut Langkah-langkan Proses Pengujian Algoritma KNN :

1. Tentukan Data Latih, Data latih terdapat pada Tabel 3.1
2. Tentukan Data Uji, Data uji terdapat pada Tabel 3.2

**Tabel 3.2.** Data Uji

No	ASM	CON	COR	VAR	IDM	ENT
Mtg	0,406194977	212,4915303	8912,64068	5,903573427	5,903573427	5879,095837
Mth	0,361661701	49,17081283	5173,729598	5,700538829	0,733512261	0,733512261

3. Tentukan Parameter K, untuk proses kali ini kita menggunakan paramter  $k=3$ ,  $k=5$ ,  $k=7$ ,  $k=9$
4. Kemudian cari jarak ecludiannya menggunakan rumus dan (2.8). Berikut adalah contoh perhitungan menggunakan ecludiean. Perhitungan dilakukan dengan 1 data latih dan 1 data uji :

$$\begin{aligned}
 d1 &= ((uji1_{fitur1}-latih_{fitur1})^2 + (uji1_{fitur2}-latih_{fitur2})^2 + (uji1_{fitur3}-latih_{fitur3})^2 + \dots \\
 &\quad + (uji1_{fitur10}-latih_{fitur10})^2)^2 \\
 &= ((0,406194977-0,484192)^2 + (212,4915303-79,575082)^2 + \\
 &\quad (8912,64068-4767,067356)^2 + (5,903573427-3430,480656)^2 + \\
 &\quad (5,903573427-0,76244)^2 + (5879,095837-4,692197)^2)^2 \\
 &= 7964,911659
 \end{aligned}$$

Perhitungan manual diatas merupakan perhitungan data uji 1 dengan data latih 1. Selanjutnya kita harus menghitung nilai euclidean dari data uji 1 terhadap data latih 2, data uji 1 dan data latih 3 dan seterusnya sehingga perhitungan nilai euclidean dari data uji 1 terdapat pada tabel 3.3 dibawah ini.

**Tabel 3.3.** Tabel nilai euclidean pada citra tebu

No.	Citra Latih	Euclidean	Kelas
1	1.jpg	7964,911659	Mtg
2	2.jpg	7845,150759	Mtg
3	3.jpg	8453,851202	Mtg
4	4.jpg	7863,06729	Mtg
5	5.jpg	7849,598878	Mtg
6	71.jpg	8355,458836	Mth
7	72.jpg	8172,103053	Mth
8	73.jpg	7855,025962	Mth
9	74.jpg	7854,452742	Mth
10	75.jpg	7961,642483	Mth

5. Urutkan jarak yang terbentuk (urut naik)

Setelah diketahui jarak setiap data pelatihan, maka data di urutkan atau di sorting mulai dari data terkecil sampai data yang paling besar lalu cari jarak yang paling kecil (terdekat) dengan data pelatihan, terlihat ada Tabel 3.4.

**Tabel 3.4.** Pengurutan Nilai Euclidean Dari Nilai Terkecil

No.	Citra Latih	Euclidean	Kelas
1	2.jpg	7845,150759	Mtg
2	5.jpg	7849,598878	Mtg
3	74.jpg	7854,452742	Mth
4	73.jpg	7855,025962	Mth
5	4.jpg	7863,06729	Mtg
6	75.jpg	7961,642483	Mth
7	1.jpg	7964,911659	Mth
8	72.jpg	8172,103053	Mth
9	71.jpg	8355,458836	Mth
10	3.jpg	8453,851202	Mtg

6. Tentukan jarak terdekat sampai urutan K, terlihat pada tabel 3.5

**Tabel 3.5.** Jarak Terdekat dari K

No.	Citra Latih	Euclidean	Kelas	K
1	2.jpg	7845,150759	Mtg	1
2	5.jpg	7849,598878	Mtg	2
3	74.jpg	7854,452742	Mth	3
4	73.jpg	7855,025962	Mth	4
5	4.jpg	7863,06729	Mtg	5
6	75.jpg	7961,642483	Mth	6
7	1.jpg	7964,911659	Mth	7
8	72.jpg	8172,103053	Mth	8
9	71.jpg	8355,458836	Mth	9
10	3.jpg	8453,851202	Mtg	10

7. Hasil Klasifikasi dari Algoritm KNN

- a. Hasil dari pengklasifikasian pada data uji1 menggunakan 10 data latih teradap parameter k=3, terlihat pada tabel 3.6.

**Tabel 3.6** Hasil Klasifikasi dengan parmeter k=3

No.	Citra Latih	Euclidean	Kelas	
1	2.jpg	7845,150759	Mtg	1
2	5.jpg	7849,598878	Mtg	2
3	74.jpg	7854,452742	Mth	3
4	73.jpg	7855,025962	Mth	4
5	4.jpg	7863,06729	Mtg	5
6	75.jpg	7961,642483	Mth	6
7	1.jpg	7964,911659	Mth	7
8	72.jpg	8172,103053	Mth	8
9	71.jpg	8355,458836	Mth	9
10	3.jpg	8453,851202	Mtg	10

Pada tabel di atas terlihat bahwa tetanga terdekat menunjukkan pada kelas matang, sehingga untuk data uji 1 menggunakan data 10 data latih dengan parameter  $k=3$  adalah Tebu Matang.

- b. Hasil dari pengklasifikasian pada data uji1 menggunakan 10 data latih terhadap parameter  $k=5$ , terlihat pada tabel 3.7.

**Tabel 3.7** Hasil Klasifikasi dengan parameter  $k=5$

No.	Citra Latih	Euclidean	Kelas	K
1	2.jpg	7845,150759	Mtg	1
2	5.jpg	7849,598878	Mtg	2
3	74.jpg	7854,452742	Mth	3
4	73.jpg	7855,025962	Mth	4
5	4.jpg	7863,06729	Mtg	5
6	75.jpg	7961,642483	Mth	6
7	1.jpg	7964,911659	Mth	7
8	72.jpg	8172,103053	Mth	8
9	71.jpg	8355,458836	Mth	9
10	3.jpg	8453,851202	Mtg	10

- c. Hasil dari pengklasifikasian pada data uji1 menggunakan 10 data latih terhadap parameter  $k=7$ , terlihat pada tabel 3.8.

**Tabel 3.8** Hasil Klasifikasi dengan parameter  $k=7$

No.	Citra Latih	Euclidean	Kelas	K
1	2.jpg	7845,150759	Mtg	1
2	5.jpg	7849,598878	Mtg	2
3	74.jpg	7854,452742	Mth	3
4	73.jpg	7855,025962	Mth	4
5	4.jpg	7863,06729	Mtg	5
6	75.jpg	7961,642483	Mth	6
7	1.jpg	7964,911659	Mth	7
8	72.jpg	8172,103053	Mth	8
9	71.jpg	8355,458836	Mth	9
10	3.jpg	8453,851202	Mtg	10

Pada tabel di atas terlihat bahwa tetanga terdekat menunjukkan pada kelas mentah, sehingga untuk data uji 1 menggunakan data 10 data latih dengan parameter  $k=7$  adalah Tebu Mentah.

- d. Hasil dari pengklasifikasian pada data uji 1 menggunakan 10 data latih teradap parameter  $k=9$ , terlihat pada tabel 3.9.

**Tabel 3.9** Hasil Klasifikasi dengan parmeter  $k=9$

No.	Citra Latih	Euclidean	Kelas	K
1	2.jpg	7845,150759	Mtg	1
2	5.jpg	7849,598878	Mtg	2
3	74.jpg	7854,452742	Mth	3
4	73.jpg	7855,025962	Mth	4
5	4.jpg	7863,06729	Mtg	5
6	75.jpg	7961,642483	Mth	6
7	1.jpg	7964,911659	Mth	7
8	72.jpg	8172,103053	Mth	8
9	71.jpg	8355,458836	Mth	9
10	3.jpg	8453,851202	Mtg	10

Pada tabel di atas terlihat bahwa tetanga terdekat menunjukkan pada kelas mentah, sehingga untuk data uji 1 menggunakan data 10 data latih dengan parameter  $k=9$  adalah Tebu Mentah.

8. Lakukan langkah ke 4 sampa ke 7 untuk data uji ke 2, untuk hasil klasifikasi KNN dapat dilihat pada tabel 3.10

**Tabel 3.10** hasil klasifikasi KNN pada data uji 2

No.	Citra Latih	Euclidean	Kelas	K
1	3.jpg	4632,382	Mtg	1
2	5.jpg	4715,025	Mtg	2
3	2.jpg	4719,49	Mtg	3
4	1.jpg	4908,881	Mtg	4
5	4.jpg	5161,439	Mtg	5
6	73.jpg	5648,535	Mth	6
7	74.jpg	5682,451	Mth	7
8	75.jpg	6275,591	Mth	8
9	72.jpg	7410	Mth	9
10	71.jpg	7871,405	Mth	10

Pada tabel di atas menunjukkan hasil dari klasifikasi KNN untuk data uji 2 menggunakan data 10 data latih dengan parameter  $k=3$  adalah tebu matang,  $k=5$  adalah tebu matang,  $k=7$  adalah tebu matang,  $k=9$  adalah tebu matang.

### 3.5. Desain Antar Muka

Perencanaan sistem merupakan desain antarmuka untuk menampilkan citra yang akan diproses dalam system yang akan dibuat. Desain antarmuka tersebut dapat dilihat dalam tahapan-tahapan sebagai berikut:

#### 1. Menu Utama

Pada menu utama dalam tampilan ini terdapat beberapa tombol yang berfungsi untuk memproses objek secara jelas, dalam menu utama terdapat beberapa menu antara lain:

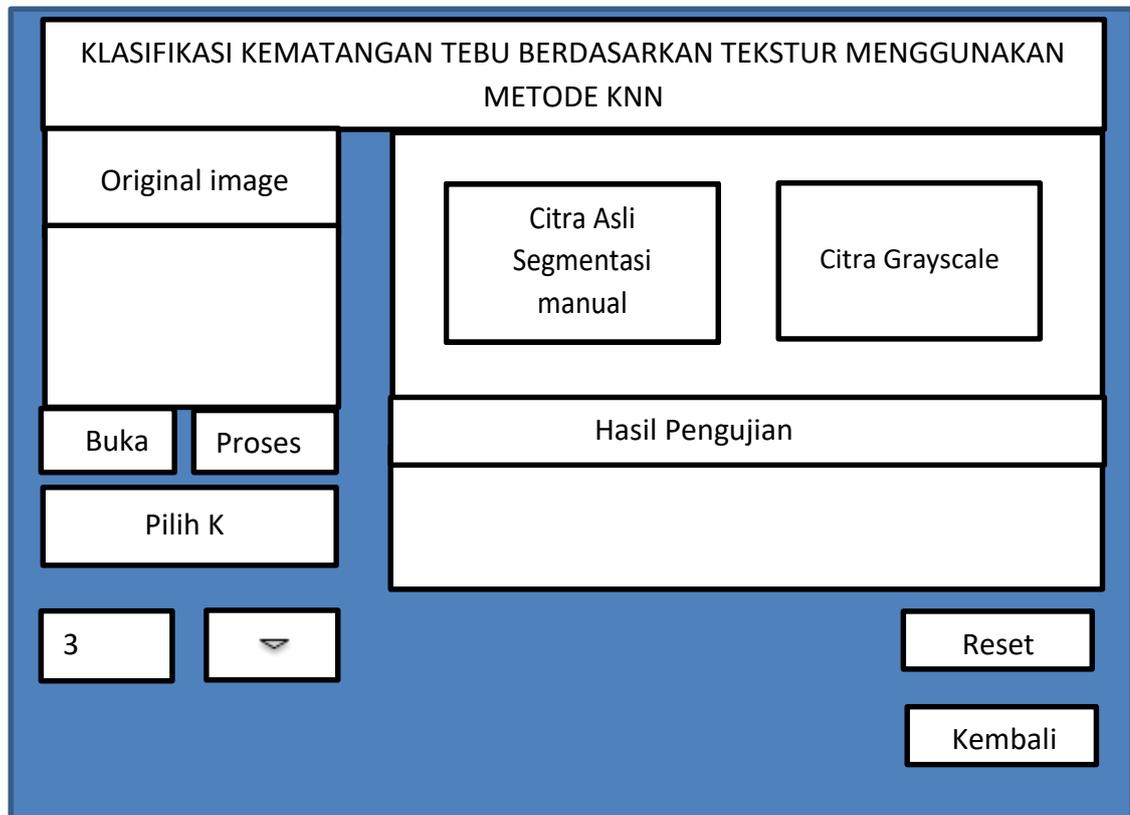
- a. Proses Pengujian
- b. Keluar



**Gambar 3.9** Antar Muka Utama

- a. Proses Pengujian : berfungsi untuk mengidentifikasi objek. Dalam proses ini, terdapat banyak proses. Berikut proses yang ada dalam proses pengujian :
  - Buka : untuk memilih objek mana yang akan diidentifikasi
  - Proses : Digunakan untuk memproses data

- Kembali : Untuk mengembalikan aplikasi ke tampilan menu utama
- Pilih K : Untuk memilih nilai K tetangga terdekat. K yang digunakan : 3, 5, 7, dan 9
- Original Image : Gambar citra awal yang sudah di segmentasi manual.



**Gambar 3.10** Desain Antar Muka Proses pengujian

### 3.6. Skenario Pengujian

Citra yang digunakan dalam skripsi ini berjumlah 400 citra yaitu :

1. Terdapat 300 citra latih, terbagi dalam 150 citra latih tebu matang, 150 citra latih tebu mentah.
2. Terdapat 100 citra Uji, terbagi dalam 50 citra latih tebu matang, 50 citra latih tebu mentah.
3. Lakukan Pre-Processing yaitu grayscaling, setelah itu ekstrasi ciri tekstur tebu dengan melakukan perhitungan mencari nilai ASM, CON, COR, VAR, IDM, ENT.

4. Dari nilai ekstarti ciri tersebut kemudian dilakukan klasifikasi dengan menggunakan K-NN (K-Nearest Neighbor). K yang digunakan adalah  $k=3$ ,  $k=5$ ,  $k=7$ ,  $k=9$ .
5. Langkah selanjutnya yaitu mencari nilai akurasi dari proses klasifikasi. Nilai akurasi dari klasifikasi didapatkan dengan membandingkan jumlah kelas yang benar dibagi dengan jumlah seluruh data dan dikalikan 100. Berikut adalah rumus nilai akurasi :

$$\text{a). Nilai akurasi} = \frac{\text{Jumlah kelas yang benar}}{\text{Jumlah seluruh data}} \times 100 \%$$

- b). Kemudian menampilkan hasil percobaan tersebut pada tabel 3.11 confusion sebagai berikut :

**Tabel 3.11** Matriks Hasil Prediksi

		Hasil Prediksi	
		MTG	MTH
Kelas Asli	MTG	A	B
	MTH	C	D

Keterangan :

MTG = Tebu Matang

MTH = Tebu Mentah

A = Tebu matang terbaca matang

B = Tebu matang terbaca mentah

C = Tebu mentah terbaca matang

D = Tebu mentah terbaca mentah