

Format Buku Kerja Praktik Mahasiswa (BKPM)

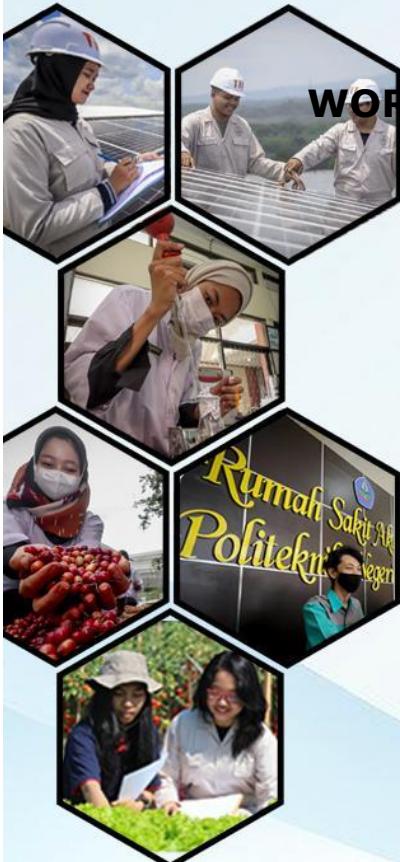


BUKU KERJA PRAKTIK MAHASISWA (BKPM)

**WORKSHOP PENGOLAHAN CITRA DAN VISION
TIF150705
SEMESTER V**

OLEH :

Zilvanhisna Emka Fitri, ST., MT



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI JEMBER
TAHUN 2022**

KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET DAN TEKNOLOGI
POLITEKNIK NEGERI JEMBER

LEMBAR PENGESAHAN
BUKU KERJA PRAKTIKUM MAHASISWA
WORKSHOP PENGOLAHAN CITRA DAN VISION

Mengetahui,

Koord. Program Studi,

Koord./Tim Mata Kuliah,

Penulis,

Trismayanti Dwi P, S.Kom., M.Cs
NIP. 199002272018032001

Zilvanhisna Emka F., ST., MT
NIP.19920302018032001

Zilvanhisna Emka F., ST., MT
NIP. 19920302018032001

Menyetujui,
Ketua Jurusan Teknologi Informasi,

Hendra Yufit Riskiawan, S.Kom., M.Cs
NIP. 19830203 2006041003

KATA PENGANTAR

Puji Syukur kehadirat Tuhan Yang Maha Esa karena atas limpahan rahmat-Nya sehingga kami dapat menyelesaikan Buku Kerja Praktikum Mahasiswa (BKPM) Pengolahan Citra dan Vision dalam pemenuhan Capaian Pembelajaran Mata Kuliah (CPMK) sesuai dengan Rancangan Pembelajaran Semester (RPS). Materi Workshop terdiri dari teknik-teknik dasar pengolahan citra digital seperti dasar pemrograman matlab, konversi citra warna, operasi dasar citra, *image enhancement*, konvolusi, segmentasi citra, operasi morfologi, ekstraksi fitur, filtering, deteksi tepi. Serta beberapa contoh implementasi computer vision dalam berbagai kasus, seperti pada bidang pertanian untuk deteksi sortasi mutu atau varietas tanaman, deteksi kematangan dan penyakit/hama. Pada bidang lain seperti pengenalan motif batik, deteksi keaslian uang kertas dan hand tracking. BKPM ini disusun berdasarkan metode *Student Center Learning* yaitu menempatkan mahasiswa sebagai pusat kegiatan belajar. Penulis menyadari masih banyak kekurangan dalam penyusunan BKPM ini. Oleh karena itu, kami sangat mengharapkan kritik dan saran demi perbaikan dan kesempurnaan modul ini. Kami mengucapkan terima kasih kepada berbagai pihak yang telah membantu proses penyelesaian BKPM ini. Semoga BKPM ini dapat bermanfaat bagi kita semua.

Jember, 12 Agustus 2022

Penulis,



Zilvanhisna Emka Fitri, ST., MT

NIP.1992030 201803 2 001

DAFTAR ISI

LEMBAR PENGESAHAN	2
KATA PENGANTAR	3
Acara 1	6
Acara 2	8
Acara 3	11
Acara 4	14
Acara 5	19
Acara 6	22
Acara 7	26
Acara 8	30
Acara 9	32
Acara 10	35
Acara 11	38
Acara 12	41
Acara 13	43
Acara 14	47
Acara 15	50
Acara 16	58
Acara 17	64
Acara 18	70
Acara 19	77
Acara 20	80
Acara 21	84
Acara 22	87
Acara 23	90
Acara 24	94
Acara 25	103
Acara 26	105
Acara 27	108
Acara 28	111
Acara 29	115
Acara 30	122
Acara 31	124
Acara 32	126
Acara 33-34.....	128

Acara 35-36.....	131
Acara 37	137
Acara 38	139
Acara 39	141
Acara 40	144
Acara 41	146
Acara 42	148
Acara 43	150
Acara 44	153
Acara 45-48.....	155
Acara 49-52.....	178
Acara 53-54.....	182
Acara 55-56.....	187
Acara 57-60.....	195

Acara 1

Materi Pembelajaran	: <i>Pengantar Pengolahan Citra dan Vision</i>
Acara Praktikum/Praktik	: Minggu 1/1
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK) :

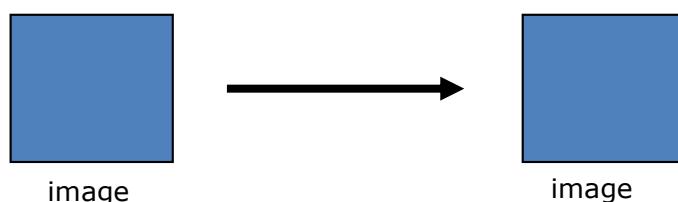
1. Mahasiswa dapat menjelaskan tentang Pengolahan Citra dan Vision
2. Mahasiswa mampu memahami contoh implementasi Citra dan Vision

b. Indikator Penilaian :

Kemampuan mahasiswa dalam memahami Citra dan Vision

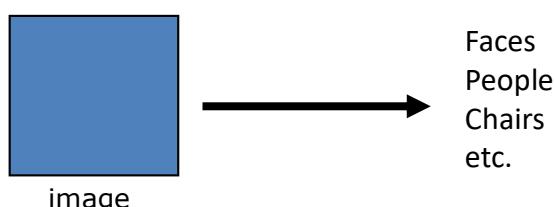
c. Dasar Teori :

Pengolahan Citra merupakan bidang ilmu yang mempelajari tentang bagaimana suatu citra itu dibentuk, diolah, dan dianalisis sehingga menghasilkan informasi yang dapat dipahami oleh manusia. Bidang ilmu tersebut terbagi menjadi dua subkeilmuan yaitu Pengolahan Citra Digital (PCD) dan Visi Komputer. Proses melalui transformasi dari satu citra kepada citra yang lain dengan melakukan perbaikan maupun transformasi citra berdasarkan algorithma tertentu disebut dengan Pengolahan Citra Digital. Beberapa contoh tahapan Pengolahan Citra Digital meliputi penajaman citra, penonjolan fitur tertentu dari citra, kompresi dan koreksi citra yang kabur atau tidak fokus. Sehingga dapat disimpulkan, input pada PCD adalah image/citra dan outputnya pun berupa citra yang sudah diolah menggunakan operasi-operasi tertentu.



Gambar 1. Ilustrasi Proses Pengolahan Citra Digital

Sedangkan proses pengolahan citra ini bertujuan untuk menghasilkan informasi berupa data maupun obyek yang lebih spesifik dan lebih bermanfaat bagi penggunanya. Pada visi computer, output yang dihasilkan tidak terbatas citra saja, namun dapat pula berupa simbolik maupun geometric.



Gambar 2. Ilustrasi Proses Visi Komputer

Contoh aplikasi pada bidang Pengolahan Citra Digital



Gambar 3. Proses Penajaman Citra

d. Alat dan Bahan :

1. Kertas A4
2. Pulpen
3. Laptop
4. Matlab 2020a atau Python

e. Prosedur Kerja ;

Mahasiswa menyiapkan instalasi program matlab atau python terlebih dahulu dan pastikan laptop anda sudah compatible dengan aplikasi yang telah terinstall.

f. Hasil dan Pembahasan: -

g. Rubrik Penilaian :

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 2

Pokok Bahasan	:Pengantar Pengolahan Citra dan Vision
Acara Praktikum/Praktik	:Minggu 1/2
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	:100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

- a. Mahasiswa mampu memahami pengolahan citra dan vision menggunakan pemrograman Aplikasi Matlab.
- b. Mahasiswa mampu menjalankan dan memahami *tools* yang terdapat pada pemrograman aplikasi Matlab

b. Indikator Penilaian

Kemampuan mahasiswa dalam menjalankan aplikasi maupun *tools* pada pemrograman aplikasi Matlab

c. Dasar Teori

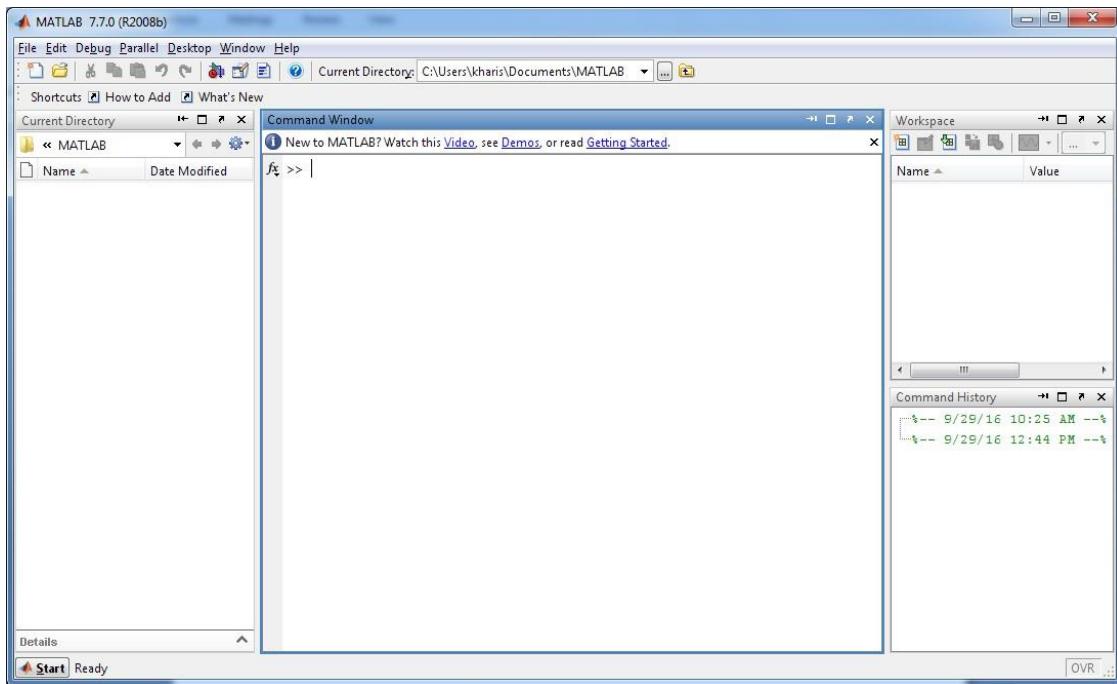
MATLAB (*Matrik Labority*) merupakan perangkat lunak produksi dari The Math Work, Inc. Matlab mampu mengintegrasikan komputasi, visualisasi, dan pemrograman dalam suatu model yang sangat mudah untuk pakai dimana masalah-masalah dan penyelesaiannya diekspresikan dalam notasi matematika yang familiar. Penggunaan Matlab meliputi bidang-bidang:

1. Matematika dan komputasi
2. Pembentukan Algorithm
3. Pemodelan, simulasi, dan pembuatan prototype
4. Analisa data, explorasi, dan visualisasi
5. Grafik keilmuan dan bidang rekayasa
6. Pengolahan Citra Digital

Matlab digunakan sebagai platform untuk menganalisis data dan visualisasi yang dirancang untuk membuat manipulasi matriks sesedehana mungkin. Disamping itu, Matlab memiliki kemampuan grafis yang kuat dan dengan bahasa pemrograman sendiri. Matlab memiliki banyak fitur-fitur yang sudah dikembangkan, dan lebih dikenal dengan nama *toolbox*. **Toolbox-toolbox** ini merupakan kumpulan dari fungsi-fungsi MATLAB (M-Files) yang telah dikembangkan ke suatu lingkungan kerja Matlab untuk memecahkan masalah dalam kelas particular. Untuk pengolahan citra toolbox yang digunakan adalah *Image-processing Toolbox* (IPT).

Struktur data dasar MATLAB adalah matriks. Pada MATLAB sebuah variable tunggal adalah matriks 1×1 , string adalah sebuah $1 \times n$ matriks dari karakter. Gambar adalah $n \times m$ matriks dari pixel. Pixel adalah elemen citra, elemen terkecil citra digital yang dapat dilihat mata. Ketika MATLAB dijalankan pertama kali, Matlab

desktop tampil, berisi *tools* (*graphical user interface*) untuk mengatur *file*, *variable*, dan aplikasi Matlab. Pada pertama kali Matlab dijalankan maka akan tampil *desktop* dengan ilustrasi sebagai berikut.



Gambar 4. Tampilan Awal Aplikasi Matlab

MATLAB IDE memiliki lima komponen, yaitu *Command Window*, *Workspace Browser*, *Current Directory Browser*, *Command History Window* dan kosong atau banyak *Figure Windows* yang aktif untuk menampilkan obyek grafik.

1. *Workspace* berisi kumpulan variable-variable yang terbentuk sepanjang sesi Matlab dan disimpan di memory.
2. *Current directory* browser adalah operasi Matlab *file* menggunakan *current directory* dan *search path* sebagai referensi. *File* yang akan dijalankan harus berada di *current directory* atau ada pada *serach path*.
3. *Command history* digunakan untuk menyimpan baris-baris perintah yang telah diketikkan di *command window*. Kita dapat melihat ungsi-fungsi yang digunakan sebelumnya, mengcopy, dan menjalankan kembali dari *command history*.
4. *Figure* adalah jendela untuk menampilkan gambar grafik seperti perintah *plot* dan sejenisnya. Gambar pada jendela *figure* ini dapat disimpan ke dalam sebuah file yang sewaktu-waktu dapat dipanggil kembali.

Command Windows merupakan window yang muncul ketika kita membuka pertama kali setiap kita menjalankan aplikasi MATLAB. *Command Window* (layar perintah) dapat kita gunakan untuk menjalankan program/perintah yang dibuat pada layar editor matlab serta untuk menampilkan hasil. Pada windows/layar ini kita dapat mengakses perintah maupun komponen pendukung (*help file* dll) yang ada di matlab

secara langsung. Salah satu ciri dari *command windows* ditandai dengan tanda prompt (>>).

d. Alat dan Bahan

1. Kertas A4
2. Pulpen
3. Laptop
4. Matlab 2020a

e. Prosedur Kerja

-

f. Hasil dan Pembahasan

1. Dokumentasi Tugas berupa laporan *screenshot* tahapan hasil praktik dalam membuat dan menyimpan project.
2. Dokumentasi Tugas berupa file word/pdf

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 3

Pokok Bahasan	:Pengantar Pengolahan Citra dan Vision
Acara Praktikum/Praktik	:Minggu 1/3
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	:100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa dapat membuat project baru pada pemrograman Matlab
2. Mahasiswa mampu menyimpan project Matlab
3. Mahasiswa mampu menampilkan citra menggunakan Matlab

b. Indikator Penilaian

Kemampuan mahasiswa dalam membuat dan menyimpan project serta menampilkan citra menggunakan Matlab

c. Dasar Teori

Membuat Project

Terdapat tiga cara yang dapat dilakukan untuk membuat project baru pada Aplikasi Matlab:

1. Dengan menggunakan shortcut **Ctrl+N**
2. Dengan klik symbol **+** pada Editor
3. Pilih **New – Script** pada menu toolbar

Menyimpan Project Matlab

Cara menyimpan project pada Matlab cukup sederhana dan relative sama dengan aplikasi pemrograman lain. Cara pertama, anda bisa menggunakan shortcut Ctrl+S dan memilih menu Save pada Toolbar.

Image Loading and Displaying

1. Imread

Gambar dimuat ke dalam memori kerja menggunakan perintah:

```
>> f = imread('rice.jpg');
```

Tanda titik koma dalam Matlab diakhir perintah digunakan untuk menghilangkan output (tidak ditampilkan). Jika tidak disertakan, maka Matlab akan menampilkan output ke layar. Penyebutan nama file tanpa path akan mengasumsikan bahwa file tersebut berada di *current working directory* (directory kerja). Selain itu pembacaan file juga dapat dilakukan dengan path relative. Perhatikan contoh berikut:

```
>> f = imread('F:\citra\ rice.jpg');
```

2. Imshow

Untuk menampilkan gambar dapat digunakan perintah berikut:

```
>> imshow(F)
```

Dimana F adalah variable yang akan ditampilkan tetapi dengan syarat F harus sudah di open terlebih dahulu.

3. Figure

Gambaran pada umumnya ditampilkan didalam *figure window*. Fungsi figure menghasilkan suatu jendela figure baru dan memajang gambar didalamnya:

```
>> figure, imshow(f)
```

4. Imwrite

Gambar dapat disimpan ke disk menggunakan fungsi imwrite, yang memiliki fungsi dasar:

```
>> imwrite(array name, 'file name')
```

String yang menjadi isi parameter nama file harus memasukkan estensi format file yang dikenali. Alternatifnya, format yang diinginkan dapat disebutkan secara eksplisit pada masukkan parameter ketiga. Tampat penyimpanan fungsi ini (lokasi file output) ada di direktori kerja. Contoh dibawah ini adalah untuk menyimpan gambar i ke TIFF file dengan nama rice_01:

```
>> imwrite(i , 'rice_01.tif');
```

5. Image Information

Informasi tentang sebuah gambar dapat ditemukan dengan :

```
>> iminfo filename
```

6. Clc

clc digunakan untuk membersihkan semua yang di input dan di outputkan dari Command Window display

```
>> clc
```

d. Alat dan Bahan

1. Kertas A4
2. Pulpen
3. Laptop
4. Matlab 2020a

e. Prosedur Kerja

1. Buatlah laporan berupa screenshot tahapan dalam membuat project baru, menyimpan project dan menampilkan gambar pada Matlab.
2. Dokumentasi Tugas berupa word/pdf.

f. Hasil dan Pembahasan

-

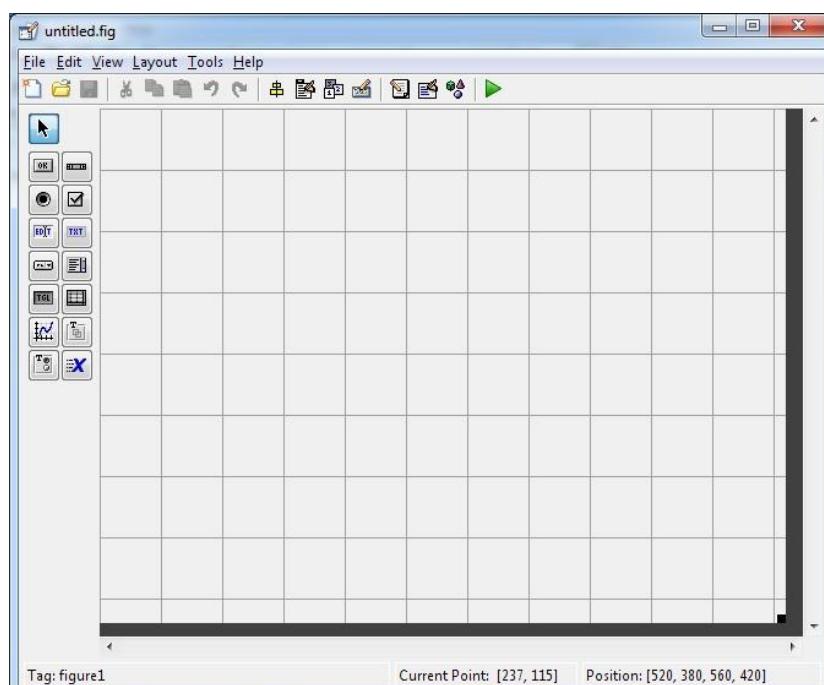
g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 4

Pokok Bahasan	:Pengantar Pengolahan Citra dan Vision
Acara Praktikum/Praktik	:Minggu 1/3
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	:100 menit

- a. Capaian Pembelajaran Mata Kuliah (CPMK)
 1. Mahasiswa dapat memahami fungsi GUI pada Matlab
 2. Mahasiswa dapat memahami kegunaan *tools* pada GUI pada Matlab
 3. Mahasiswa mampu menampilkan citra dengan mengimplementasikan GUI pada Matlab
- b. Indikator Penilaian
Kemampuan mahasiswa dalam menampilkan citra menggunakan fitur GUI pada Matlab
- c. Dasar Teori
GUI (Graphical User Interface) atau biasa disebut GUIDE (GUI builder) Matlab merupakan aplikasi display dari MATLAB yang mengandung tugas, perintah, atau komponen program yang mempermudah user (pengguna) dalam menjalankan sebuah program dalam MATLAB. Untuk memulai penggunaan GUI Matlab dapat dilakukan dengan dua cara yaitu:
 1. Melalui command window dengan mengetik >> guide
 2. Arahkan kursor pada menu Home, pilih New – Graphical User Interface



Gambar 5. Tampilan Awal GUI

Tampilan GUI terdapat berbagai komponen yang terdiri dari beberapa uicontrol (kontrol user interface), seperti pada bahasa pemrograman visual lainnya, yaitu: *pushbutton*, *togglebutton*, *radiobutton*, *checkboxes*, *edit text*, *static text*, *slider*, *frames*, *listboxes*, *popup menu*, dan *axes*. Kita dapat meletakkan semua kontrol pada layout editor dan selanjutnya hanya tinggal mengaturnya melalui property inspector. Semua kontrol pada GUI dapat dimunculkan pada *layout/figure* dengan cara mendrag and drop kontrol yang diinginkan ke *figure*. Adapun penjelasan fungsi masing-masing kontrol pada tabel berikut:

Tabel 1. UI Control pada GUI Matlab

UIControl	Nama UIControl	Keterangan
	Select Tool	Untuk menyeleksi tools
	Push Button	Untuk membuat Button pada figure
	Radio Button	Untuk membuat button pilihan
	Edit Text	Memasukkan atau memodifikasi suatu text
	Pop-up Menu	Menampilkan daftar pilihan pada string property
	Toogle Button	Membuat button pada figure
	Axes	Menampilkan sebuah grafik untuk gambar
	Button Group	Membuat button pada figure
	Slider	Membuat slider secara vertical dan horizontal
	Check Box	Membuat button pilihan mandiri
	Statistic Text	Untuk membuat atau menampilkan text
	List Box	Membuat list pada figure
	Tables	Untuk membuat table pada figure

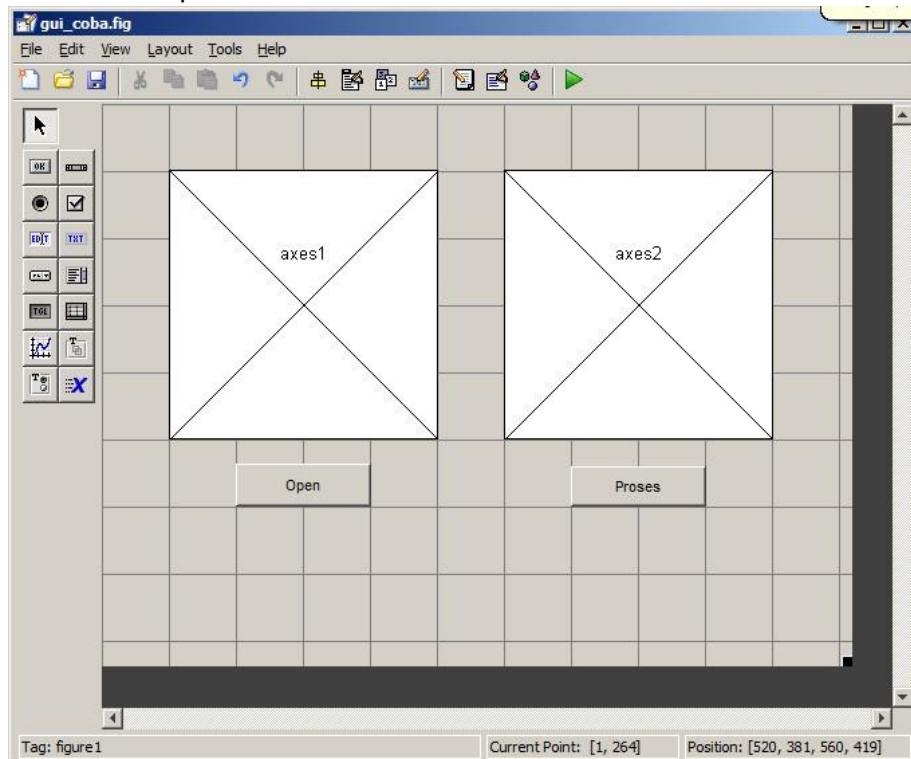
	Panel	Membuat panel pada figure
	ActiveX Control	Mengaktifkan control lain

d. Alat dan Bahan

1. Kertas A4
2. Pulpen
3. Laptop
4. Matlab 2020a

e. Prosedur Kerja

1. Setelah membuat GUI baru, klik Uicontrol axes, buat dua Uicontrol axes pada *figure* GUI yang sudah tersedia. Kemudian buat 2 tombol *push button* pada *figure*, kemudian letakkan setiap *push button* di bawah *axes1* dan *axes2*. Beri nama 2 *push button* dengan nama **Open** untuk *button* di bawah *axes1* dan **Proses** untuk *push button* di bawah *axes2* dengan klik dua kali pada *push button* tersebut, kemudian pilih *String* untuk merubah nama pada *push button* seperti Gambar 6.



Gambar 6. Tampilan Dua Axes pada Figure

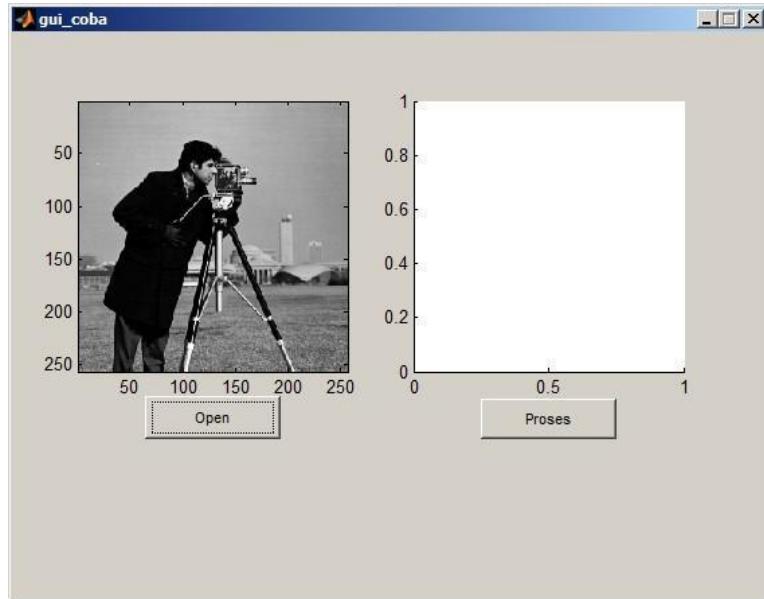
2. Kemudian klik kanan pada button Citra awal, kemudian pilih *view callback* > *callback*. Kemudian akan masuk pada script GUI dan masukkan script dibawah ini pada function *pushbutton1*:

```

function pushbutton1_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
open=guidata(gcbo); [namafайл,direktori]=uigetfile('*.jpg;*.bmp;*.tif', 'OpenImage');
I=imread(namafайл); set(open.figure1,'CurrentAxes',open.axes1); set(imagesc(I)); colormap('gray');
set(open.axes1,'Userdata',I);

```

Maka akan menghasilkan tampilan seperti terlihat pada Gambar 7 dibawah ini:



Gambar 7. Fungsi *button open*

3. Lalu klik kanan pada push button Proses dan tulislah *script* pada *button* tersebut dengan seperti dibawah ini:

```

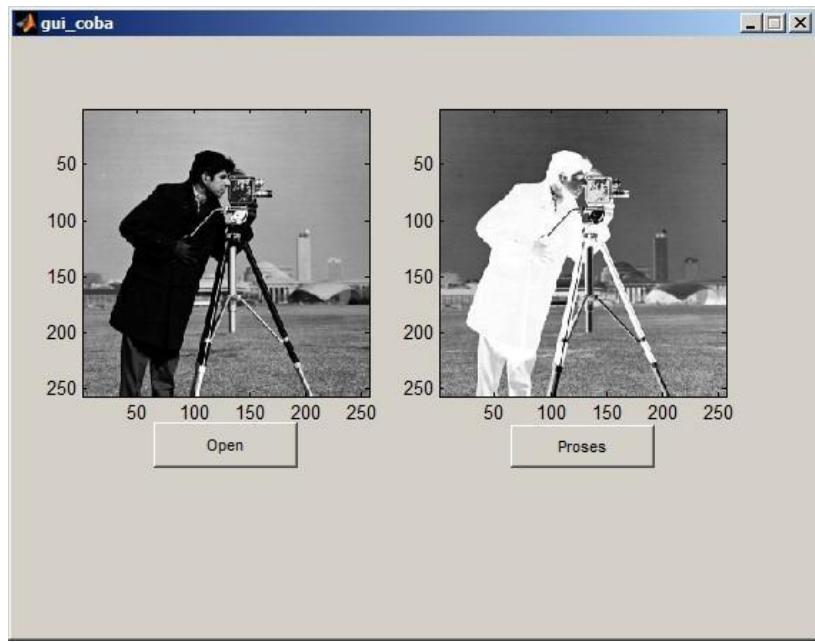
function pushbutton2_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

open=guidata(gcbo); [namafайл,direktori]=uigetfile('*.jpg;*.bmp;*.tif', 'OpenImage');
I=imread(namafайл); set(open.figure1,'CurrentAxes',open.axes1); set(imagesc(I)); colormap('gray');
set(open.axes1,'Userdata',I);

open=guidata(gcbo); I=get(open.axes1,'Userdata');
[r c] = size(I);
for x = 1 : r
    for y = 1 : c
        J(x,y) = 255 - I(x,y);
    end
end
set(open.figure1,'CurrentAxes',open.axes2); set(imagesc(J)); colormap('gray');
set(open.axes2,'Userdata',J);

```

Maka akan tampil seperti pada Gambar 8 dibawah ini:



Gambar 8. Tampilan Button Proses

f. Hasil dan Pembahasan

-

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 5

Pokok Bahasan	:Operasi Dasar Pengolahan Citra
Acara Praktikum/Praktik	:Minggu 2/1
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	:100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa dapat memahami representasi citra dalam sebuah matriks
2. Mahasiswa dapat memahami citra grayscale
3. Mahasiswa mampu melakukan proses konversi citra RGB menjadi grayscale

b. Indikator Penilaian

Kemampuan mahasiswa dalam memahami representasi citra dan mengkonversi citra RGB menjadi grayscale

c. Dasar Teori

Sebuah citra dalam prosesnya direpresentasikan sebagai sebuah matriks dengan karakteristik tertentu, antara lain:

- Satuan atau bagian terkecil dari suatu elemen disebut **Piksel** (*pixel* atau *Picture Element*) yang berarti element citra
- Citra dibentuk dari kotak-kotak persegi empat yang teratur sehingga jarak horizontal dan vertikal antar piksel adalah sama pada seluruh bagian Citra.
- Dalam Komputer, setiap piksel diwakili oleh dua bilangan bulat (integer) untuk menunjukkan lokasinya dalam bidang citra.
- Sebuah nilai dalam bilangan bulat untuk menunjukkan cahaya atau terang gelap piksel tersebut
- Untuk menunjukkan lokasi piksel koordinat (0,0) digunakan untuk posisi kiri atas dalam bidang citra
- Koordinat (m-1,n-1) untuk posisi kanan bawah dalam citra berukuran m x n piksel.
- Tingkat Pencahayaan piksel bilangan bulat 8 bit dengan lebar selang selang nilai 0-255 (0 = warna hitam) dan 255 untuk warna putih dan tingkat abu-abu antara nilai 0 dan 255

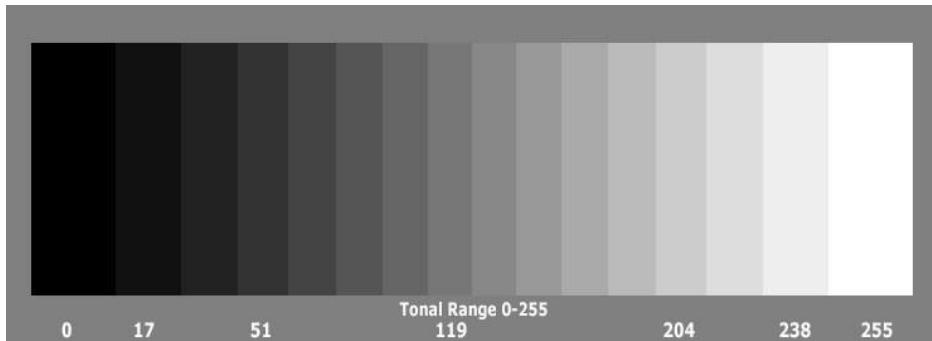
- Berikut merupakan representasi matriks pada citra grayscale:

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \cdots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}$$

183	160	94	153	194	163	132	165
183	153	116	176	187	166	130	169
179	168	171	182	179	170	131	167
177	177	179	177	179	165	131	167
178	178	179	176	182	164	130	171
179	180	180	179	183	169	132	169
179	179	180	182	183	170	129	173
180	179	181	179	181	170	130	169

Gambar 9. Representasi Matriks

Grayscale atau abu-abu pada sebuah image digital adalah image yang pada setiap pixelnya hanya berisikan informasi intensitas warna putih dan hitam (berada dalam range 0-255). Image Grayscale memiliki banyak variasi nuansa abu-abu sehingga berbeda dengan image hitam-putih. Grayscale juga disebut monokromatik karena tidak memiliki warna lain selain variasi intensitas putih dan hitam. Sebuah image yang dijadikan Grayscale akan terkesan berbeda bila dibandingkan dengan image berwarna.



Gambar 10. Representasi gradasi graylevel dalam range 0-255

- d. Alat dan Bahan
 - 5. Kertas A4
 - 6. Pulpen
 - 7. Laptop
 - 8. Matlab 2020a
- e. Prosedur Kerja
 1. Siapkan 1 Citra dan Matlab
 2. Kemudian buat script baru dan tulis kode berikut pada script tsb (gunakan citra yang berbeda)

```
gambar= imread('Lenna.png');
gbr_gray=rgb2gray(gambar);
subplot (1,2,1), imshow(gambar),title ('Citra Asli');
subplot (1,2,2), imshow(gbr_gray),title('Citra Grayscale');
```
- f. Hasil dan Pembahasan

Apabila kode diimplementasi dengan benar, maka akan menghasilkan citra seperti berikut:



1. Sertakan screenshot kode dan hasil pada laporan
2. Pada workspace, lakukan Analisa perubahan-perubahan yang terjadi pada matriks dari variable-variabel yang telah diinisialisasi

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 6

Pokok Bahasan	:Operasi Dasar Pengolahan Citra
Acara Praktikum/Praktik	:Minggu 2/2
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	:100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa dapat menjelaskan tentang Operasi Titik Citra: Negatif dan Clipping
2. Mahasiswa dapat menjelaskan tentang Operasi Dasar Citra: Pengurangan dan Penjumlahan
3. Mahasiswa dapat mengimplementasi Operasi Titik Citra dan Operasi Dasar Citra menggunakan Matlab

b. Indikator Penilaian

Kemampuan mahasiswa dalam Operasi Titik Citra dan Operasi Dasar Citra

c. Dasar Teori

1. Operasi Negatif

Operasi negatif bertujuan untuk mendapatkan citra negatif dengan cara mengurangi nilai intensitas piksel dari nilai keabuan maksimum. Secara umum persamaannya adalah sebagai berikut:

$$f(x,y)' = 255 - f(x,y)$$

2. Operasi Clipping

Operasi clipping adalah operasi pemotongan jika nilai intensitas piksel hasil suatu operasi pengolahan citra terletak di bawah nilai intensitas minimum atau di atas nilai intensitas maksimum.

$$f(x,y)' = \begin{cases} 255, & f(x,y) > 255 \\ f(x,y), & 0 \leq f(x,y) \leq 255 \\ 0, & f(x,y) < 0 \end{cases}$$

3. Operasi Penjumlahan

Penjumlahan citra adalah operasi menjumlahkan dua matriks yang berukuran sama. Secara umum, persamaannya adalah sebagai berikut:

$$\begin{aligned} C(x,y) &= A(x,y) + \\ &\quad B(x,y) \end{aligned}$$

C adalah citra baru yang intensitas tiap pikselnya adalah jumlah dari intensitas tiap piksel pada matriks A dan matriks B.

4. Operasi Pengurangan

Pengurangan citra adalah operasi saling mengurangkan dua matriks yang berukuran sama. Secara umum, persamaannya adalah sebagai berikut:

$$C(x,y) = A(x,y) - B(x,y)$$

C adalah citra baru yang intensitas tiap pikselnya adalah selisih dari intensitas tiap piksel pada matriks A dan matriks B.

d. Alat dan Bahan

1. Kertas A4
2. Pulpen
3. Laptop
4. Matlab 2020a

e. Prosedur Kerja

1. Tuliskan script berikut untuk mengimplementasi operasi negative (gunakan citra yang berbeda):

```
F = imread('cameraman.jpg');
F = rgb2gray(F);
[r c] = size(F);
for x = 1 : r
    for y = 1 : c
        G(x,y) = 255-F(x,y);
    end
end
subplot (1,2,1), imshow(F), title ('Citra Asli');
subplot (1,2,2), imshow(G), title('Citra Negatif');
```

2. Tuliskan script berikut untuk mengimplementasi operasi Clipping (gunakan citra yang berbeda):

```
F = imread('cameraman.jpg');
F = rgb2gray(F);
J = clipping(F);
subplot (1,2,1), imshow(F), title ('Citra Asli');
subplot (1,2,2), imshow(J), title('Citra Clipping');
```

3. Buatlah 1 function baru (New+Function) dengan nama clipping.m dan tuliskan script berikut:

```
function J = clipping(I)

for x = 1 : size(I,1)
    for y = 1 : size(I,2)
        if I(x,y) > 255
            J(x,y) = 255;
        elseif I(x,y) < 0
            J(x,y) = 0;
        else
            J(x,y) = I(x,y);

        end
    end
end
end
```

4. Tuliskan script berikut untuk mengimplementasi operasi Penjumlahan (gunakan citra yang berbeda):

```

A = double(imread('cameraman.tif'));
B = double(imread('rice.png'));
[r1 c1] = size(A);
[r2 c2] = size(B);
if (r1 == r2) && (c1 == c2)

    for x = 1 : r1
        for y = 1 : c1
            C(x,y) = A(x,y) + B(x,y);
        end
    end
end
C = clipping(C);
C=uint8(C);
subplot (1,3,1), imshow(uint8(A)),title ('Citra 1');
subplot (1,3,2), imshow(uint8(B)),title ('Citra 2');
subplot (1,3,3), imshow(uint8(C)),title ('Citra Penjumlahan');

```

5. Tuliskan script berikut untuk mengimplementasi operasi Pengurangan
(gunakan citra yang berbeda):

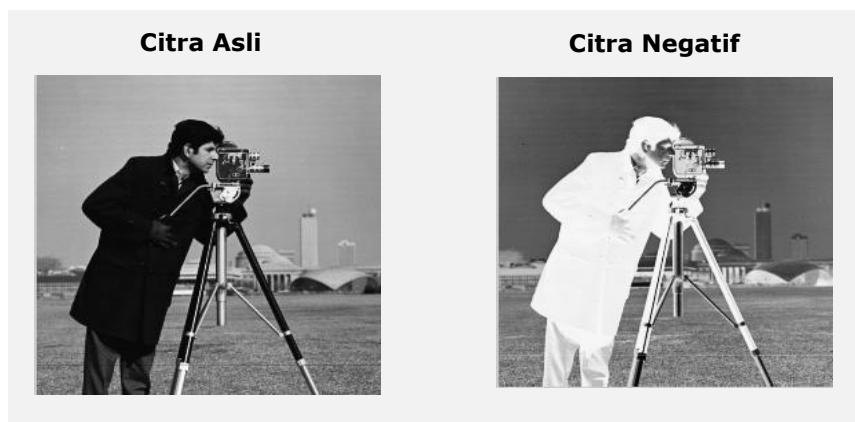
```

A = double(imread('cameraman.tif'));
B = double(imread('rice.png'));
[r1 c1] = size(A);
[r2 c2] = size(B);
if (r1 == r2) && (c1 == c2)
    for x = 1 : r1
        for y = 1 : c1
            C(x,y) = A(x,y) - B(x,y);
        end
    end
end
C = clipping(C);
subplot (1,3,1), imshow(uint8(A)),title ('Citra 1');
subplot (1,3,2), imshow(uint8(B)),title ('Citra 2');
subplot (1,3,3), imshow(uint8(C)),title ('Citra Pengurangan');

```

f. Hasil dan Pembahasan

- Hasil Pengolahan Citra Negatif



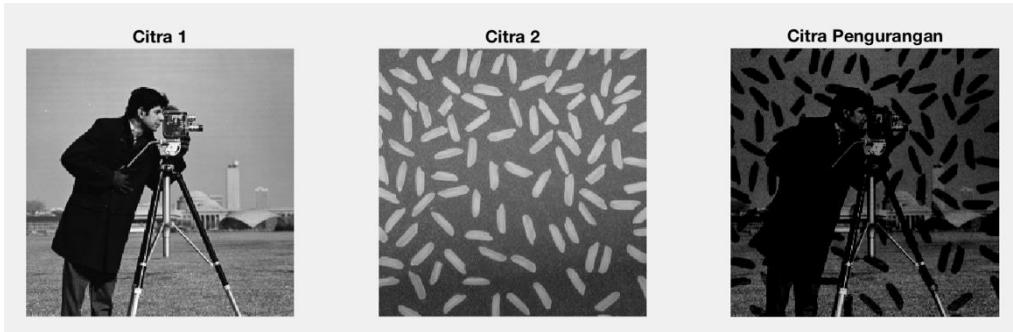
- Hasil Pengolahan Citra Clipping



3. Hasil Pengolahan Citra Penjumlahan



4. Hasil Pengolahan Citra Pengurangan



g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 7

Pokok Bahasan	:Operasi Dasar Pengolahan Citra
Acara Praktikum/Praktik	:Minggu 2/3
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	:100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa dapat menjelaskan tentang Operasi Dasar Citra: Perkalian, dan Pembagian
2. Mahasiswa dapat menjelaskan tentang Operasi Boolean Citra: AND, OR, dan XOR
3. Mahasiswa dapat mengimplementasi Operasi Dasar Citra dan Operasi Boolean menggunakan Matlab

b. Indikator Penilaian

Kemampuan mahasiswa dalam memahami dan menerapkan Operasi Dasar Citra dan Operasi Boolean menggunakan Matlab

c. Dasar Teori

1. Operasi Perkalian Citra

Perkalian citra A dengan skalar c akan menghasilkan citra baru B yang intensitasnya lebih terang dari semula. Kenaikan intensitas setiap piksel sebanding dengan c. Operasi perkalian citra dengan scalar digunakan untuk kalibrasi kecerahan.

$$B(x,y) = A(x,y) * c$$

2. Operasi Pembagian Citra

Pembagian citra A dengan scalar c akan menghasilkan citra baru B yang intensitasnya lebih gelap dari semula. Penurunan intensitas setiap piksel berbanding terbalik dengan c. Operasi pembagian citra dengan scalar digunakan untuk normalisasi kecerahan.

$$B(x,y) = A(x,y) / c$$

3. Operasi AND

Tabel 2. Tabel Kebenaran Logika AND

A	B	A and B
0	0	0
0	1	0
1	0	0
1	1	1

4. Operasi OR

Tabel 3. Tabel Kebenaran Logika OR

A	B	A or B
0	0	0
0	1	1
1	0	1
1	1	1

5. Operasi XOR

Tabel 4. Tabel Kebenaran Logika XOR

A	B	A xor B
0	0	0
0	1	1
1	0	1
1	1	0

d. Alat dan Bahan

1. Kertas A4
2. Pulpen
3. Laptop
4. Matlab 2020a

e. Prosedur Kerja

1. Tuliskan script berikut untuk mengimplementasi operasi perkalian (gunakan citra yang berbeda):

```
A = double(imread('cameraman.tif'));
[r c] = size(A);
for x = 1 : r
    for y = 1 : c
        B(x,y) = A(x,y) .* 2;
    end
end
B = clipping(B);
subplot (1,2,1), imshow(uint8(A)), title ('Citra 1');
subplot (1,2,2), imshow(uint8(B)), title ('Citra Perkalian');
```

2. Tuliskan script berikut untuk mengimplementasi operasi pembagian (gunakan citra yang berbeda):

```
A = double(imread('cameraman.tif'));
[r c] = size(A);
for x = 1 : r
    for y = 1 : c
        B(x,y) = A(x,y) ./ 2;
    end
end
B = clipping(B);
subplot (1,2,1), imshow(uint8(A)), title ('Citra 1');
subplot (1,2,2), imshow(uint8(B)), title ('Citra Pembagian');
```

3. Tuliskan script berikut untuk mengimplementasi operasi AND (gunakan citra yang berbeda):

```
A = im2bw(imread('cameraman.tif'));
B = im2bw(imread('rice.png'));
C = and(A,B);
subplot (1,3,1), imshow(A),title ('Citra 1');
subplot (1,3,2), imshow(B),title ('Citra 2');
subplot (1,3,3), imshow(C),title ('Citra And');
```

4. Tuliskan script berikut untuk mengimplementasi operasi OR (gunakan citra yang berbeda):

```
A = im2bw(imread('cameraman.tif'));
B = im2bw(imread('rice.png'));
C = or(A,B);
subplot (1,3,1), imshow(A),title ('Citra 1');
subplot (1,3,2), imshow(B),title ('Citra 2');
subplot (1,3,3), imshow(C),title ('Citra OR');
```

5. Tuliskan script berikut untuk mengimplementasi operasi XOR (gunakan citra yang berbeda):

```
A = im2bw(imread('cameraman.tif'));
B = im2bw(imread('rice.png'));
C = xor(A,B);
subplot (1,3,1), imshow(A),title ('Citra 1');
subplot (1,3,2), imshow(B),title ('Citra 2');
subplot (1,3,3), imshow(C),title ('Citra XOR');
```

f. Hasil dan Pembahasan

1. Hasil Pengolahan Citra Perkalian



2. Hasil Pengolahan Citra Pembagian



3. Hasil Pengolahan Citra AND



4. Hasil Pengolahan Citra OR



5. Hasil Pengolahan Citra XOR



g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 8

Pokok Bahasan	:Operasi Dasar Pengolahan Citra
Acara Praktikum/Praktik	:Minggu 2/4
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	:100 menit

- a. Capaian Pembelajaran Mata Kuliah (CPMK)
 - 1. Mahasiswa dapat menjelaskan tentang Operasi Dasar Geometri: Translasi, Cropping, dan Flipping
 - 2. Mahasiswa mampu mengimplementasi Operasi Dasar Geometri: Translasi, Cropping, dan Flipping menggunakan Matlab
- b. Indikator Penilaian
 - Kemampuan mahasiswa dalam memahami dan mengimplementasi Operasi Dasar Geometri
- c. Dasar Teori

1. Operasi Translasi

Rumus translasi citra:

$$x' = x + Tx$$

$$y' = y + Ty$$

Yang dalam hal ini, Tx adalah besar pergeseran dalam arah x, sedangkan Ty adalah besar pergeseran dalam arah y

2. Operasi Cropping

Rumus cropping citra:

$$w' = xR - xL$$

$$h' = yB - yT$$

Yang dalam hal ini, w' adalah lebar citra baru yang diperoleh setelah proses cropping. Sedangkan h' adalah tinggi citra baru. xR dan xL adalah dua titik disebelah kiri dan kanan pada arah sumbu x. yB dan yT adalah dua titik disebelah atas dan bawah pada arah sumbu y. Keempat titik xR , xL , yB , yT akan digunakan sebagai koordinat-koordinat dimana citra akan dipotong.

3. Operasi Flipping

Flipping adalah operasi geometri yang sama dengan pencerminan. Ada dua macam flipping: horizontal dan vertical. Flipping vertikal adalah pencerminan pada sumbu-X dari citra A menjadi citra B, yang diberikan oleh:

$$x' = 2xc - x$$

$$y' = y$$

- d. Alat dan Bahan

1. Kertas A4
 2. Pulpen
 3. Laptop
 4. Matlab 2020a
- e. Prosedur Kerja
1. Buatlah kelompok maksimal 5 orang, dan buatlah sebuah fungsi berdasarkan tahapan pada dasar teori menggunakan sebuah citra.
 2. Buatlah laporan dan presentasikan hasil operasi dasar pengolahan citra
- f. Hasil dan Pembahasan
-
- g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 9

Pokok Bahasan	:Image Enhancement
Acara Praktikum/Praktik	:Minggu 3/1
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	:100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami definisi dari pengolahan citra dan vision.
2. Mahasiswa mampu memahami proses Image Enhancement.
3. Mahasiswa mampu memberikan contoh penerapan teknik image enhancement

b. Indikator Penilaian

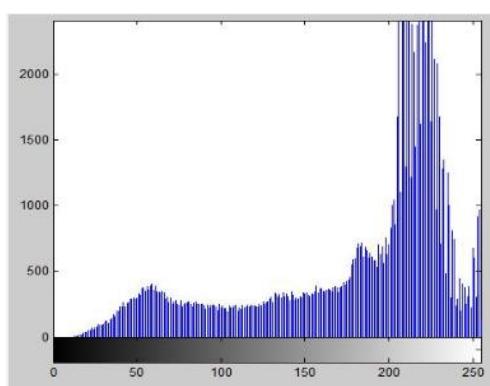
Ketepatan dalam menjelaskan mengenai teknik image enhancement

c. Dasar Teori

Peningkatan kualitas citra (Image Enhancement) bertujuan untuk memperbaiki citra yang mengalami derau (noise), kabur dan sebagainya untuk memperoleh citra berkualitas lebih baik sehingga mempermudah analisis citra. Contoh operasi perbaikan kualitas citra :

1. Perbaikan kecerahan (brightness)
2. Perbaikan kontras (contrast stretching)
3. Operasi Negasi (Invers)
4. Histogram Equalization

Sebelum membahas terkait perbaikan kualitas citra, kita harus memahami terlebih dahulu tentang histogram citra. Histogram citra adalah grafik yang menggambarkan penyebaran nilai-nilai intensitas piksel dari suatu citra atau bagian tertentu di dalam citra. Histogram citra digambarkan pada koordinat kartesian dimana sumbu x (absis) menunjukkan tingkat warna dan sumbu y (ordinat) menunjukkan frekuensi kemunculan (jumlah piksel) seperti yang ditunjukkan pada Gambar 9.



Gambar 11. Histogram Citra

d. Alat dan Bahan

1. Kertas A4
2. Pulpen
3. Laptop
4. Citra
5. Matlab 2020a

e. Prosedur Kerja

1. siapkan foto dan matlab

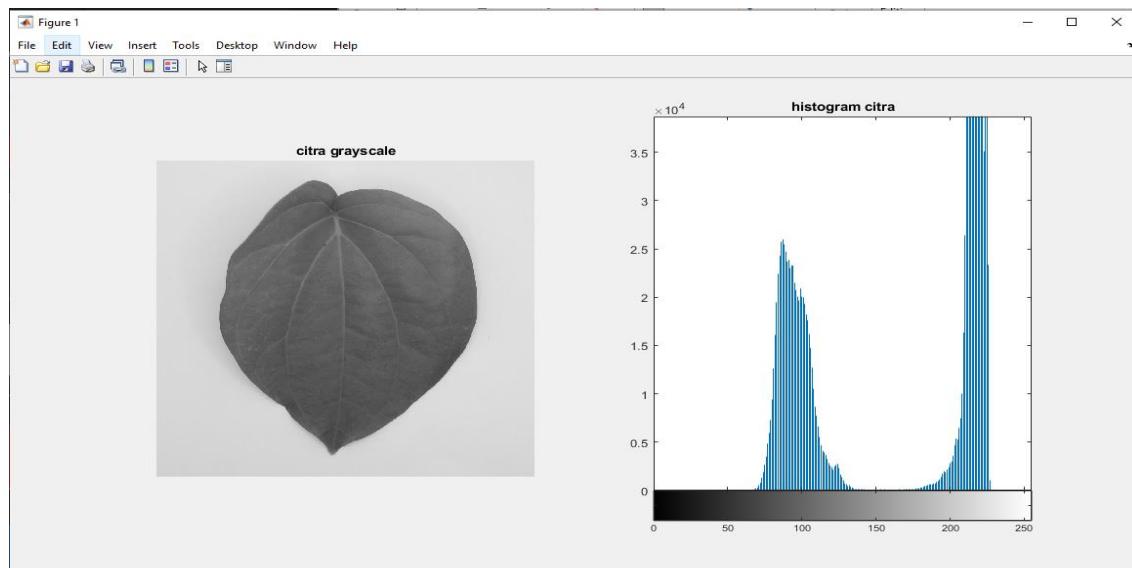


2. kemudian tulis sourcecode berikut :

```
% untuk membaca citra
gambar = imread('E:\1.Vani\coba\sirih.jpg'); %lokasi citra
sirih = rgb2gray(gambar); %konversi warna dari RGB ke grayscale

% menampilkan hasil konversi warna grayscale
subplot(1,2,1),imshow(sirih),title('citra grayscale');

%imhist digunakan untuk menampilkan histogram
```



f. Hasil dan Pembahasan

Dokumentasi Tugas berupa laporan (definisi, proses dan penerapan histogram citra). Dokumentasi Tugas berupa file ppt

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 10

Pokok Bahasan	:Image Enhancement
Acara Praktikum/Praktik	:Minggu 3/2
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	:100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami definisi dari pengolahan citra dan vision.
2. Mahasiswa mampu memahami proses Image Enhancement.
3. Mahasiswa mampu memberikan contoh penerapan teknik image enhancement.

b. Indikator Penilaian

Ketepatan dalam menjelaskan mengenai teknik image enhancement

c. Dasar Teori

Operasi negative (Negasi/invers) bertujuan untuk mendapatkan citra negative meniru film negative pada fotografi dengan cara mengurangi nilai intensitas piksel dari nilai keabuan maksimum. Contohnya pada citra dengan 256 derajat keabuan (8 bit) digunakan persamaan rumus :

$$f_0(x, y) = 255 - f_i(x, y)$$

Sedangkan pada citra dengan 128 derajat keabuan menggunakan persamaan rumus :

$$f_0(x, y) = 127 - f_i(x, y)$$

Hasil operasi negative ditunjukkan pada Gambar 10



Gambar 12. Hasil Operasi Negatif (Kanan)

d. Alat dan Bahan

1. Kertas A4
2. Pulpen

3. Laptop
 4. Citra
 5. Matlab 2020a
- e. Prosedur Kerja
1. siapkan foto dan matlab



2. kemudian tulis sourcecode berikut :

```
gambar = imread('E:\1.Vani\New folder (2)\Data Tomat New\B_96.jpg');
tomat = rgb2gray(gambar);
negasi = 255-tomat;

subplot(1,4,1),imshow(gambar),title('citra asli');
subplot(1,4,2),imshow(red),title('citra grayscale');
subplot(1,4,3),imshow(negasi),title('citra negasi');
```

3. ubahlah code 'tomat = rgb2gray(gambar);' menjadi :

- a. tomat = gambar (:,:,1);
- b. tomat = gambar (:,:,2);
- c. tomat = gambar (:,:,3);

4. kemudian lakukan proses operasi negasi dan Analisa hasilnya serta bandingkan dengan Langkah sebelumnya

5. Lakukan juga Analisa dari histogramnya

- f. Hasil dan Pembahasan
1. Lakukan Analisa hasil operasi negasi dan histogramnya
 2. Dokumentasi Tugas berupa laporan (definisi, proses dan penerapan histogram citra)
 3. Dokumentasi Tugas berupa file ppt

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 11

Pokok Bahasan	:Image Enhancement
Acara Praktikum/Praktik	:Minggu 3/3
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	:100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami definisi dari pengolahan citra dan vision.
2. Mahasiswa mampu memahami proses Image Enhancement.
3. Mahasiswa mampu memberikan contoh penerapan teknik image enhancement.

b. Indikator Penilaian

Ketepatan dalam menjelaskan mengenai teknik image enhancement

c. Dasar Teori

Umumnya citra grayscale mempunyai 256 warna, citra tersebut akan tampak gelap bila seluruh komponen warnanya mendekati nilai 0 namun citra tersebut akan tampak terang bila komponen warnanya mendekati nilai 255. Persamaan rumus yang digunakan pada operasi kecerahan atau *brightness* :

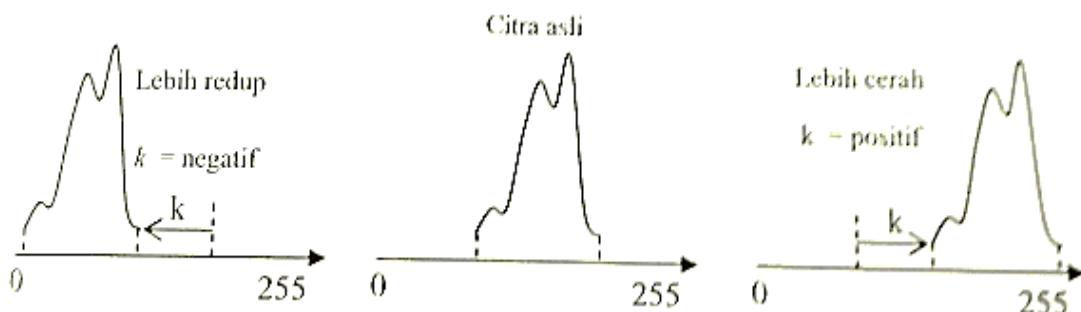
$$f_o(x, y) = f_i(x, y) + k$$

Dimana f_o = nilai intensitas (warna) setelah dilakukan operasi fungsi kecerahan

f_i = nilai intensitas (warna) sebelum dilakukan operasi fungsi kecerahan

k = konstanta kecerahan

Apabila k bernilai positif maka citra hasil operasi lebih cerah bila dibandingkan citra asli namun bila k bernilai negatif maka citra hasil operasi lebih redup atau gelap bila dibandingkan dengan citra asli. Penambahan nilai k pada operasi kecerahan akan mempengaruhi histogram dari citra tersebut seperti yang ditunjukkan pada Gambar 11.



Gambar 13. Hasil histogram citra pada operasi kecerahan

Kontras berhubungan terkait sebaran terang (*lightness*) dan gelap (*darkness*). Proses ini bertujuan untuk mengatasi kekurangan cahaya atau kelebihan cahaya pada saat pengambilan citra. Caranya dengan memperluas sebaran nilai keabuan piksel. Proses ini juga dikenal dengan nama *contrast stretching*

Citra dibagi menjadi 3 kontras seperti yang ditunjukkan pada Gambar 12:

1. Citra Kontras Rendah

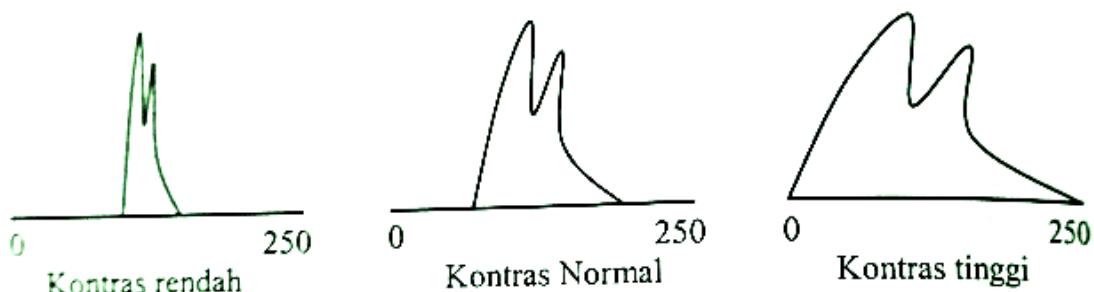
Citra disebut berkontras rendah biasanya terjadi akibat kurangnya pencahayaan sehingga memiliki kurva histogram yang sempit.

2. Citra Kontras Tinggi

Citra disebut berkontras tinggi memiliki kurva histogram yang terlalu lebar sehingga sebaran intensitas terang dan gelap merata ke seluruh skala intensitas.

3. Citra Kontras Normal

Citra tersebut memiliki kurva histogram tidak terlalu sempit dan tidak terlalu lebar.



Gambar 14. Histogram citra pada operasi kontras

d. Alat dan Bahan

1. Kertas A4
2. Pulpen
3. Laptop
4. Citra
5. Matlab 2020a

e. Prosedur Kerja

1. siapkan foto dan matlab



2. kemudian tulis sourcecode berikut :

```
gambar = imread('E:\1.Vani\9.Penyakit Tanaman Timun\Embut Tepung (2).jpg');
daun = rgb2gray(gambar);
cerah = daun + 50;
gelap = daun - 50;
subplot(1,4,1),imshow(gambar),title('citra asli');
subplot(1,4,2),imshow(daun),title('citra grayscale');
subplot(1,4,3),imshow(cerah),title('citra brightness');
subplot(1,4,4),imshow(gelap),title('citra darkness');
```

3. lakukan percobaan operasi *contrast stretching* dengan sourcecode berikut :

```
gambar = imread('E:\1.Vani\9.Penyakit Tanaman Timun\Embut Tepung (2).jpg');
daun = rgb2gray(gambar);
contrast = imadjust(daun);
subplot(1,3,1),imshow(gambar),title('citra asli');
subplot(1,3,2),imshow(daun),title('citra grayscale');
subplot(1,3,3),imshow(cerah),title('citra contrast stretching');
```

f. Hasil dan Pembahasan

1. Lakukan prosedur kerja dan analisa hasil pekerjaan
2. Analisa hasil histogram pada citra brightness dan citra darkness
3. Dokumentasi Tugas berupa laporan (definisi, proses dan penerapan histogram citra)
4. Dokumentasi Tugas berupa file ppt

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

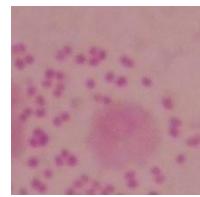
Acara 12

Pokok Bahasan	:Image Enhancement
Acara Praktikum/Praktik	:Minggu 3/4
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	:100 menit

- a. Capaian Pembelajaran Mata Kuliah (CPMK)
 1. Mahasiswa mampu memahami definisi dari pengolahan citra dan vision.
 2. Mahasiswa mampu memahami proses Image Enhancement.
 3. Mahasiswa mampu memberikan contoh penerapan teknik image enhancement
- b. Indikator Penilaian
Ketepatan dalam menjelaskan mengenai teknik image enhancement
- c. Dasar Teori
Untuk memperoleh histogram citra sesuai dengan keinginan kita, maka penyebaran nilai-nilai intensitas pada citra harus diubah. Terdapat dua metode pengubahan citra berdasarkan histogram:
 1. Perataan historam (histogram equalization)
Nilai-nilai intensitas di dalam citra diubah sehingga penyebarannya seragam (uniform).
 2. Spesifikasi histogram (histogram spesification)
Nilai-nilai intensitas di dalam citra diubah agar diperoleh histogram dengan bentuk yang dispesifikasikan oleh pengguna.
- d. Alat dan Bahan
 1. Kertas A4
 2. Pulpen
 3. Laptop
 4. Citra
 5. Matlab 2020a
- e. Prosedur Kerja
 1. siapkan foto dan matlab



(a)



(b)

2. kemudian tulis sourcecode berikut :

```
gambar = imread('E:\1.Vani\8. identifikasi bakteri\data crop\Neiseria gonorrhoea\go_59.jpg');
bakteri = rgb2gray(gambar);
HQ=histeq(TBC);
subplot(1,3,1), imshow(gambar), title('citra asli');
subplot(1,3,2), imshow(bakteri), title('citra grayscale');
subplot(1,3,3), imshow(HQ), title('Hist. Equalization');
```

3. bandingkan dengan hasilnya dengan hasil *contrast stretching* yang telah dilakukan minggu lalu.

4. Lakukan analisa pada hasil tersebut !

f. Hasil dan Pembahasan

1. Lakukan prosedur kerja dan analisa hasil pekerjaan
2. Analisa hasil histogram pada histogram equalization
3. Dokumentasi Tugas berupa laporan (definisi, proses dan penerapan histogram citra)
4. Dokumentasi Tugas berupa file ppt

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 13

Pokok Bahasan	:Konvolusi (1)
Acara Praktikum/Praktik	:Minggu 4/1
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	:100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu menjelaskan konsep operasi konvolusi pada citra digital.
2. Mahasiswa mampu menerapkan operasi konvolusi pada aplikasi pengolahan citra dan vision.

b. Indikator Penilaian

Kemampuan mahasiswa dalam menjelaskan dan mengimplementasikan operasi Konvolusi pada citra dengan berbagai kernel

c. Dasar Teori

Konvolusi 2D didefinisikan sebagai proses untuk memperoleh suatu piksel didasarkan pada nilai piksel itu sendiri dan tetangganya, dengan melibatkan suatu matriks yang disebut **kernel** sebagai representasi pembobotan. Operasi Konvolusi, Menggeser kernel piksel per piksel hasilnya disimpan dalam matriks baru
Konvolusi 2 buah fungsi $f(x)$ dan $g(x)$ didefinisikan sbb:

$$h(x) = f(x) * g(x) = \int_{-\infty}^{\infty} f(a)g(x-a)da$$

Dalam hal ini tanda $*$ menyatakan operator konvolusi dan peubah (variabel) a adalah *dummy variable*. Untuk fungsi diskrit, konvolusi didefinisikan sebagai:

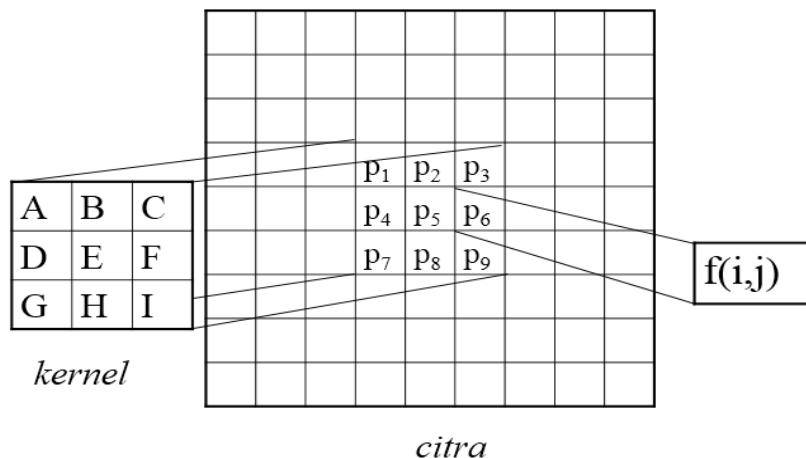
$$h(x) = f(x) * g(x) = \sum_{a=-\infty}^{\infty} f(a)g(x-a)$$

Dimana : $g(x)$ adalah kernel, **Kernel** merupakan suatu matrik.

Operasi Konvolusi digunakan pada:

- Perbaikan Citra (image enhancement)
 - Morfologi Citra – kernel disebut strel
- Penghilang derau (noise)
 - Filtering (Low/High Pass Filter)– kernel disebut mask
- Penghalusan/pelembutan citra
- Deteksi Tepi (penajaman tepi)

Biasanya kernel konvolusi dinyatakan dalam matriks (umumnya 3x3, 2x2 atau 2x1). Ukuran matriks ini biasanya lebih kecil dari matriks citra. Setiap elemen matriks disebut koefisien konvolusi



Gambar 15. Operasi Konvolusi

(Sumber : <https://dufalhakujin.wordpress.com/2016/10/23/teknik-konvolusi-transformasi-fourier-ft/>)

$$f(i, j) = Ap_1 + Bp_2 + Cp_3 + Dp_4 + Ep_5 + Fp_6 + Gp_7 + Hp_8 + Ip_9$$

Hasil **$f(i,j)$** diletakkan pada titik tengah piksel yang dikenakan kernel. Pada contoh diatas di posisi piksel **p5**

- d. Alat dan Bahan
 - 1. Kertas A4
 - 2. Pulpen
 - 3. Laptop
 - 4. Aplikasi spreadsheet (Ms. Excel)
- e. Prosedur Kerja
 1. Tuliskan contoh matrik critra asli dan kernel berikut ini dalam lembar/kertas A4 Anda

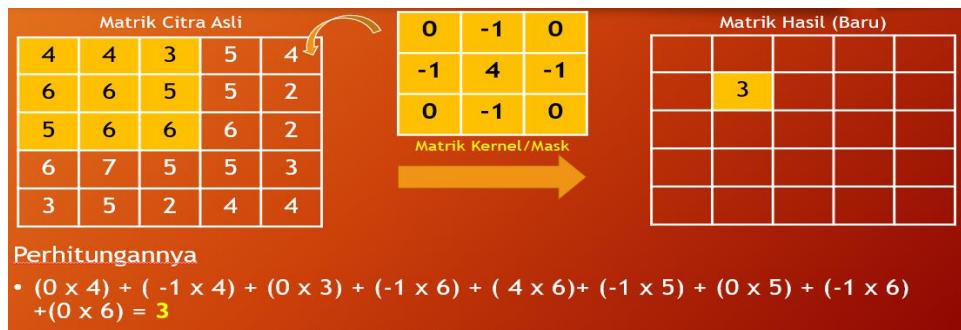
Matrik Citra Asli

4	4	3	5	4
6	6	5	5	2
5	6	6	6	2
6	7	5	5	3
3	5	2	4	4

Matrik Kernel/Mask

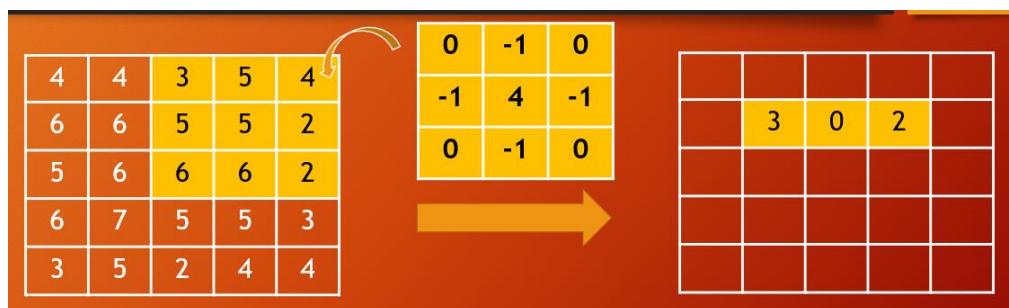
0	-1	0
-1	4	-1
0	-1	0

2. Hitung satu persatu operasi konvolusi dengan cara berikut

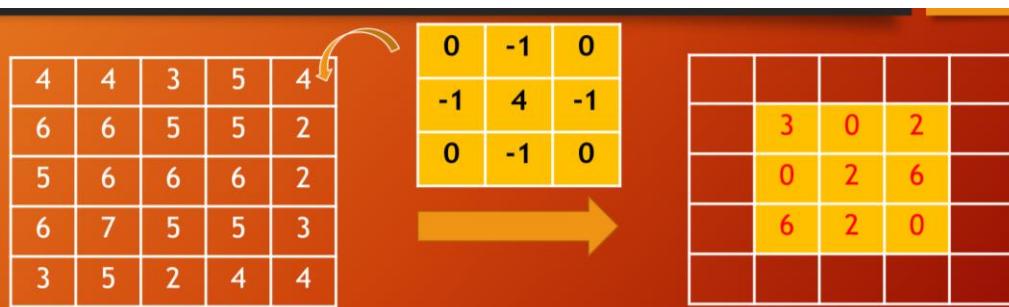
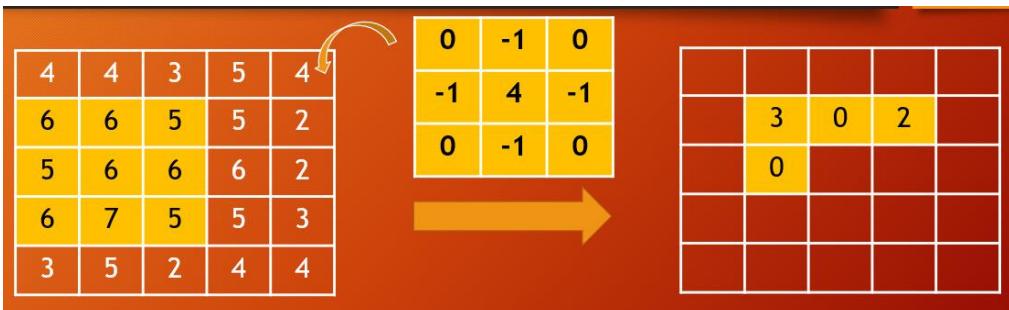


Letakkan angka hasil perhitungan, di titik pusat piksel yang dikenakan operasi konvolusi tersebut.

3. Lanjutkan hingga semua piksel dalam satu baris dioperasikan dengan matriks kernel.



4. Lanjutkan ke baris dibawahnya, hingga semua piksel dioperasikan.



- ❑ Bila hasil konvolusi negatif, maka nilai dijadikan 0.
- ❑ Bila hasil konvolusi > derajat keabuan maksimum, maka nilai diubah ke derajat keabuan maksimum (nilai lebih dari 255 maka di set 255)

f. Hasil dan Pembahasan

1. Bagaimana dengan piksel yang ada di tepi/pinggir citra?

4	4	3	5	4	?
6	6	5	5	2	?
5	6	6	6	2	?
6	7	5	5	3	
3	5	2	4	4	

Bagaimana piksel pinggir?

2. Piksel-piksel pinggir **diabaikan**, tidak dikonvolusi. Nilai piksel pinggir menggunakan nilai pada citra semula.
3. **Duplikasi** elemen citra, misalnya elemen kolom pertama disalin ke kolom M-1 dst.
4. Elemen bertanda “?” diasumsikan bernilai **0** atau konstanta lain, sehingga hasil akhir seperti ini

4	4	3	5	4
6	3	0	2	2
5	0	2	6	2
6	6	2	0	3
3	5	2	4	4

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 14

Pokok Bahasan	:Konvolusi (2)
Acara Praktikum/Praktik	:Minggu 4/2
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	:100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu menjelaskan konsep operasi konvolusi pada citra digital.
2. Mahasiswa mampu menerapkan operasi konvolusi pada aplikasi pengolahan citra dan vision.

b. Indikator Penilaian

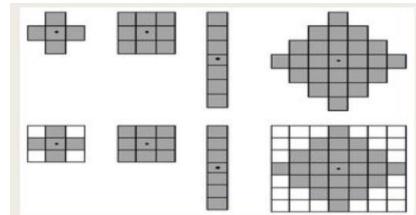
Kemampuan mahasiswa dalam menjelaskan dan mengimplementasikan operasi Konvolusi pada citra dengan berbagai kernel

c. Dasar Teori

Kernel dalam pembahasan pengolahan citra digital merupakan matriks kecil yang digunakan dalam konvolusi. Dalam beberapa teknik pengolahan citra digital, kernel dikenal juga dengan istilah mask (biasanya digunakan dalam filtering) dan strel – structuring element (dalam operasi morfologi). Beberapa kernel yang umum digunakan antara lain:

1. Morfologi

Diagram showing several binary kernels for morphological operations, including a 3x3 square, a 3x3 cross, vertical and horizontal lines, and a 1x5 horizontal line with the third cell highlighted in blue.



2. Filtering

$$\bullet \text{ High Pass Filter}$$
$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{bmatrix} \quad \begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix} \quad \begin{bmatrix} 1 & -2 & 1 \\ -2 & 5 & -2 \\ 1 & -2 & 1 \end{bmatrix}$$

• Low Pass Filter

$$\begin{bmatrix} \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \end{bmatrix} \quad \begin{bmatrix} \frac{1}{16} & \frac{1}{8} & \frac{1}{16} \\ \frac{1}{8} & \frac{1}{4} & \frac{1}{8} \\ \frac{1}{16} & \frac{1}{8} & \frac{1}{16} \end{bmatrix} \quad \begin{bmatrix} \frac{1}{10} & \frac{1}{10} & \frac{1}{10} \\ \frac{1}{10} & \frac{1}{5} & \frac{1}{10} \\ \frac{1}{10} & \frac{1}{10} & \frac{1}{10} \end{bmatrix}$$

3. Deteksi Tepi

-1	0
0	1

0	-1
1	0

-1	-2	-1
0	0	0
1	2	1

-1	0	1
-2	0	2
-1	0	1

-1	-1	-1
0	0	0
1	1	1

-1	0	1
-1	0	1
-1	0	1

Robert

Sobel

Prewitt

d. Alat dan Bahan

1. Kertas A4
2. Pulpen
3. Laptop
4. Aplikasi spreadsheet (Ms. Excel)

e. Prosedur Kerja

1. Tuliskan nilai-nilai matriks citra berikut didalam file *spreadsheet*

62	195	179	179	179	179	179	179	166	109
127	133	92	91	91	90	89	89	79	-9.7
97	82	60	55	49	44	38	33	76	14
83	64	115	111	97	83	81	45	75	84
126	87	111	85	27	3.5	7.1	13	43	84
119	143	76	10	-4.1	11	8.9	70	119	63
117	59	38	-47	24	23	13	74	127	28
120	53	17	63	46	35	33	110	162	28
124	108	90	138	117	131	131	115	187	74
204	126	127	128	129	130	126	114	143	63

2. Diberikan matriks kernel sebagai berikut

-0,5	0,2	0,1
0,2	0,1	0,4
0,3	-0,1	-0,2

3. Lakukan perhitungan operasi Konvolusi pada matriks citra asli dan simpan hasilnya di matriks baru, diskusikan bersama teman atau Dosen pengampu.

f. Hasil dan Pembahasan

-

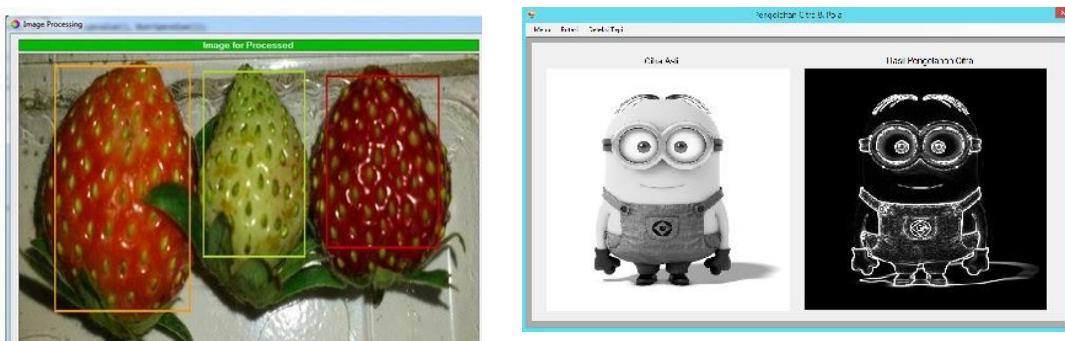
g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 15

Pokok Bahasan	: Segmentasi Citra - ROI
Acara Praktikum/Praktik	: Minggu 4/3
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	: 100 menit

- a. Capaian Pembelajaran Mata Kuliah (CPMK)
 1. Mahasiswa mampu menjelaskan konsep segmentasi citra.
 2. Mahasiswa mampu menerapkan teknik segmentasi citra berdasarkan *Region Of Interest*.
- b. Indikator Penilaian
Kemampuan mahasiswa dalam menjelaskan dan mengimplementasikan segmentasi citra berdasarkan *Region Of Interest*.
- c. Dasar Teori
Segmentasi menurut KBBI ialah Pembagian dalam segmen (satu bagian yang memiliki struktur yang sama). **Segmentasi Citra** artinya membagi citra menjadi region-region atau objek-objek. Level sampai sejauh mana pembagian bisa dilakukan tergantung pada permasalahan yang diselesaikan.



Gambar 16. Contoh hasil segmentasi citra

Algoritma-algoritma segmentasi didasarkan pada karakteristik nilai derajat kecerahan citra, yaitu:

- **Diskontinuitas**, pendekatannya dengan mempartisi citra berdasarkan pada perubahan intensitas yang menonjol, seperti deteksi titik, garis, dan tepi citra. Cara yang paling umum digunakan untuk mencari diskontinuitas adalah dengan menjalankan suatu *filter (kernel)* pada seluruh area citra.
- **Similaritas**, mempartisi/mensegmentasi citra kedalam region/wilayah yang memiliki kesamaan/kemiripan menurut kriteria yang sudah ditentukan(Region-Based). Thresholding (telah dibahas di pertemuan awal tentang binerisasi citra), region growing, dan region splitting/merging

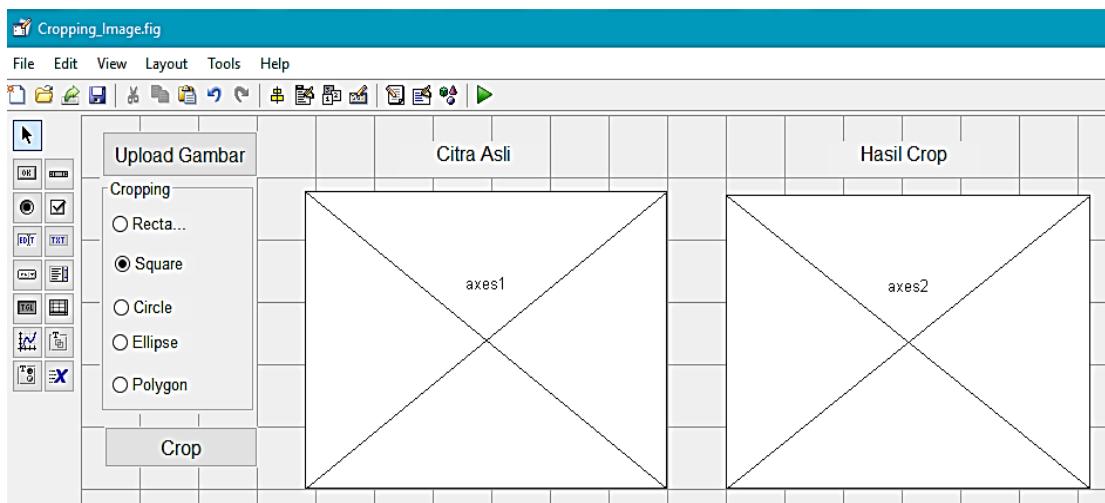
Region Of Interest (ROI) mampu memberikan tanda terhadap area tertentu sehingga dapat digunakan untuk mengoptimalkan kinerja sistem. Tanpa penggunaan ROI, pemrosesan pada citra dilakukan pada seluruh piksel citra tanpa terkecuali. ROI merupakan salah satu teknik segmentasi sebagai proses pengolahan citra dimana pengguna mampu mengolah citra yang mengandung informasi data citra yang dikehendaki. Proses untuk mendapatkan ROI salah satunya adalah dengan cara melakukan *cropping* pada suatu citra

d. Alat dan Bahan

1. Kertas A4
2. Pulpen
3. Laptop
4. Citra (image)
5. Aplikasi Matlab

e. Prosedur Kerja

1. Buat rancangan User Interface seperti berikut ini, didalam Matlab Anda, dengan komponen-komponen 2 tombol, 5 radio button, dan 2 *image frame*.



2. Kemudian tuliskan kode program berikut, dibagian *editor code*

```

function varargout = Cropping_Image(varargin)
%
% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',         mfilename, ...
    'gui_Singleton',    gui_Singleton, ...
    'gui_OpeningFcn',   @Cropping_Image_OpeningFcn, ...
    'gui_OutputFcn',    @Cropping_Image_OutputFcn, ...
    'gui_LayoutFcn',   [], ...
    'gui_Callback',     []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

% --- Executes just before Cropping_Image is made visible.
function Cropping_Image_OpeningFcn(hObject, eventdata, handles, varargin)
handles.output = hObject;

guidata(hObject, handles);
movegui(hObject,'center');
clc;clear;

% --- Outputs from this function are returned to the command line.
function varargout = Cropping_Image_OutputFcn(hObject, eventdata, handles)
varargout{1} = handles.output;

```

3. Di bagian pushbutton1 (**Upload Gambar**), tuliskan kode berikut.

```

% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)

axes(handles.axes1)
cla reset
set(gca,'XTick',[])
set(gca,'YTick',[])

axes(handles.axes2)
cla reset
set(gca,'XTick',[])
set(gca,'YTick',[])

[filename pathname] = uigetfile({'*.*'});

```

```

if isequal(filename,0)
    return
else
    Info = imfinfo(fullfile(pathname,filename));
    BitDepth = Info.BitDepth;
    if BitDepth == 1
        msgbox('Please insert RGB image or Grayscale image')
        return
    end
    Img = imread(fullfile(pathname,filename));
    axes(handles.axes1)
    cla reset
    set(gca,'XTick',[])
    set(gca,'YTick',[])
    imshow(Img)
end

set(handles.radioButton1,'Enable','on')
set(handles.radioButton2,'Enable','on')
set(handles.radioButton3,'Enable','on')
set(handles.radioButton4,'Enable','on')
set(handles.radioButton5,'Enable','on')
set(handles.pushbutton2,'Enable','on')

handles.Img = Img;
handles.BitDepth = BitDepth;
guidata(hObject, handles)

```

4. Lanjutkan dengan menulis kode program untuk masing-masing RadioButton

```

% --- Executes on button press in radioButton1.
function radioButton1_Callback(hObject, eventdata, handles)

    set(handles.radioButton1,'Value',1)
    set(handles.radioButton2,'Value',0)
    set(handles.radioButton3,'Value',0)
    set(handles.radioButton4,'Value',0)
    set(handles.radioButton5,'Value',0)

% --- Executes on button press in radioButton2.
function radioButton2_Callback(hObject, eventdata, handles)

    set(handles.radioButton1,'Value',0)
    set(handles.radioButton2,'Value',1)
    set(handles.radioButton3,'Value',0)
    set(handles.radioButton4,'Value',0)
    set(handles.radioButton5,'Value',0)

```

```

% --- Executes on button press in radiobutton3.
function radiobutton3_Callback(hObject, eventdata, handles)

    set(handles.radioButton1,'Value',0)
    set(handles.radioButton2,'Value',0)
    set(handles.radioButton3,'Value',1)
    set(handles.radioButton4,'Value',0)
    set(handles.radioButton5,'Value',0)

% --- Executes on button press in radiobutton4.
function radiobutton4_Callback(hObject, eventdata, handles)

    set(handles.radioButton1,'Value',0)
    set(handles.radioButton2,'Value',0)
    set(handles.radioButton3,'Value',0)
    set(handles.radioButton4,'Value',1)
    set(handles.radioButton5,'Value',0)

% --- Executes on button press in radiobutton5.
function radiobutton5_Callback(hObject, eventdata, handles)

    set(handles.radioButton1,'Value',0)
    set(handles.radioButton2,'Value',0)
    set(handles.radioButton3,'Value',0)
    set(handles.radioButton4,'Value',0)
    set(handles.radioButton5,'Value',1)

```

5. Kemudian untuk pushbutton2 (Tombol **Crop**)

```

% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)

Img = handles.Img;
[m,n,~] = size(Img);

val1 = get(handles.radioButton1,'Value');
val2 = get(handles.radioButton2,'Value');
val3 = get(handles.radioButton3,'Value');
val4 = get(handles.radioButton4,'Value');
val5 = get(handles.radioButton5,'Value');

axes(handles.axes2)
cla reset
set(gca,'XTick',[])
set(gca,'YTick',[])

axes(handles.axes1)
cla reset
set(gca,'XTick',[])
set(gca,'YTick',[])

imshow(Img);

```

```

if val1 == 1
    h = imrect(gca, [n/2 m/2 0.2*n 0.2*m]);
    wait(h);
    mask = createMask(h);
elseif val2 == 1
    h = imrect(gca, [n/2 m/2 0.2*m 0.2*m]);
    setFixedAspectRatioMode(h, 'true')
    wait(h);
    mask = createMask(h);
elseif val3 == 1
    h = imellipse(gca, [n/2 m/2 0.2*m 0.2*m]);
    setFixedAspectRatioMode(h, 'true')
    wait(h);
    mask = createMask(h);
elseif val4 == 1
    h = imellipse(gca, [n/2 m/2 0.2*n 0.2*m]);
    wait(h);
    mask = createMask(h);
elseif val5 == 1
    h = impoly(gca);
    wait(h);
    mask = createMask(h);
end

[B,~] = bwboundaries(mask, 'noholes');
for k = 1:length(B)
    boundary = B{k};
    axes(handles.axes1);
    cla reset
    set(gca, 'XTick', [])
    set(gca, 'YTick', [])
    imshow(Img);
    hold on
    plot(boundary(:,2), boundary(:,1), 'b', 'LineWidth', 2);
    hold off
end

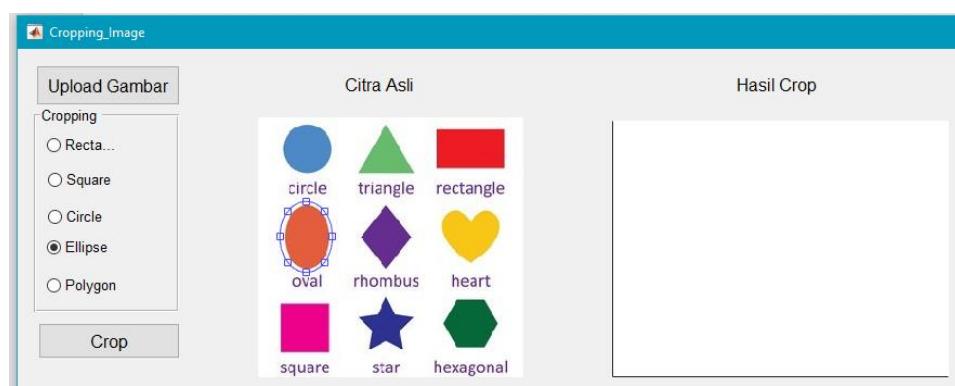
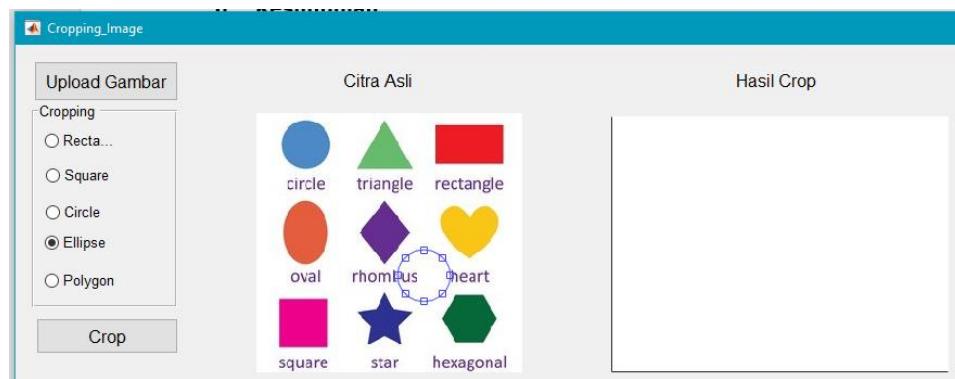
if handles.BitDepth == 8
    Img(~mask) = 0;
    [row, col] = find(mask==1);
    h_Gray = imcrop(Img, [min(col) min(row) max(col)-min(col) max(row)-min(row)]);
    axes(handles.axes2)
    imshow(h_Gray);
else
    R = Img(:,:,1);
    G = Img(:,:,2);
    B = Img(:,:,3);
    R(~mask) = 0;
    G(~mask) = 0;
    B(~mask) = 0;
    RGB = cat(3,R,G,B);
    [row, col] = find(mask==1);
    h_RGB = imcrop(RGB, [min(col) min(row) max(col)-min(col) max(row)-min(row)]);
    axes(handles.axes2)
    imshow(h_RGB);
end

```

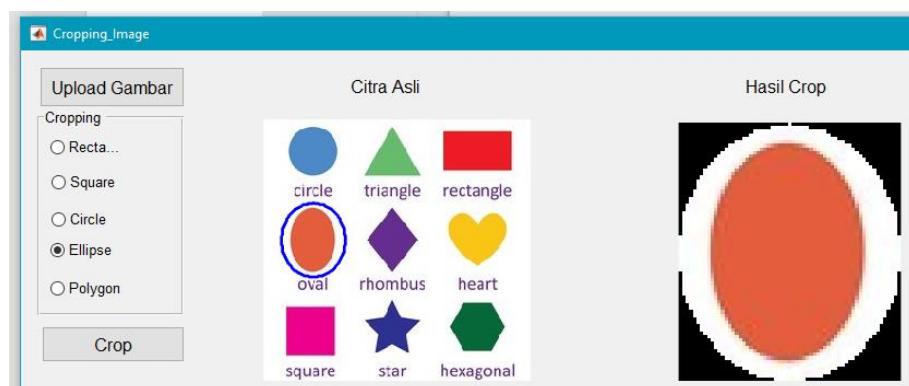
f. Hasil dan Pembahasan

Periksa kembali kode program, dan perbaiki jika terjadi *error*. Kemudian jalankan.

- Pilih jenis bentuk mask untuk **crop**, misal **Ellipse**, kemudian akan muncul tanda, silahkan letakkan tanda tersebut ke area yang akan di seleksi dengan **Klik, tahan dan geser**. Anda dapat mengatur bentuk dan ukuran dari **tanda** tersebut.



- Kemudian **Double Click** pada area citra yang ada dalam tanda tersebut, maka hasilnya seperti ini



g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 16

Pokok Bahasan	:Segmentasi Citra – Background removal
Acara Praktikum/Praktik	:Minggu 4/4
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	:100 menit

- a. Capaian Pembelajaran Mata Kuliah (CPMK)
 1. Mahasiswa mampu menjelaskan konsep segmentasi citra.
 2. Mahasiswa mampu menerapkan teknik segmentasi citra – *Background removal*.
- b. Indikator Penilaian

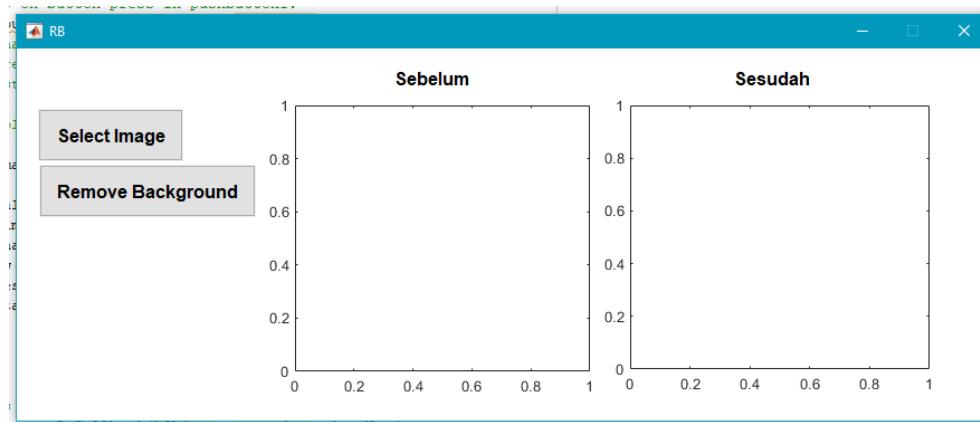
Kemampuan mahasiswa dalam menjelaskan dan mengimplementasikan segmentasi citra – *Background removal*.
- c. Dasar Teori

Mayoritas teknik deteksi citra tergantung melibatkan pengaturan latar belakang dan mendeteksi perubahan mana yang terjadi. Mendefinisikan latar belakang bisa sangat sulit bila berisi bentuk, bayangan, dan objek bergerak. Dalam mendefinisikan latar belakang, diasumsikan bahwa objek diam dapat bervariasi dalam warna dan intensitas dari waktu ke waktu. Sistem deteksi latar depan yang sangat baik harus dapat: (1) Memperkirakan latar belakang dengan tepat, (2) Handal terhadap perubahan pencahayaan, gerakan berulang (daun, gelombang, bayangan).

Background didefinisikan bagian citra (lingkungan) yang memiliki ketidaksamaan dengan objek. *Background removal* adalah prosedur *digital image processing* yang dapat digunakan untuk mengklasifikasikan bagian-bagian dari suatu gambar dalam hal ini wilayah yang tidak diinginkan. Banyak aplikasi *digital image processing* dan *computer vision* memerlukan *Background removal* sebelum analisis dan pemrosesan lebih lanjut.
- d. Alat dan Bahan
 1. Kertas A4
 2. Pulpen
 3. Laptop
 4. Citra (image)
 5. Aplikasi Matlab
- e. Prosedur Kerja
 1. Siapkan gambar, coba dulu gambar obyek yang memiliki *background* homogen (tidak variatif) misal seperti ini



2. Buka aplikasi Matlab anda → Buat Proyek baru → Pilih menu Home □ New □ App kemudian setting user interface seperti ini



3. Tambahkan kode program ini dalam editor

```
[function varargout = RB(varargin)

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',         mfilename, ...
                   'gui_Singleton',    gui_Singleton, ...
                   'gui_OpeningFcn',   @RB_OpeningFcn, ...
                   'gui_OutputFcn',    @RB_OutputFcn, ...
                   'gui_LayoutFcn',    [] , ...
                   'gui_Callback',     []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

]function RB_OpeningFcn(hObject, eventdata, handles, varargin)
handles.output = hObject;

guidata(hObject, handles);

]function varargout = RB_OutputFcn(hObject, eventdata, handles)
varargout{1} = handles.output;
```

4. Dibagian Tombol **Select Image**, isi dengan program berikut, dimana variabel **Im** nantinya akan menangkap citranya.

```
% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)

global Im
[nama_file,nama_path]= uigetfile({'*.*'});

if ~isequal(nama_file,0)
    Im = imread(fullfile(nama_path,nama_file));
    axes(handles.axes1)
    imshow(Im)
    handles.Img=Im;
    guidata(hObject,handles)
else

end
```

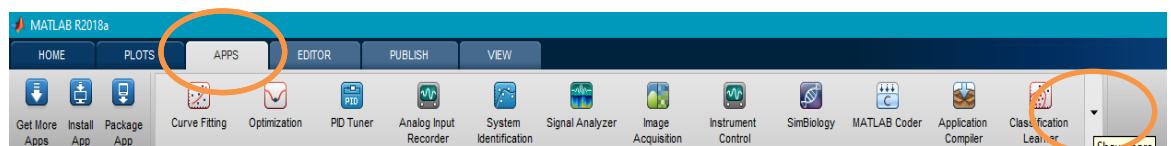
5. Dibagian Tombol **Remove Background**, isi dengan program berikut

```
% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)

global Im
[BW, Img] = remove_bg(Im);
axes(handles.axes2)
imshow(Img)
```

6. Seleksi area gambar, yang akan dihapus (*background*) menggunakan pendekatan ROI, harus didahului dengan memberikan "referensi" dengan cara berikut

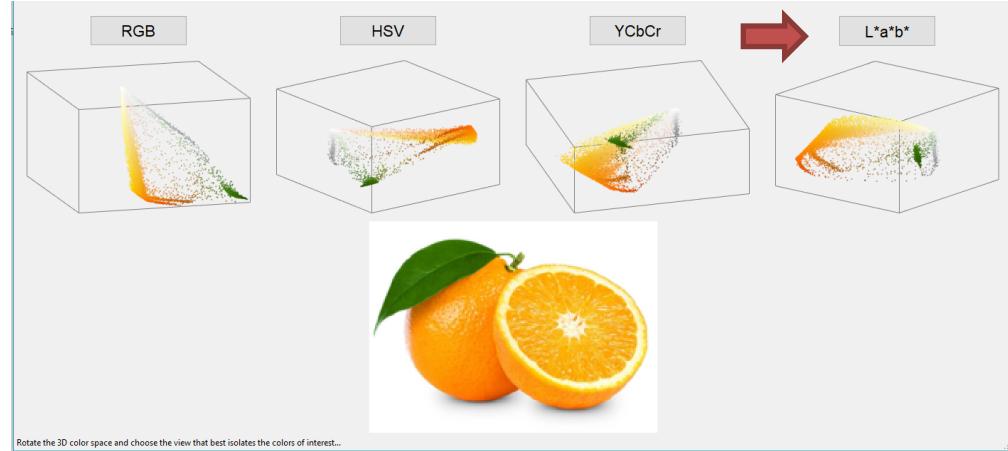
- Klik menu **APP** dibagian atas GUI Matlab
- Klik panah bawah



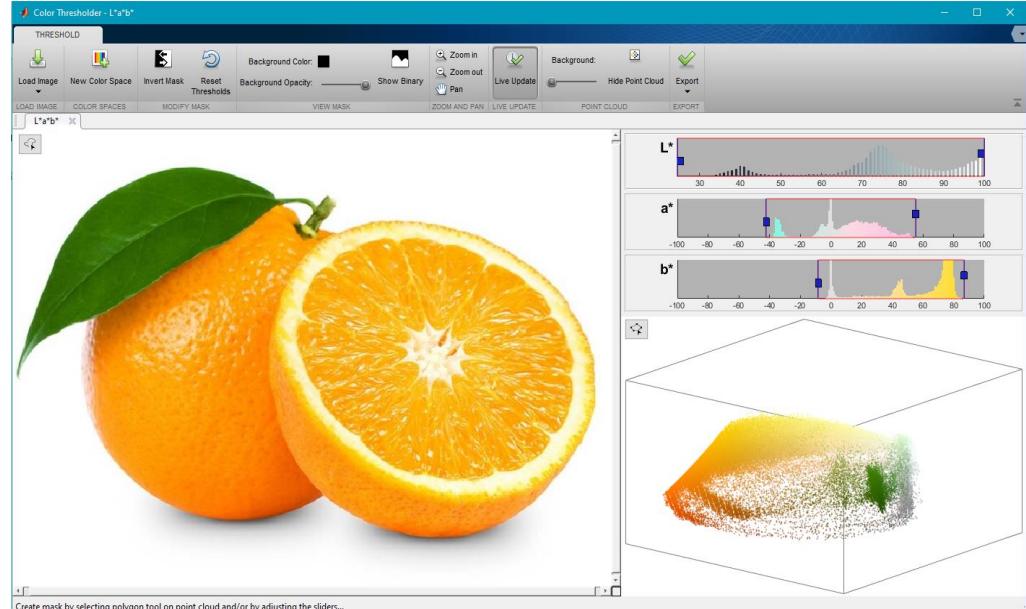
- Di bagian **Image Processing and Computer Vision**, pilih **Color Thresholder**



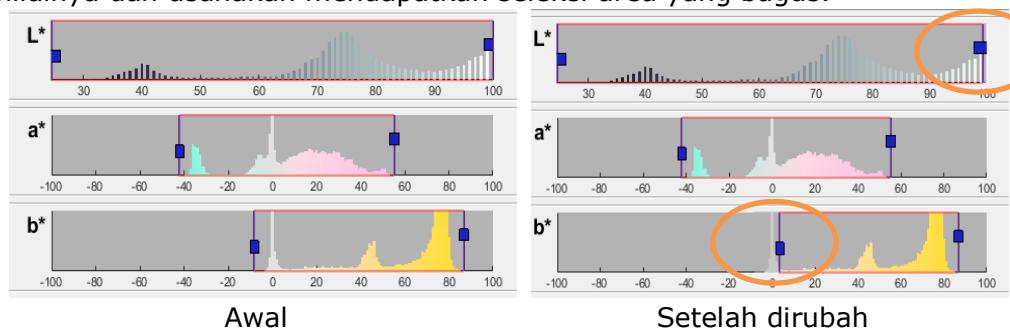
Kemudian pilih *Load Image* *Load Image From File*, arahkan ke lokasi penyimpanan gambar Anda, kemudian akan muncul seperti ini



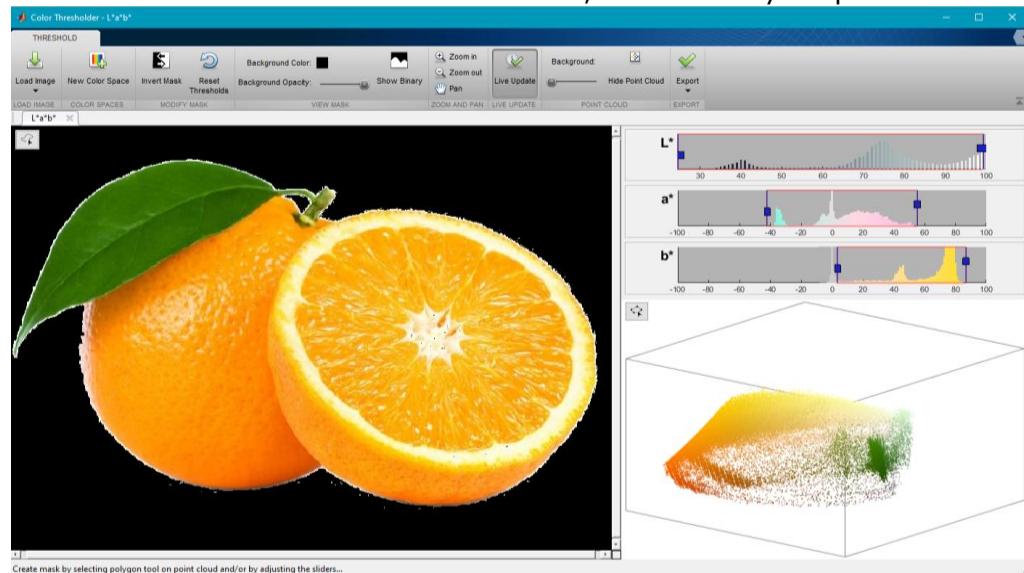
- d) Pilih salah satu (*double click* dibagian nama) channel warna, Misal **L*a*b***, maka akan muncul jendela baru



- e) Lakukan perubahan pada nilai parameter L^* atau a^* atau b^* , geset batas nilainya dan usahakan mendapatkan seleksi area yang bagus.



- f) Dilakukan perubahan sedikit di channel L* batas atas kurang dari 100 dan channel b* batas bawah ditambah dari nol, maka hasilnya seperti ini



- g) Kemudian klik **Export** **Export Function**, akan menghasilkan satu file baru, kemudian disimpan dengan nama **remove_bg.m** | yang akan dipanggil dalam kode program **[BW, Img] = remove_bg(Img);**. Berikut hasil kode program dari export function tersebut

```
% Convert RGB image to chosen color space
I = rgb2lab(RGB);

% Define thresholds for channel 1 based on histogram settings
channel1Min = 24.542;
channel1Max = 99.290;

% Define thresholds for channel 2 based on histogram settings
channel2Min = -41.946;
channel2Max = 55.284;

% Define thresholds for channel 3 based on histogram settings
channel3Min = 3.608;
channel3Max = 86.821;

% Create mask based on chosen histogram thresholds
sliderBW = (I(:,:,:,1) >= channel1Min) & (I(:,:,:,1) <= channel1Max) & ...
            (I(:,:,:,2) >= channel2Min) & (I(:,:,:,2) <= channel2Max) & ...
            (I(:,:,:,3) >= channel3Min) & (I(:,:,:,3) <= channel3Max);
BW = sliderBW;

% Initialize output masked image based on input image.
maskedRGBImage = RGB;

% Set background pixels where BW is false to zero.
maskedRGBImage(repmat(~BW,[1 1 3])) = 0;

end
```

f. Hasil dan Pembahasan

Periksa kembali kode program, dan perbaiki jika terjadi *error*. Kemudian jalankan hasil seperti ini.



g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 17

Pokok Bahasan	:Operasi Citra Biner
Acara Praktikum/Praktik	:Minggu 5/1
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	:100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

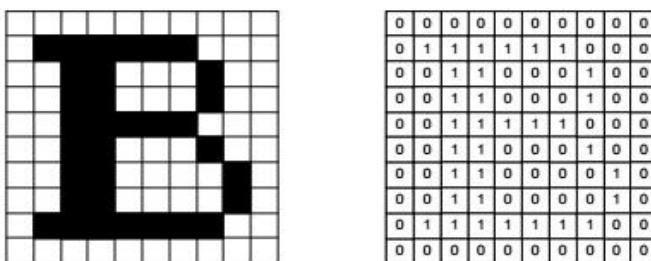
1. Mahasiswa mampu memahami definisi dari pengolahan citra dan vision.
2. Mahasiswa mampu memahami proses operasi citra biner.
3. Mahasiswa mampu memberikan contoh penerapan operasi citra biner.

b. Indikator Penilaian

Ketepatan dalam menjelaskan mengenai operasi citra biner

c. Dasar Teori

Citra biner (*binary image*) merupakan citra yang hanya mempunyai dua nilai derajat keabuan yaitu hitam (0) dan putih (1).



Gambar 17. Huruf "B" dan representasi biner dari derajat keabuannya

Keuntungan dari citra biner :

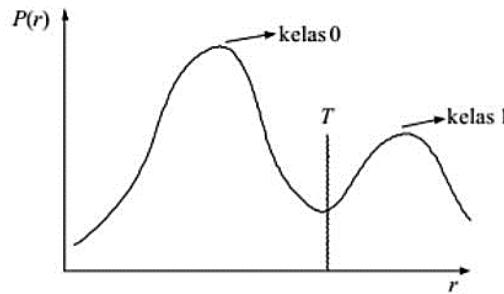
1. Kebutuhan memori kecil (nilai derajat keabuan hanya membutuhkan representasi 1 bit).
2. Waktu pemrosesan lebih cepat.
3. Diplikasikan untuk pemrosesan pengenalan objek (pengenalan karakter sec. optik, analisis kromosom, pengenalan sparepart komponen industri, dll).

Konversi citra grayscale ke citra biner dilakukan dengan operasi pengambangan (thresholding). Operasi ini mengelompokkan nilai derajat keabuan setiap piksel ke dalam 2 kelas, hitam dan putih. Dua pendekatan yang digunakan dalam operasi thresholding adalah *global image thresholding* dan *locally adaptive image thresholding*.

Pada *Global image thresholding*, setiap piksel di dalam citra dipetakan pada dua nilai yaitu 1 atau 0 dengan persamaan rumus pada satu batas ambang :

$$fb(i,j) = \begin{cases} 1, & fg(i,j) \leq T \\ 0, & \text{lainnya} \end{cases}$$

dimana $fg(i,j)$ adalah citra grayscale, $fb(i,j)$ adalah citra biner dan T adalah nilai ambang. Pada operasi ini, obyek diberi warna putih (1) sedangkan latarbelakang (background) berwarna hitam (0). Dalam menentukan nilai ambang (T) maka harus ditampilkan dahulu histogram citra. Kemudian dilakukan analisa untuk menentukan obyek dan background, nilai T dipilih pada nilai minimum local yang terdapat diantara dua puncak seperti pada Gambar 15.



Gambar 18. Penentuan nilai ambang T

Jika nilai intensitas obyek diketahui dalam selang $[T_1, T_2]$ maka kita dapat menggunakan persamaan rumus :

$$fb(i,j) = \begin{cases} 1, & T_1 \leq fg(i,j) \leq T_2 \\ 0, & \text{lainnya} \end{cases}$$

Pengembangan secara global tidak selalu tepat untuk seluruh macam citra. Untuk itu dilakukan proses pengembangan lokal dimana citra dipecah

- d. Alat dan Bahan
 - 1. Kertas A4
 - 2. Pulpen
 - 3. Laptop
 - 4. Citra (image)
 - 5. Aplikasi Matlab
- e. Prosedur Kerja
 - 1. Siapkan gambar dan buka aplikasi matlab



Gambar 19. Varietas mangga (a) apel, (b) gedong gincu, (c) gadung, (d) golek dan (e) manalagi

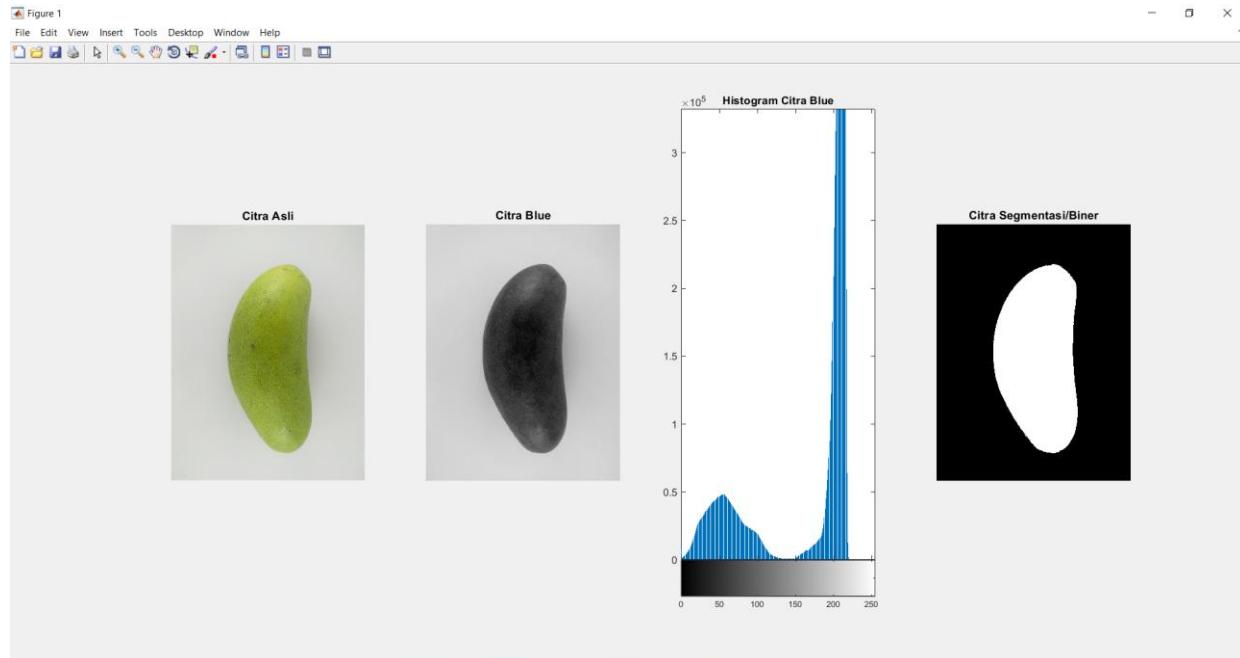
2. ketikkan sourcecode berikut :

```
gambar= imread('C:\Users\ACER\Downloads\training_208.jpg');
blue=gambar(:,:,3);
[M,N]=size(blue);
B_seg = zeros(M,N);
for k = 1 : M
    for l = 1 : N
        if blue(k,l)>120
            B_seg(k,l)=0;
        else
            B_seg(k,l)=1;
        end
    end
end
B_seg2=imfill(B_seg);
```

3. Menampilkan hasil kedalam subplot dengan sourcecode berikut :

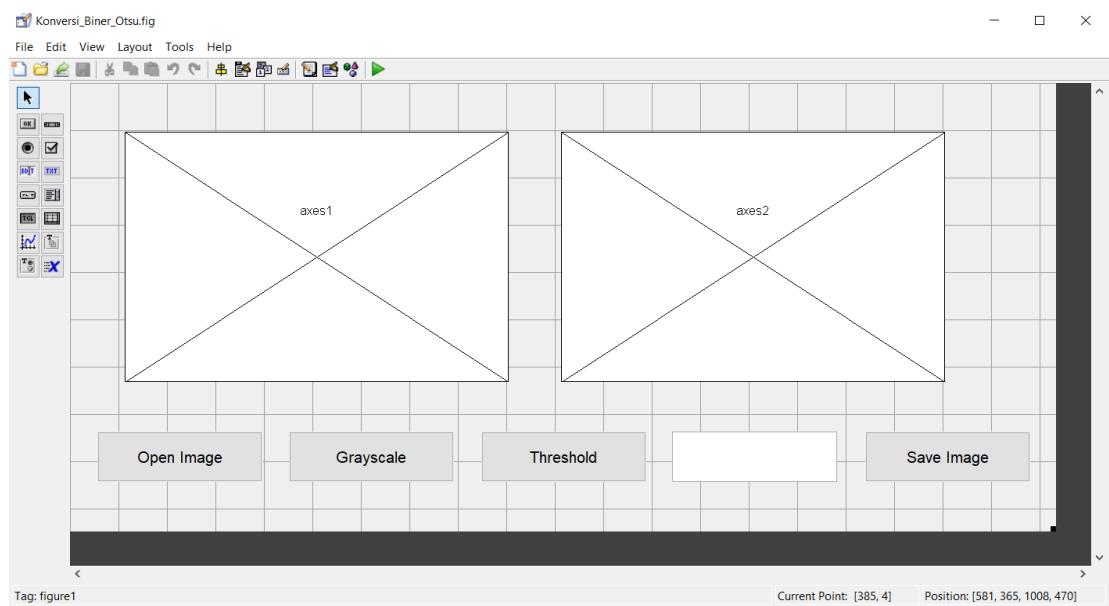
```
subplot(1,4,1),imshow(gambar);
subplot(1,4,2),imshow(blue);
subplot(1,4,3),imhist(blue);
subplot(1,4,4),imshow(B_seg2);
```

4. Tampilkan hasil seperti Gambar berikut :



Gambar 20. Hasil operasi citra biner

5. Buatlah GUI pada matlab seperti pada Gambar berikut :



Gambar 21. Tampilan GUI untuk Konversi Biner menggunakan Metode Otsu

6. Konversi citra biner menggunakan metode Otsu dengan sourcecode berikut:

```
% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
[name_file1,name_path1] = uigetfile( ...
{ '*.bmp;*.jpg;*.tif','Files of type (*.bmp,*.jpg,*.tif)';
 '*.bmp','File Bitmap (*.bmp)';...
 '*.jpg','File jpeg (*.jpg)';
 '*.tif','File Tif (*.tif)';
 '*.','All Files (*.*)'},...
'Open Image');
if ~isequal(name_file1,0)
    handles.data1 = imread(fullfile(name_path1,name_file1));
    guidata(hObject,handles);
    axes(handles.axes1);
    imshow(handles.data1);
else
    return;
end

% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
image1 = handles.data1;
gray = rgb2gray(image1);
axes(handles.axes2);
imshow(gray);
handles.data2 = gray;
guidata(hObject,handles);

% --- Executes on button press in pushbutton3.
function pushbutton3_Callback(hObject, eventdata, handles)
thresh = handles.data3;
[name_file_save,path_save] = uiputfile( ...
{ '*.bmp','File Bitmap (*.bmp)';
 '*.jpg','File jpeg (*.jpg)';
 '*.tif','File Tif (*.tif)';
 '*.','All Files (*.*)'},...
'Save Image');
if ~isequal(name_file_save,0)
    imwrite(thresh,fullfile(path_save,name_file_save));
else
    return;
end

% --- Executes on button press in pushbutton4.
function pushbutton4_Callback(hObject, eventdata, handles)
gray = handles.data2;
value = graythresh(gray);
thresh = im2bw(gray,value);
thresh = imcomplement(thresh);
axes(handles.axes2);
imshow(thresh);
handles.data3 = thresh;
guidata(hObject,handles);
set(handles.edit1,'String',value)
```

7. Bandingkan hasil citra biner menggunakan sourcecode pada nomer (2) dan menggunakan metode otsu,kemudian lakukan Analisa !

f. Hasil dan Pembahasan

1. Dokumentasi Tugas berupa laporan (definisi, proses dan penerapan operasi citra biner)
2. Dokumentasi Tugas berupa file ppt

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 18

Pokok Bahasan	:Operasi Morfologi
Acara Praktikum/Praktik	:Minggu 5/2
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	:100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami definisi dari pengolahan citra dan vision.
2. Mahasiswa mampu memahami proses operasi morfologi.
3. Mahasiswa mampu memberikan contoh penerapan operasi morfologi.

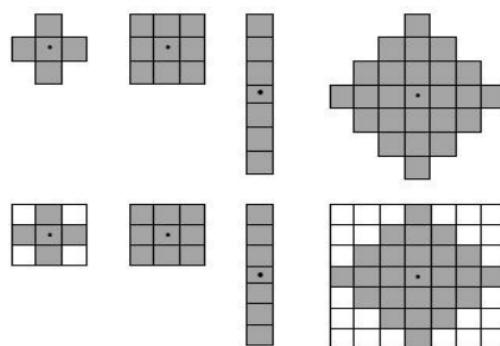
b. Indikator Penilaian

Ketepatan dalam menjelaskan mengenai operasi morfologi

c. Dasar Teori

Kata *morphology* merupakan cabang ilmu biologi yang memperlajari bentuk dan struktur hewan maupun tumbuhan. Sedangkan dalam konteks *mathematical morphology* sebagai tool untuk pengekstrakan komponen citra yang berguna dalam representasi dan deskripsi bentuk daerah, seperti boundaries, skeletons, dan convex hull. Teknik morfologi juga digunakan untuk pre atau *post-processing*, seperti morfologi *filtering*, *thinning* dan *pruning*. Secara umum, pemrosesan citra secara morfologi dilakukan dengan cara mem-*passing* sebuah *structuring element* terhadap sebuah citra dengan cara yang hampir sama dengan konvolusi. *Structuring element* dapat diibaratkan dengan mask pada pemrosesan citra biasa (bukan secara morfologi).

Structuring element (Strel) dapat berukuran sembarang dan memiliki titik poros yang disebut dengan titik *origin* atau titik asal atau titik acuan. Biasanya titik acuan dari strel ditandai dengan tanda titik hitam. Jika tidak ditandai dengan titik hitam tersebut, maka diasumsikan titik acuan berada di pusat simetri seperti yang ditunjukkan pada Gambar 22.



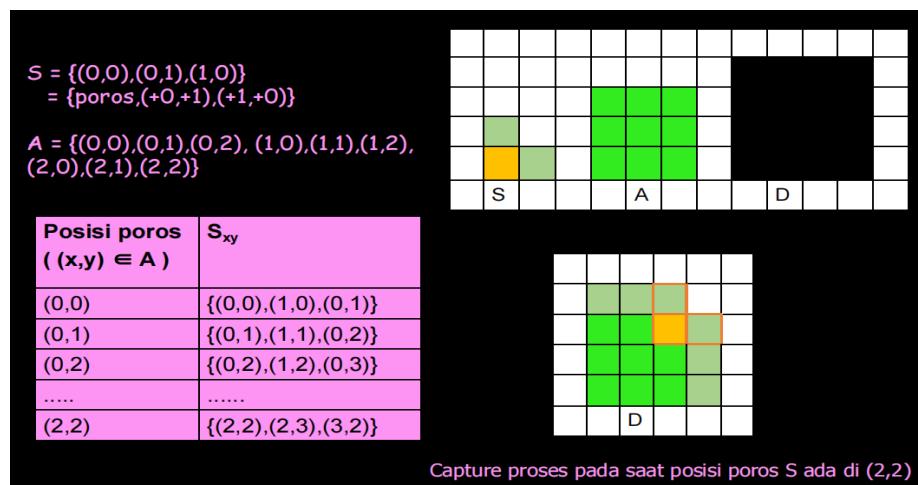
Gambar 22. Contoh variasi strel

Dilasi merupakan proses penggabungan titik-titik latar (0) menjadi bagian dari objek (1), berdasarkan structuring element S yang digunakan dengan persamaan rumus :

$$D(A, S) = A \oplus S$$

Cara dilasi adalah:

- Untuk setiap titik pada A, lakukan hal berikut:
 - a. letakkan titik poros S pada titik A tersebut
 - b. beri angka 1 untuk semua titik (x,y) yang terkena / tertimpa oleh struktur S pada posisi tersebut



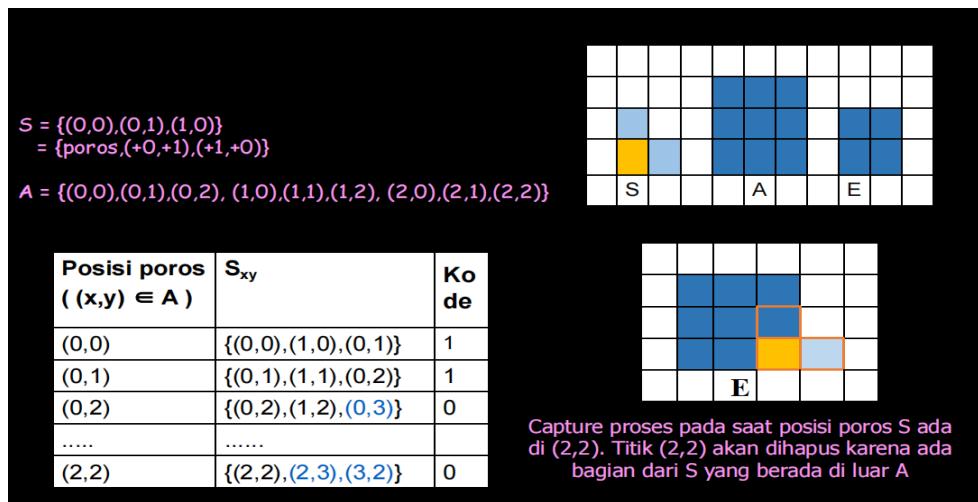
Gambar 23. Contoh proses dilatası

Erosi merupakan proses penghapusan titik-titik objek (1) menjadi bagian dari latar (0), berdasarkan structuring element S yang digunakan menggunakan persamaan rumus :

$$E(A, S) = A \otimes S$$

Cara erosi adalah:

- Untuk setiap titik pada A, lakukan hal berikut:
 - a. letakkan titik poros S pada titik A tersebut
 - b. jika ada bagian dari S yang berada di luar A, maka titik poros dihapus / dijadikan latar.



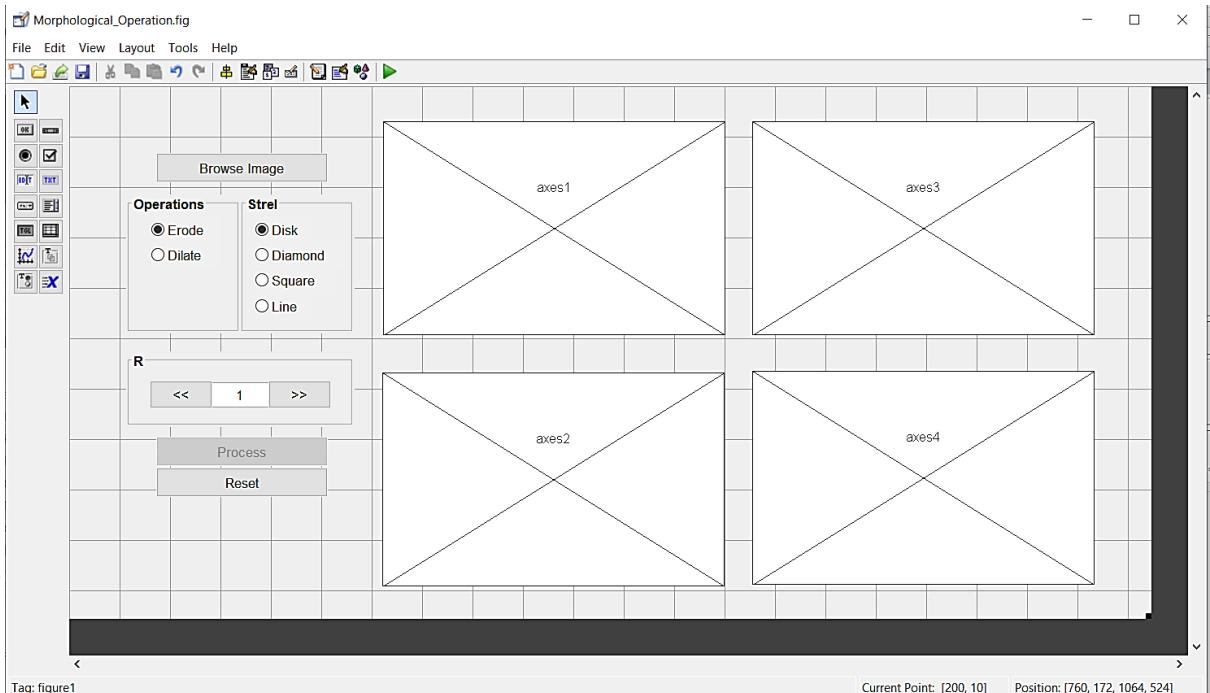
Gambar 24. Contoh proses erosi

d. Alat dan Bahan

1. Kertas A4
2. Pulpen
3. Laptop
4. Citra (image)
5. Aplikasi Matlab

e. Prosedur Kerja

1. buatlah GUI pada matlab seperti pada gambar berikut :



Gambar 25. GUI Operasi Morfologi

2. ketikkan sourcecode berikut :

```
% --- Executes on button press in radiobutton1.
function radiobutton1_Callback(hObject, eventdata, handles)
% Hint: get(hObject,'Value') returns toggle state of radiobutton1
set(handles.radiobutton1,'Value',1) %untuk erode(erosi)
set(handles.radiobutton2,'Value',0) %untuk dilate(dilatasasi)

% --- Executes on button press in radiobutton2.
function radiobutton2_Callback(hObject, eventdata, handles)
% Hint: get(hObject,'Value') returns toggle state of radiobutton2
set(handles.radiobutton1,'Value',0)
set(handles.radiobutton2,'Value',1)

% --- Executes on button press in radiobutton5.
function radiobutton5_Callback(hObject, eventdata, handles)
% Hint: get(hObject,'Value') returns toggle state of radiobutton5
set(handles.radiobutton5,'Value',1) %untuk strel disk
set(handles.radiobutton6,'Value',0) %untuk strel diamond
set(handles.radiobutton7,'Value',0) %untuk strel square
set(handles.radiobutton8,'Value',0) %untuk strel line
%untuk jika memilih salah satu strel maka value yang dipilih bernilai 1
%(pada masing-masing radiobutton 6-8)

% --- Executes on button press in pushbutton1. (untuk tombol browse)
function pushbutton1_Callback(hObject, eventdata, handles)
filename pathname] = uigetfile({'*.*'});
if ~isequal(filename,0)
    Info = imfinfo(fullfile(pathname,filename));
    if Info.BitDepth == 1
        msgbox('Citra masukan harus citra RGB atau Grayscale');
        return
    elseif Info.BitDepth == 8
        Img_Gray = imread(fullfile(pathname,filename));
        axes(handles.axes1)
        cla('reset')
        imshow(Img_Gray)
        title('Grayscale Image')
        Img_bw = im2bw(Img_Gray,graythresh(Img_Gray));
        axes(handles.axes2)
        cla('reset')
        imshow(Img_bw)
        title('Binary Image')
    else
        Img_Gray = rgb2gray(imread(fullfile(pathname,filename)));
        axes(handles.axes1)
        cla('reset')
        imshow(Img_Gray)
        title('Grayscale Image')
        Img_bw = im2bw(Img_Gray,graythresh(Img_Gray));
        axes(handles.axes2)
        cla('reset')
        imshow(Img_bw)
        title('Binary Image')
    end
else
    return
end
handles.Img_Gray = Img_Gray;
handles.Img_Gray2 = Img_Gray;
handles.Img_bw = Img_bw;
handles.Img_bw2 = Img_bw;
guidata(hObject,handles);
set(handles.pushbutton2,'Enable','on')
```

```

% --- Executes on button press in pushbutton2.(untuk tombol process)
function pushbutton2_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton2 (see GCBO)
% eventdata   reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
Img_Gray = handles.Img_Gray;
Img_bw = handles.Img_bw;

% Operations
val1 = get(handles.radioButton1,'Value');
val2 = get(handles.radioButton2,'Value');

% Structure Element
val5 = get(handles.radioButton5,'Value');
val6 = get(handles.radioButton6,'Value');
val7 = get(handles.radioButton7,'Value');
val8 = get(handles.radioButton8,'Value');

% R
R = str2double(get(handles.edit1,'String'));

%
if val5 == 1
    se = strel('disk',R);
elseif val6 == 1
    se = strel('diamond',R);
elseif val7 == 1
    se = strel('square',R);
elseif val8 == 1
    se = strel('line',R,45);
end

if val1 == 1
    Morph_Gray =imerode(Img_Gray,se);
    Morph_bw = imerode(Img_bw,se);
elseif val2 == 1
    Morph_Gray = imdilate(Img_Gray,se);
    Morph_bw = imdilate(Img_bw,se);
end

axes(handles.axes1)
cla('reset')
imshow(Img_Gray)
title('Grayscale Image')
axes(handles.axes2)
cla('reset')
imshow(Img_bw)
title('Binary Image')
axes(handles.axes3)
cla('reset')
imshow(Morph_Gray)
title('Morphological Operation')
axes(handles.axes4)
cla('reset')
imshow(Morph_bw)
title('Morphological Operation')

```

```

% --- Executes on button press in pushbutton3.(Untuk tombol reset)
function pushbutton3_Callback(hObject, eventdata, handles)
axes(handles.axes1)
cla('reset')
set(gca,'XTick',[])
set(gca,'YTick',[])

axes(handles.axes2)
cla('reset')
set(gca,'XTick',[])
set(gca,'YTick',[])

axes(handles.axes3)
cla('reset')
set(gca,'XTick',[])
set(gca,'YTick',[])

axes(handles.axes4)
cla('reset')
set(gca,'XTick',[])
set(gca,'YTick',[])

set(handles.pushbutton2,'Enable','off')
set(handles.edit1,'String',1)
set(handles.radioButton1,'Value',1)
set(handles.radioButton2,'Value',0)
set(handles.radioButton5,'Value',1)
set(handles.radioButton6,'Value',0)
set(handles.radioButton7,'Value',0)
set(handles.radioButton8,'Value',0)

% --- Executes on button press in pushbutton4.(untuk tombol >>)
function pushbutton4_Callback(hObject, eventdata, handles)
val = str2double(get(handles.edit1,'String'))+1;
if val > 10
    val = 10;
end
set(handles.edit1,'String',val)

% --- Executes on button press in pushbutton5. (untuk tombol <<)
function pushbutton5_Callback(hObject, eventdata, handles)
val = str2double(get(handles.edit1,'String'))-1;
if val < 1
    val = 1;
end

```

f. Hasil dan Pembahasan

1. Dokumentasi Tugas berupa laporan (definisi, proses dan penerapan operasi morfologi closing dan opening)
2. Dokumentasi Tugas berupa file ppt

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 19

Pokok Bahasan	:Operasi Morfologi (lanjutan)
Acara Praktikum/Praktik	:Minggu 5/3
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	:100 menit

- a. Capaian Pembelajaran Mata Kuliah (CPMK)
 - 1. Mahasiswa mampu memahami definisi dari pengolahan citra dan vision.
 - 2. Mahasiswa mampu memahami proses operasi morfologi.
 - 3. Mahasiswa mampu memberikan contoh penerapan operasi morfologi.

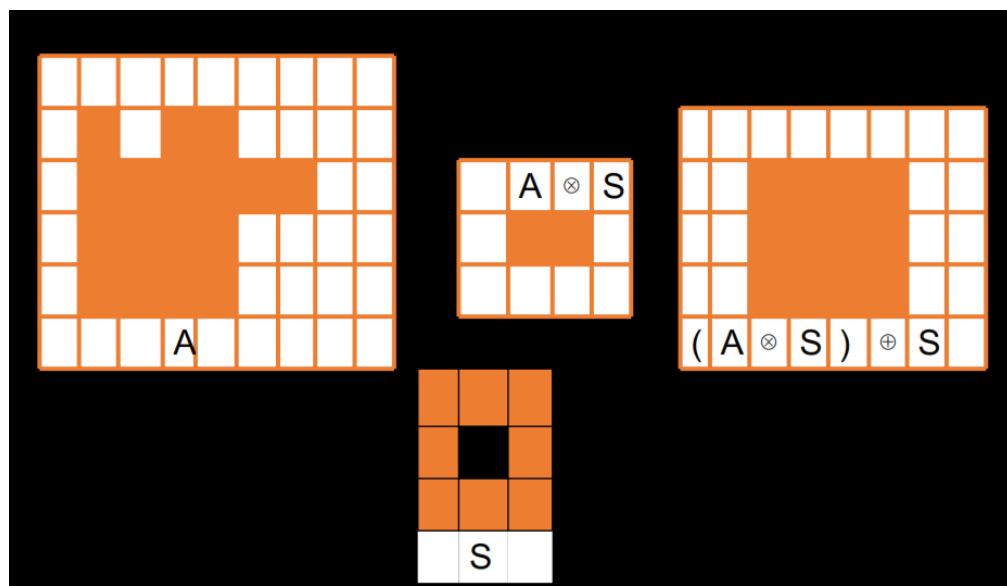
- b. Indikator Penilaian

Ketepatan dalam menjelaskan mengenai operasi morfologi

- c. Dasar Teori

Opening adalah proses erosi yang diikuti dengan dilasi. Efek yang dihasilkan adalah menghilangnya objek-objek kecil dan kurus, memecah objek pada titik-titik yang kurus, dan secara umum men-smooth-kan batas dari objek besar tanpa mengubah area objek secara signifikan menggunakan persamaan rumus :

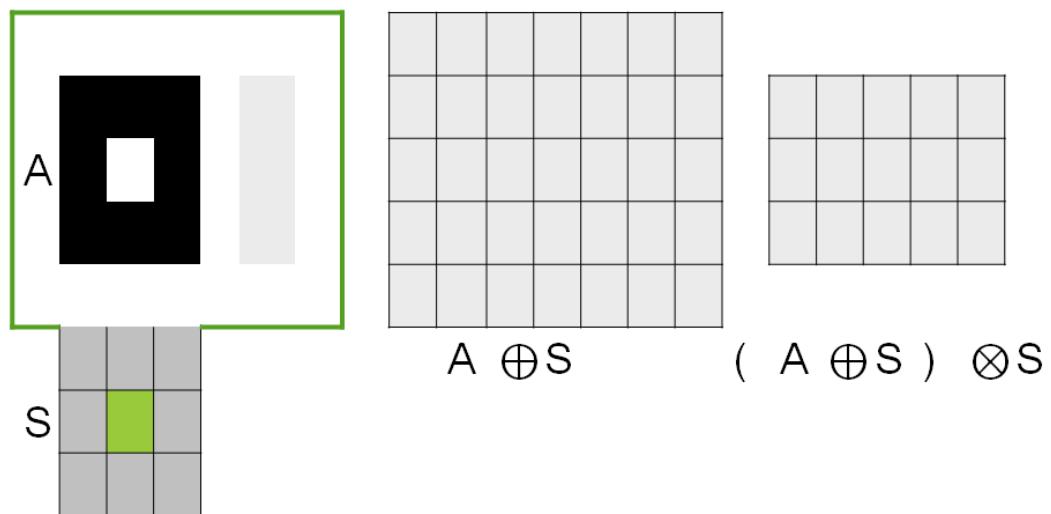
$$A \cdot S = (A \otimes S) \oplus S$$



Gambar 26. Contoh proses opening

Closing adalah proses dilasi yang diikuti dengan erosi. Efek yang dihasilkan adalah mengisi lubang kecil pada objek, menggabungkan objek-objek yang berdekatan, dan secara umum mensmooth-kan batas dari objek besar tanpa mengubah area objek secara signifikan menggunakan persamaan rumus :

$$A \cdot S = (A \oplus S) \otimes S$$

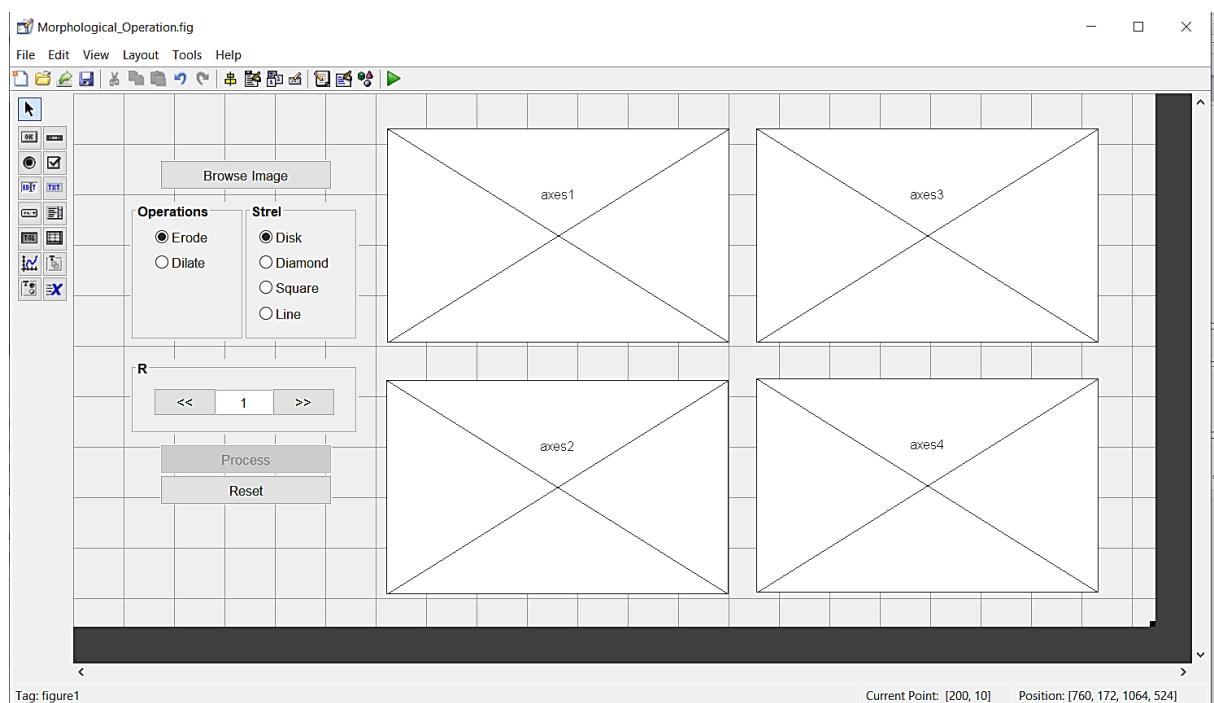


Gambar 27. Contoh proses closing

- d. Alat dan Bahan
1. Kertas A4
 2. Pulpen
 3. Laptop
 4. Citra (image)
 5. Aplikasi Matlab

e. Prosedur Kerja

1. buatlah GUI pada matlab seperti pada gambar berikut :



Gambar 28. GUI Operasi Morfologi

2. buatlah sourcecode seperti pada pertemuan ke 18, dengan menambahkan radiobutton3 untuk operasi opening dan radiobutton4 untuk operasi closing
3. Lakukan Analisa pada operasi morfologi

f. Hasil dan Pembahasan

1. Dokumentasi Tugas berupa laporan (definisi, proses dan penerapan operasi morfologi closing dan opening)
2. Dokumentasi Tugas berupa file ppt

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 20

Pokok Bahasan	:Operasi Morfologi (lanjutan)
Acara Praktikum/Praktik	:Minggu 5/4
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	:100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami definisi dari pengolahan citra dan vision.
2. Mahasiswa mampu memahami proses operasi morfologi.
3. Mahasiswa mampu memberikan contoh penerapan operasi morfologi.

b. Indikator Penilaian

Ketepatan dalam menjelaskan mengenai operasi morfologi.

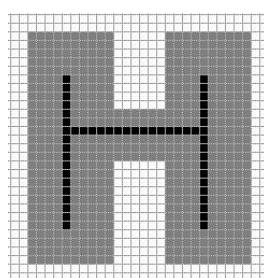
c. Dasar Teori

Erosi yang dimodifikasi sehingga tidak boleh ada objek yang terpecah. Hasilnya adalah berupa garis yang menunjukkan topologi objek semula. Tujuan: me-remove piksel tertentu pada objek sehingga tebal objek tersebut menjadi hanya satu piksel.

Thinning tidak boleh:

1. Menghilangkan end-point
2. Memutus koneksi yang ada
3. Mengakibatkan excessive erosi

Salah satu kegunaan thinning adalah pada proses seperti *optical character recognition*, *fingerprint recognition* dan *document processing*. Ada banyak cara mengimplementasikan thinning, salah satu diantaranya adalah dengan hit-or-miss transform.



Gambar 29. Salah satu contoh thinning

Thinning dapat didefinisikan sebagai:

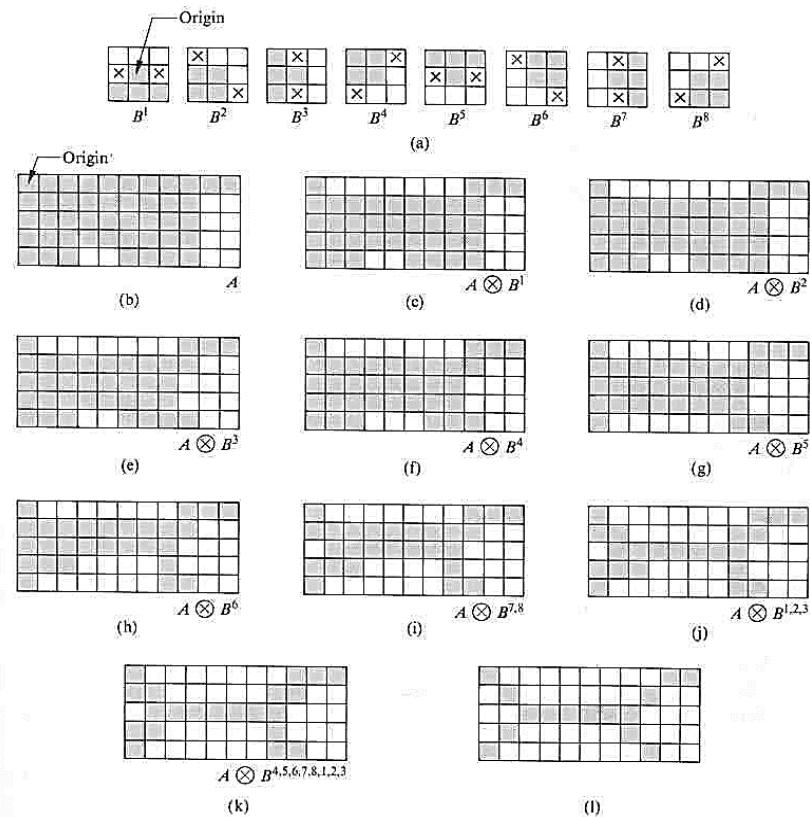
$$\begin{aligned}\text{Thinning}(A, \{B\}) &= A - (A * \{B\}) \\ &= A - ((\dots(A * B_1) * B_2) .. B_n)\end{aligned}$$

dengan $B_1, B_2, B_3..B_n$ adalah Structuring element.

Note:

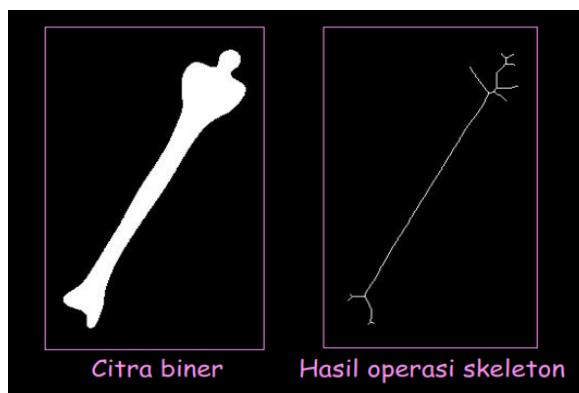
$A - (A * B)$ berarti kebalikan dari $A * B$

Yang match dihapus, sedangkan yang tidak match dipertahankan



Gambar 30. Variasi contoh thinning

Skeletonization adalah cara lain untuk mengurangi obyek citra biner menjadi himpunan kerangka tipis yang menahan informasi penting mengenai bentuk asli obyek.



Gambar 31. Contoh skeletonization

- d. Alat dan Bahan
 - 1. Kertas A4
 - 2. Pulpen
 - 3. Laptop
 - 4. Citra (image)
 - 5. Aplikasi Matlab

e. Prosedur Kerja

1. Gambar yang digunakan :



Gambar 32. Contoh huruf “A”

2. Buatlah operasi morfologi thinning menggunakan sourcecode berikut :

```
huruf=imread('E:\1.Vani\BKPM PCV\huruf.jpg');
gray=rgb2gray(huruf);
[M,N]=size(gray);
H_seg = zeros(M,N);
for k = 1 : M
    for l = 1 : N
        if gray(k,l)<20
            H_seg(k,l)=1;
        else
            H_seg(k,l)=0;
        end
    end
end
%BW2 = bwmorph(BW,operation,n)
thinning=bwmorph(H_seg,'thin',25);
```

3. Buatlah operasi morfologi skeletoning menggunakan sourcecode berikut :

```
%BW2 = bwmorph(BW,operation,n)
thinning=bwmorph(H_seg,'thin',25);
skeletonizing=bwmorph(H_seg,'skel',70);
subplot(1,4,1),imshow(gray);
subplot(1,4,2),imshow(H_seg);
subplot(1,4,3),imshow(thinning);
subplot(1,4,4),imshow(skeletonizing);
```

4. Amati dan lakukan Analisa dari kedua morfologi tersebut !

f. Hasil dan Pembahasan

1. Dokumentasi Tugas berupa laporan (definisi, proses dan penerapan operasi morfologi closing dan opening)
2. Dokumentasi Tugas berupa file ppt

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 21

Pokok Bahasan	:Ekstraksi Fitur Warna
Acara Praktikum/Praktik	:Minggu 6/1
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	:100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami definisi dari pengolahan citra dan vision.
2. Mahasiswa mampu memahami proses Ekstraksi Fitur.
3. Mahasiswa mampu memberikan contoh penerapan Ekstraksi Fitur.

b. Indikator Penilaian

Ketepatan dalam menjelaskan mengenai teknik Ekstraksi Fitur

c. Dasar Teori

Fitur merupakan karakteristik yang unik dari suatu obyek. Syarat yang harus dipenuhi dalam menentukan karakteristik fitur yang baik :

1. Dapat membedakan suatu obyek dengan yang lainnya (discrimination)
2. Memperhatikan kompleksitas komputasi dalam memperoleh fitur
3. Tidak terikat (independence), bersifat invariant terhadap berbagai transformasi (rotasi, penskalaan, pergeseran, dll).
4. Jumlahnya sedikit karena fitur yang jumlahnya sedikit akan menghemat waktu komputasi dan ruang penyimpanan untuk proses selanjutnya (proses pembuatan fitur).

Ekstraksi fitur dibagi menjadi 3 yaitu fitur warna, morfologi (bentuk) dan tekstur. Pada pertemuan ini, kita akan membahas ekstraksi fitur warna. Umumnya metode ini menggunakan histogram warna dikarenakan histogram warna digunakan untuk merepresentasikan ciri warna suatu citra. Ekstraksi fitur warna membutuhkan ruang warna seperti RGB, HSV, CMY, CIELab, dll.

d. Alat dan Bahan

1. Kertas A4
2. Pulpen
3. Laptop
4. Citra (image)
5. Aplikasi Matlab

e. Prosedur Kerja

1. Gambar yang digunakan :

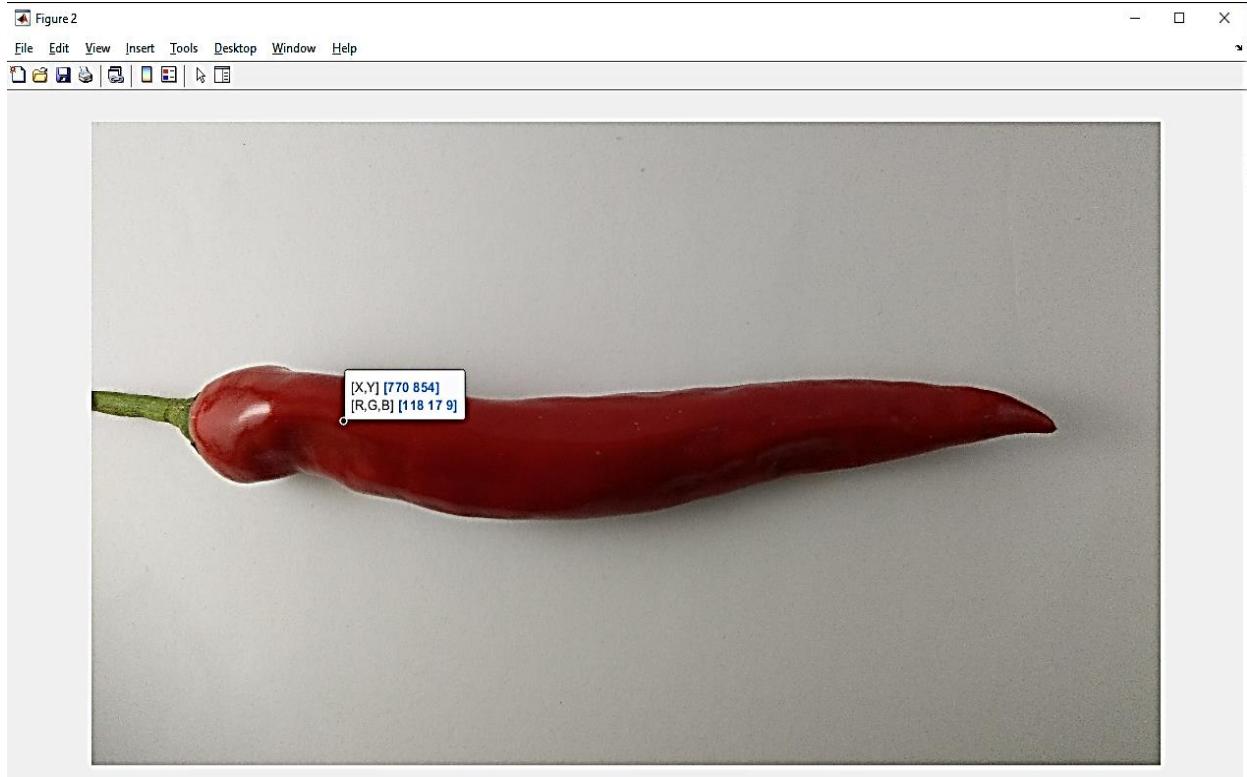


Gambar 33. Contoh citra yang digunakan

2. untuk mengetahui nilai RGB pada setiap piksel citra, sourcecode berikut :

```
gambar = imread(imread('E:\1.Vani\BKPM PCV\190.jpg'));
figure, imshow(gambar);
```

3. kemudian buka hasil citra dan klik icon  data tips kemudian sorot ke piksel yang diinginkan seperti pada Gambar 34.

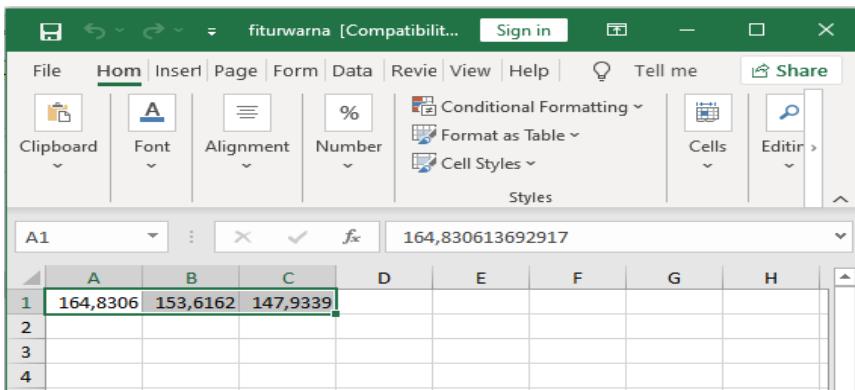


Gambar 34. Data nilai RGB pada piksel tertentu

4. untuk mendapatkan nilai fitur rata-rata dari citra gunakan sourcecode berikut:

```
gambar = imread(imread('E:\1.Vani\BKPM PCV\190.jpg'));
red = mean(mean(gambar(:,:,1)));
green = mean(mean(gambar(:,:,2)));
blue = mean(mean(gambar(:,:,3)));
fitur = [red green blue]; % mematrikkan fitur
xlswrite('fiturwarna.xls',fitur); % menyimpan fitur ke file excel
```

5. Melakukan pengecekan fitur pada data excel



6. Lakukan konversi warna ke HSV, Grayscale, CMY dan CIELab, kemudian ekstraksi fitur warnanya.
- f. Hasil dan Pembahasan
1. Dokumentasi Tugas berupa laporan (definisi, proses dan penerapan ekstraksi warna)
 2. Dokumentasi Tugas berupa file ppt
- g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 22

Pokok Bahasan	:Ekstraksi Fitur Bentuk
Acara Praktikum/Praktik	:Minggu 6/2
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	:100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami definisi dari pengolahan citra dan vision.
2. Mahasiswa mampu memahami proses Ekstraksi Fitur.
3. Mahasiswa mampu memberikan contoh penerapan Ekstraksi Fitur.

b. Indikator Penilaian

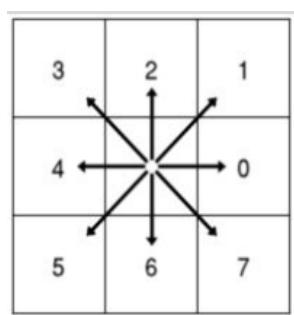
Ketepatan dalam menjelaskan mengenai teknik Ekstraksi Fitur

c. Dasar Teori

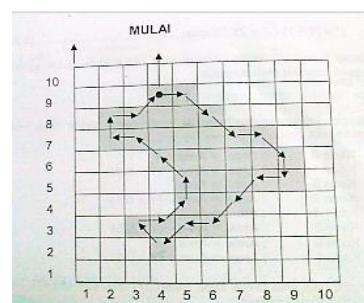
Fitur morfologi (bentuk) adalah fitur yang merepresentasikan bentuk obyek. Sebelum dilakukan proses tersebut dilakukan proses analisis bentuk terlebih dahulu, seperti :

1. Kode Rantai (Chain Code)
2. Atribut Topologi
3. Atribut Geometri

Kode Rantai (Chain Code) sering digunakan untuk mendeskripsikan/ mengkodekan bentuk (contour) suatu obyek. Pembentukan kode rantai dimulai dengan menentukan piksel pertama dari obyek. Berdasarkan piksel tersebut, kode rantai dibentuk dengan cara mengikuti aturan arah kode rantai. Berdasarkan kode rantai tersebut digunakan menghitung area, keliling dan faktor bentuk.



(a)



(b)

Gambar 35. (a) arah kode rantai dan (b) contoh kode rantai beserta obyeknya

Area merupakan jumlah piksel-piksel penyusun obyek dan satunya piksel sehingga membentuk luasan. Area juga mencerminkan ukuran atau berat obyek.

Perimeter merupakan panjang perbatasan obyek tersebut dengan menghitung jumlah piksel dari batas obyek. Untuk mencari faktor bentuk digunakan persamaan rumus :

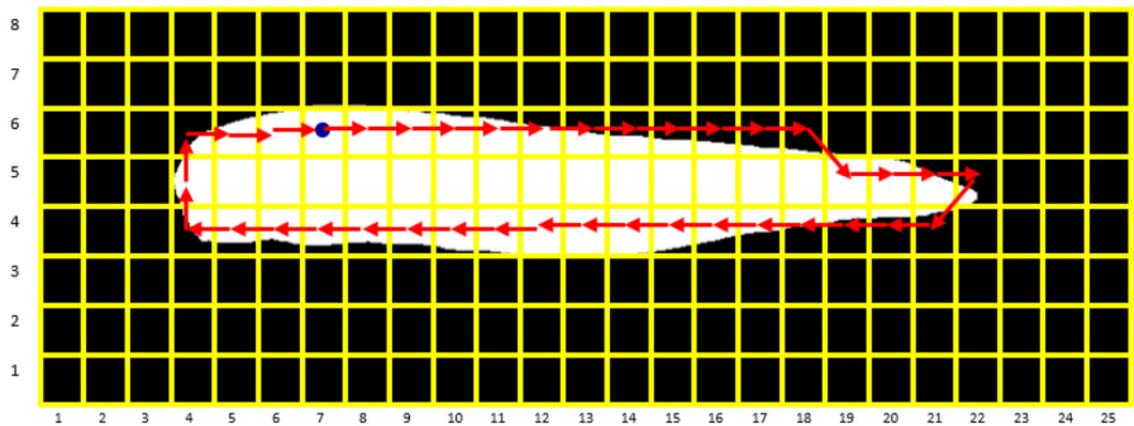
$$S = \frac{Perimeter^2}{Area}$$

d. Alat dan Bahan

1. Kertas A4
2. Pulpen
3. Laptop
4. Citra (image)
5. Aplikasi Matlab

e. Prosedur Kerja

1. Menghitung Luas, Perimeter dan Faktor bentuk berdasarkan obyek berikut :



Gambar 36. Contoh citra yang digunakan

2. Tentukan kode rantai dari gambar tersebut !
3. Perhitungan luas area dapat dijabarkan sebagai berikut:

Kode 0 : Area = Area + Y

Kode 1 : Area = Area + (Y + 0,5)

Kode 2 : Area = Area

Kode 3 : Area = Area - (Y + 0,5)

Kode 4 : Area = Area - Y

Kode 5 : Area = Area - (Y - 0,5)

Kode 6 : Area = Area

Kode 7 : Area = Area + (Y - 0,5)

4. Lengkapi Tabel berikut kemudian jumlahkan area tersebut untuk mendapatkan luas area obyek.

Kode rantai																								
Ordinat (y)																								
Area																								

4. Lakukan perhitungan perimeter menggunakan rumus :

$$\text{perimeter} = \text{jumlah kode genap} + \text{jumlah kode ganjil} \sqrt{2}$$

5. Hitung faktor bentuk menggunakan persamaan rumus :

$$S = \frac{\text{Perimeter}^2}{\text{Area}}$$

f. Hasil dan Pembahasan

1. Dokumentasi Tugas berupa laporan (definisi, proses dan penerapan ekstraksi warna)

2. Dokumentasi Tugas berupa file ppt

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 23

Pokok Bahasan	:Ekstraksi Fitur Bentuk (lanjutan)
Acara Praktikum/Praktik	:Minggu 6/3
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	:100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami definisi dari pengolahan citra dan vision.
2. Mahasiswa mampu memahami proses Ekstraksi Fitur.
3. Mahasiswa mampu memberikan contoh penerapan Ekstraksi Fitur.

b. Indikator Penilaian

Ketepatan dalam menjelaskan mengenai teknik Ekstraksi Fitur

c. Dasar Teori

Atribut Geometri bertujuan untuk mencari nilai C (kebundaran/compactness/circularity). atribut kebundaran ini juga disebut rasio ketipisan yang didapatkan menggunakan persamaan rumus :

$$C = \frac{4\pi A}{P^2}$$

Untuk citra-citra yang memiliki obyek-obyek tipis seperti yang diketik (*typewritten*) atau karakter-karakter pada naskah maka rata-rata dari panjang dan lebar obyek dapat dinyatakan menggunakan persamaan rumus :

$$L_A = \frac{P_A}{2} \quad W_A = \frac{2*A_A}{P_A}$$

d. Alat dan Bahan

1. Kertas A4
2. Pulpen
3. Laptop
4. Citra RGB
5. Aplikasi Matlab

e. Prosedur Kerja

1. Masukkan citra yang digunakan:



(a)



(b)



(c)



(d)



(e)

Gambar 37. Varietas mangga (a) apel, (b) gedong gincu, (c) gadung, (d) golek dan (e) manalagi

2. Masukkan sourcecode dari proses preprocessing hingga segmentasi seperti berikut :

```
gambar = imread('E:\1.Vani\BKPM PCV\training_162.jpg');
blue = gambar(:,:,3);
[A, R] = size(blue);
Seg = zeros(A,R);
%dimulai perulangan sebanyak ukuran matrix A,R
for k = 1 : A
    for l = 1 : R
        if blue(k,l) < 160
            Seg(k,l)=1;
        else
            Seg(k,l)=0;
        end
    end
end
```

3. kemudian lakukan proses kode rantai menggunakan sourcecode berikut :

```
AA=bwlabeln(Seg,8);%pelabelan menggunakan 8 ketetanggaan
a=max(max(AA));%jumlah label
S=regionprops(AA,'Area');%ROI berdasarkan area
I_seg=ismember(AA, find([S.Area]>=100));
```

4. sourcecode mencari area, perimeter, factor bentuk dan circularity

```
%-----
% Mendapatkan Fitur Morfologi
%-----
for c=1:a
    area=0;
    perimeter=0;
    for i=1:151
        for j=1:151
            if AA(i,j)==c
                area=area+1;
            end
        end
    end

    for i=2:150
        for j=2:150
            if AA(i,j)==c
                if AA(i,j) ~= AA(i-1,j) || AA(i,j) ~= AA(i+1,j) || AA(i,j) ~= AA(i,j-1)
                || AA(i,j) ~= AA(i,j+1)
                    perimeter=perimeter+1;
                end
            end
        end
    end

    luas(c)=area;
    keliling(c)=perimeter;
end
```

```

hitung=0;
for c=1:a
    if luas(c)>=100
        hitung=hitung+1;
        luas_area(hitung)=luas(c);
        keliling_area(hitung)=keliling(c);
    end
end

jum_luas=0;
jum_keliling=0;
for i=1:hitung
    jum_luas=jum_luas + luas_area(i);
    jum_keliling=jum_keliling + keliling_area(i);
end
phi=3.14;
area_mangga=jum_luas;
perimeter_mangga=jum_keliling;
bentuk_mangga=perimeter_mangga^2/area_mangga;
circularity=4*phi*area_mangga/perimeter_mangga^2;

```

5. buatlah fitur tersebut kedalam matriks menggunakan sourcecode berikut :

```

% -----
%      Mematrikskan Fitur dan menyimpan di excel
%
fitur=[area_mangga perimeter_mangga bentuk_mangga circularity];
xlswrite('fiturbentuk.xls',fitur);

```

6. bandingkan menggunakan sourcecode berikut :

```

AA=bwlabeln(Seg,8);%pelabelan menggunakan 8 ketetanggaan
a=max(max(AA));%jumlah label
S=regionprops(AA,'Area','Perimeter');%ROI berdasarkan area dan perimeter
area_mangga=S.Area;
perimeter_mangga=S.Perimeter;
phi=3.14;
bentuk_mangga=perimeter_mangga^2/area_mangga;
circularity=4*phi*area_mangga/perimeter_mangga^2;

```

f. Hasil dan Pembahasan

1. Dokumentasi Tugas berupa laporan (definisi, proses dan penerapan ekstraksi warna)
2. Dokumentasi Tugas berupa file ppt

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 24

Pokok Bahasan	:Ekstraksi Fitur Tekstur
Acara Praktikum/Praktik	:Minggu 6/4
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	:100 menit

- a. Capaian Pembelajaran Mata Kuliah (CPMK)
 - 1. Mahasiswa mampu memahami definisi dari pengolahan citra dan vision.
 - 2. Mahasiswa mampu memahami proses Ekstraksi Fitur.
 - 3. Mahasiswa mampu memberikan contoh penerapan Ekstraksi Fitur.

- b. Indikator Penilaian

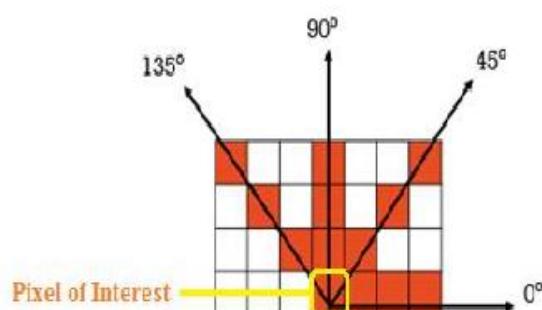
Ketepatan dalam menjelaskan mengenai teknik Ekstraksi Fitur

- c. Dasar Teori

Tekstur merupakan keteraturan pola-pola tertentu yang terbentuk dari susunan piksel-piksel dalam citra digital. Keteraturan pola-pola tersebut muncul secara berulang-ulang dengan interval jarak dan arah tertentu. Ada dua syarat terbentuknya tekstur yaitu :

- 1. Adanya pola-pola primitif (berupa titik, garis lurus, garis lengkung, luasan dan lain-lain yang mendeskripsikan bentuk) yang terdiri dari satu atau lebih piksel.
- 2. Pola-pola primitif tersebut muncul secara berulang-ulang dengan jarak dan arah tertentu sehingga dapat ditemukan atau diprediksi karakteristik perulangannya.

Salah satu metode untuk memperoleh fitur tekstur dengan cara metode statis. Metode statis menggunakan perhitungan statistika untuk membentuk fitur. Salah satu contoh metode statis adalah Gray Level Co-occurrence Matrix (GLCM). GLCM merupakan sebuah pendekatan untuk ekstraksi informasi tekstur yang berhubungan dengan transisi level keabuan antara dua piksel menggunakan sebuah matriks cooccurrence. Matriks tersebut merepresentasikan frekuensi munculnya pasangan dua piksel dengan intensitas tertentu dalam jarak d dan arah sudut θ tertentu. Oleh karena itu, matrix memberikan informasi berbeda yang diperoleh dari perbedaan jarak antar piksel(Roberti de Siqueira et al., 2013).

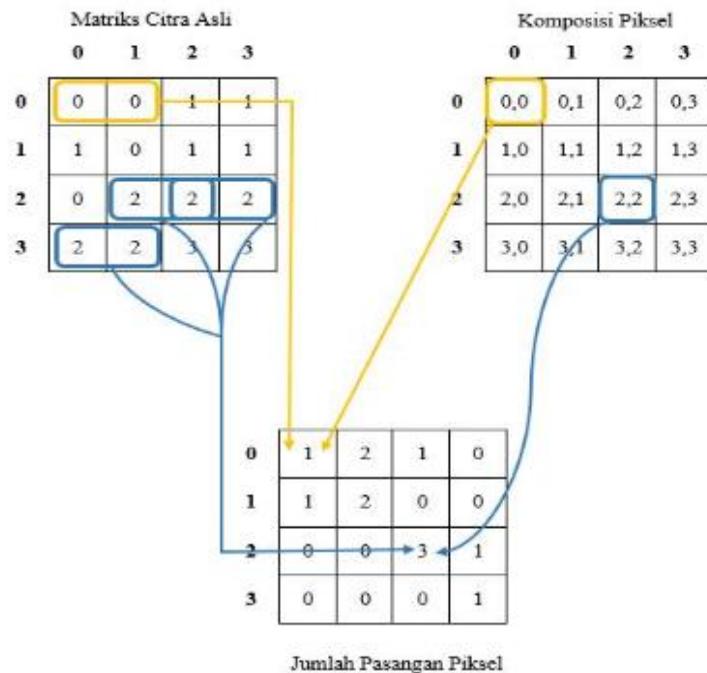


Gambar 38. Ilustrasi arah sudut GLCM

Matriks co-occurrence $p(i_1, i_2)$ dihitung berdasarkan dua langkah sederhana sebagai berikut :

1. Menentukan jarak antara dua titik dalam arah vertikal dan horizontal (vektor $d = (dx, dy)$) dimana dx dan dy dinyatakan dalam piksel.
2. Menghitung pasangan piksel yang mempunyai nilai intensitas i_1 dan i_2 dan berjarak d piksel dalam citra. Kemudian hasil perhitungan setiap pasangan nilai intensitas diletakkan pada matriks $p(i_1, i_2)$ dimana nilai intensitas i_1 sebagai absis dan nilai intensitas i_2 sebagai ordinat.

Contoh Perhitungan dengan Arah 0°



Gambar 39. Penentuan awal matriks GLCM berdasarkan dua piksel

3. Matriks yang terbentuk pada Gambar 40. dinamakan matrix framework. Matriks tersebut perlu diolah agar menjadi matriks yang simetris dengan menambahkan matriks transposnya.
4. Untuk menghilangkan ketergantungan pada ukuran citra, maka nilai-nilai elemen pada matriks GLCM perlu dinormalisasi sehingga bernilai 1. Proses tersebut ditunjukkan seperti pada Gambar 41.

$$\begin{bmatrix} 1 & 2 & 1 & 0 \\ 1 & 2 & 0 & 0 \\ 0 & 0 & 3 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} + \begin{bmatrix} 1 & 1 & 0 & 0 \\ 2 & 2 & 0 & 0 \\ 1 & 0 & 3 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 2 & 3 & 1 & 0 \\ 3 & 4 & 0 & 0 \\ 1 & 0 & 6 & 1 \\ 0 & 0 & 1 & 2 \end{bmatrix}$$

Transpos

GLCM sebelum dinormalisasai

$$\begin{bmatrix} \frac{2}{24} & \frac{3}{24} & \frac{1}{24} & \frac{0}{24} \\ \frac{3}{24} & \frac{4}{24} & \frac{0}{24} & \frac{0}{24} \\ \frac{24}{24} & \frac{24}{24} & \frac{24}{24} & \frac{24}{24} \\ \frac{1}{24} & \frac{0}{24} & \frac{6}{24} & \frac{1}{24} \\ \frac{24}{24} & \frac{24}{24} & \frac{24}{24} & \frac{24}{24} \\ \frac{0}{24} & \frac{0}{24} & \frac{1}{24} & \frac{2}{24} \\ \frac{24}{24} & \frac{24}{24} & \frac{24}{24} & \frac{24}{24} \end{bmatrix}$$

Gambar 40. Proses pembentukan matriks GLCM yang simetris

Setelah mendapatkan matriks GLCM yang simetris, langkah selanjutnya yaitu mendapatkan beberapa fitur GLCM yang sebagai berikut :

$$ASM = \sum_{i=1}^L \sum_{j=1}^L (GLCM(i,j))^2,$$

$$Contrast = \sum_{i=1}^L \sum_{j=1}^L |i-j|^2 GLCM(i,j),$$

$$IDM = \sum_{i=1}^L \sum_{j=1}^L \frac{(GLCM(i,j))^2}{1+(i-j)^2},$$

$$Entropy = \sum_{i=1}^L \sum_{j=1}^L (GLCM(i,j)) \log(GLCM(i,j)),$$

$$Correlation = \sum_{i=1}^L \sum_{j=1}^L \frac{(i-\mu i)(j-\mu j)(GLCM(i,j))}{\sigma i \sigma j}.$$

d. Alat dan Bahan

1. Kertas A4
2. Pulpen
3. Laptop
4. Citra RGB
5. Aplikasi Matlab

e. Prosedur Kerja

1. Masukkan citra yang digunakan:



Buah Naga Hijau



Buah Naga Hijau_merah



Buah Naga Kuning_merah



Buah Naga Merah

Gambar 41. Contoh citra yang digunakan

2. Masukkan sourcecode seperti berikut :

```

citra=imread('E:\1.Vani\BKPM PCV\testing_36.jpg');
I=rgb2gray(citra);
[G0, G45, G90, G135] = glcm (I);
[tinggi, lebar] = size (I);
%-----
% Step 2 : Pembentukan GLCM dan Fiturnya
%-----
% Bentuk GLCM
GLCM0 = zeros(256, 256);
total_piks0 =0;
GLCM45 = zeros(256, 256);
total_piks145 =0;
GLCM90 = zeros(256, 256);
total_piks190 =0;

GLCM135 = zeros(256, 256);
total_piks135 =0;
for y=2: tinggi-1
    for x=2: lebar-1
        %untuk sudut 0
        a = I(y,x);
        b = I(y,x+1);
        GLCM0(a+1, b+1) = GLCM0(a+1, b+1) + 1;
        total_piks0 = total_piks0 + 1;

        %untuk sudut 45
        a = I(y,x);
        b = I(y-1,x+1);
        GLCM45(a+1, b+1) = GLCM45(a+1, b+1) + 1;
        total_piks145 = total_piks145 + 1;

        %untuk sudut 90
        a = I(y,x);
        b = I(y-1,x);
        GLCM90(a+1, b+1) = GLCM90(a+1, b+1) + 1;
        total_piks190 = total_piks190 + 1;

        %untuk sudut 135
        a = I(y,x);
        b = I(y-1,x-1);
        GLCM135(a+1, b+1) = GLCM135(a+1, b+1) + 1;
        total_piks135 = total_piks135 + 1;
    end
end
GLCM0 = GLCM0 / total_piks0;
GLCM45 = GLCM45 / total_piks145;
GLCM90 = GLCM90 / total_piks190;
GLCM135 = GLCM135 / total_piks135;

```

3. kemudian menghitung ASM menggunakan sourcecode berikut :

```
%Hitung ASM
asm0 = 0.0;
asm45 = 0.0;
asm90 = 0.0;
asm135 = 0.0;

for a=0 : 255
    for b=0 : 255
        asm0 = asm0 + (GLCM0(a+1,b+1) * GLCM0(a+1,b+1));
        asm45 = asm45 + (GLCM45(a+1,b+1) * GLCM45(a+1,b+1));
        asm90 = asm90 + (GLCM90(a+1,b+1) * GLCM90(a+1,b+1));
        asm135 = asm135 + (GLCM135(a+1,b+1) * GLCM135(a+1,b+1));
    end
end
```

4. menghitung kontras menggunakan sourcecode berikut :

```
%Hitung kontras
kontras0 = 0.0;
kontras45 = 0.0;
kontras90 = 0.0;
kontras135 = 0.0;
%
for a=0 : 255
    for b=0 : 255
        kontras0 = kontras0 + (a-b)*(a-b) * GLCM0(a+1,b+1);
        kontras45 = kontras45 + (a-b)*(a-b) * GLCM45(a+1,b+1);
        kontras90 = kontras90 + (a-b)*(a-b) * GLCM90(a+1,b+1);
        kontras135 = kontras135 + (a-b)*(a-b) * GLCM135(a+1,b+1);
    end
end
```

5. menghitung IDM menggunakan sourcecode berikut :

```
%Hitung IDM
idm0 = 0.0;
idm45 = 0.0;
idm90 = 0.0;
idm135 = 0.0;

for a=0 : 255
    for b=0 : 255
        idm0 = idm0 + (GLCM0(a+1,b+1) / (1+(a-b)*(a-b)));
        idm45 = idm45 + (GLCM45(a+1,b+1) / (1+(a-b)*(a-b)));
        idm90 = idm90 + (GLCM90(a+1,b+1) / (1+(a-b)*(a-b)));
        idm135 = idm135 + (GLCM135(a+1,b+1) / (1+(a-b)*(a-b)));
    end
end
```

6. menghitung entropi menggunakan sourcecode berikut :

```
%Hitung entropi
entropi0 = 0.0;
entropi45 = 0.0;
entropi90 = 0.0;
entropi135 = 0.0;

for a=0 : 255
    for b=0 : 255
        if (GLCM0 (a+1, b+1) ~= 0)
            entropi0 = entropi0 - (GLCM0(a+1, b+1)*(log(GLCM0(a+1,b+1)))); 
        end
        if (GLCM45 (a+1, b+1) ~= 0)
            entropi45 = entropi45 - (GLCM45(a+1, b+1)*(log(GLCM45(a+1,b+1)))); 
        end
        if (GLCM90 (a+1, b+1) ~= 0)
            entropi90 = entropi90 - (GLCM90(a+1, b+1)*(log(GLCM90(a+1,b+1)))); 
        end
        if (GLCM135 (a+1, b+1) ~= 0)
            entropi135 = entropi135 - (GLCM135(a+1, b+1)*(log(GLCM135(a+1,b+1)))); 
        end
    end
end
```

7. menghitung korelasi terlebih dahulu menghitung kovarian dan standar deviasi menggunakan sourcecode berikut :

```
%Hitung Kovarians
%Hitung px [] dan py [] dulu
korelasi0 =0.0;
px0 = 0;
py0 = 0;
reratax0 = 0.0;
reratay0 = 0.0;
stdevx0 = 0.0;
stdevy0 = 0.0;

korelasi45 =0.0;
px45 = 0;
py45 = 0;
reratax45 = 0.0;
reratay45 = 0.0;
stdevx45 = 0.0;
stdevy45 = 0.0;

korelasi90 =0.0;
px90 = 0;
py90 = 0;
reratax90 = 0.0;
reratay90 = 0.0;
stdevx90 = 0.0;
stdevy90 = 0.0;
korelasi135 =0.0;
px135 = 0;
py135 = 0;
```

```

reratax135 = 0.0;
reratay135 = 0.0;
stdevx135 = 0.0;
stdevy135 = 0.0;
for a=0 : 255
    for b=0 : 255
        px0 = px0 + a * GLCM0 (a+1, b+1);
        py0 = py0 + b * GLCM0 (a+1, b+1);

        px45 = px45 + a * GLCM45 (a+1, b+1);
        py45 = py45 + b * GLCM45 (a+1, b+1);

        px90 = px90 + a * GLCM90 (a+1, b+1);
        py90 = py90 + b * GLCM90 (a+1, b+1);

        px135 = px135 + a * GLCM135 (a+1, b+1);
        py135 = py135 + b * GLCM135 (a+1, b+1);
    end
end
%Hitung deviasi standar
for a=0 : 255
    for b=0 : 255
        stdevx0 = stdevx0 + (a-px0) * (a-px0) * GLCM0(a+1, b+1);
        stdevy0 = stdevy0 + (b-py0) * (b-py0) * GLCM0(a+1, b+1);

        stdevx45 = stdevx45 + (a-px45) * (a-px45) * GLCM45(a+1, b+1);
        stdevy45 = stdevy45 + (b-py45) * (b-py45) * GLCM45(a+1, b+1);

        stdevx90 = stdevx90 + (a-px90) * (a-px90) * GLCM90(a+1, b+1);
        stdevy90 = stdevy90 + (b-py90) * (b-py90) * GLCM90(a+1, b+1);

        stdevx135 = stdevx135 + (a-px135) * (a-px135) * GLCM135(a+1, b+1);
        stdevy135 = stdevy135 + (b-py135) * (b-py135) * GLCM135(a+1, b+1);
    end
end

```

8. menghitung korelasi dan mengambil fitur tekstur menggunakan sourcecode berikut :

```
%Hitung Korelasi
for a=0 : 255
    for b=0 : 255
        korelasio = korelasio + ((a-px0)*(b-py0) * GLCM0(a+1, b+1) / (stdevx0*stdevy0));
        korelasi45 = korelasi45 + ((a-px45)*(b-py45) * GLCM45(a+1,
b+1) / (stdevx45*stdevy45));
        korelasi90 = korelasi90 + ((a-px90)*(b-py90) * GLCM90(a+1,
b+1) / (stdevx90*stdevy90));
        korelasi135 = korelasi135 + ((a-px135)*(b-py135) * GLCM135(a+1,
b+1) / (stdevx135*stdevy135));
    end
end
%Hasil Fitur
G0.asm = asm0;
G0.kontras = kontras0;
G0.idm = idm0;
G0.entropi = entropi0;
G0.korelasi = korelasio;

G45.asm = asm45;
G45.kontras = kontras45;
G45.idm = idm45;
G45.entropi = entropi45;
G45.korelasi = korelasi45;

G90.asm = asm90;
G90.kontras = kontras90;
G90.idm = idm90;
G90.entropi = entropi90;
G90.korelasi = korelasio90;

G135.asm = asm135;
G135.kontras = kontras135;
G135.idm = idm135;
G135.entropi = entropi135;
G135.korelasi = korelasio135;

fitur=[G0.asm G0.kontras G0.idm G0.entropi G0.korelasi G45.asm G45.kontras G45.idm
G45.entropi G45.korelasi G90.asm G90.kontras G90.idm G90.entropi G90.korelasi G135.asm
G135.kontras G135.idm G135.entropi G135.korelasi];
xlswrite('fiturtekstur.xls',fitur);
```

f. Hasil dan Pembahasan

1. Dokumentasi Tugas berupa laporan (definisi, proses dan penerapan ekstraksi fitur tekstur)
2. Dokumentasi Tugas berupa file ppt

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 25

Pokok Bahasan	:Low Pass Filtering
Acara Praktikum/Praktik	:Minggu 7/1
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	:100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa dapat menjelaskan tentang Metode Low Pass Filtering
2. Mahasiswa mampu mengimplementasi Metode Low Pass Filtering

b. Indikator Penilaian

Kemampuan mahasiswa dalam memahami dan mengimplementasi Metode Low Pass Filtering menggunakan matlab

c. Dasar Teori

Operasi spasial dalam pengolahan citra digital dilakukan melalui penggunaan suatu **kernel konvolusi 2-dimensi**. Metode **image enhancement** dalam **operasi spasial** antara lain **low-pass filtering** dan **high-pass filtering**. **Low-pass Filtering** **Low-pass filtering** adalah proses filter yang melewatkkan komponen citra dengan nilai intensitas yang rendah dan meredam komponen citra dengan nilai intensitas yang tinggi. **Low pass filter** akan menyebabkan citra menjadi lebih halus dan lebih blur. Aturan kernel untuk **low-pass filter** adalah:

1. Semua koefisien kernel harus positif
2. Jumlah semua koefisien kernel harus sama dengan 1. Contoh kernel yang dapat digunakan pada *low-pass filtering* adalah:

$$(i) \begin{bmatrix} 1/16 & 1/8 & 1/16 \\ 1/8 & 1/4 & 1/8 \\ 1/16 & 1/8 & 1/16 \end{bmatrix} \quad (ii) \begin{bmatrix} 1/10 & 1/10 & 1/10 \\ 1/10 & 1/5 & 1/10 \\ 1/10 & 1/10 & 1/10 \end{bmatrix} \quad (iii) \begin{bmatrix} 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \end{bmatrix}$$

$$\Sigma = 1$$

$$\Sigma = 1$$

$$\Sigma = 1$$

Gambar 42. Kernel

d. Alat dan Bahan

1. Kertas A4
2. Pulpen
3. Laptop
4. Citra (image)
5. Aplikasi Matlab

e. Prosedur Kerja

1. Siapkan citra, kemudian buka aplikasi matlab

2. Tuliskan kode berikut untuk mengimplementasikan Low Pass Filter :

```
I=imread('cameraman.tif');
lpf1=[1/16 1/8 1/16;1/8 1/4 1/8;1/16 1/8 1/16];
lpf2=[1/10 1/10 1/10;1/10 1/5 1/10;1/10 1/10 1/10];
lpf3=[1 1 1;1 1 1;1 1 1]/9;
J1=uint8(conv2(double(I),lpf1,'same'));
J2=uint8(conv2(double(I),lpf2,'same'));
J3=uint8(conv2(double(I),lpf3,'same'));
```

3. Tampilkan citra asli, citra setelah dilakukan filtering 1,2, dan 3 sehingga tampilannya seperti berikut! (Penggunaan GUI sangat dianjurkan)



4. Lakukan Analisa !

f. Hasil dan Pembahasan

1. Dokumentasi Tugas berupa laporan (definisi, proses dan penerapan low pass filtering)
2. Dokumentasi Tugas berupa file ppt

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 26

Pokok Bahasan	:High Pass Filtering
Acara Praktikum/Praktik	:Minggu 7/2
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	:100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa dapat menjelaskan tentang Metode High Pass Filtering
2. Mahasiswa mampu mengimplementasi Metode High Pass Filtering

b. Indikator Penilaian

Kemampuan mahasiswa dalam memahami dan mengimplementasi Metode High Pass Filtering

c. Dasar Teori

High-pass filtering adalah proses filter yang melewatkkan komponen citra dengan nilai intensitas yang tinggi dan meredam komponen citra dengan nilai intensitas yang rendah. *High pass filter* akan menyebabkan tepi objek tampak lebih tajam dibandingkan sekitarnya. HPF digunakan untuk melakukan proses deteksi tepi, sehingga filter ini sering disebut juga filter sharpening (penajaman). *High-pass filtering* merupakan kebalikan dari *low-pass filtering*, yaitu metode yang membuat sebuah sinyal atau citra menjadi kurang halus.

Metode yang digunakan adalah melakukan pelemahan dalam domain frekuensi yang memiliki frekuensi rendah. *highpass filtering* biasa digunakan untuk *Unsharp Masking*, *Deconvolution*, *Edge Detection*, mengurangi *blur*, atau menambah *noise*. Ciri-ciri kernel dari HPF adalah nilai-nilainya terdiri dari nilai positif, nol dan negatif, dan jumlah dari semua nilainya sama dengan nol. Aturan kernel untuk *high-pass filter* adalah:

1. Koefisien kernel boleh positif, negative, atau nol
2. Jumlah semua koefisien kernel adalah 0 atau 1

Contoh kernel yang dapat digunakan pada *high-pass filtering* adalah:

$$(i) \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix} \quad (ii) \begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix} \quad (iii) \begin{bmatrix} 1 & -2 & 1 \\ -2 & 5 & -2 \\ 1 & -2 & 1 \end{bmatrix}$$
$$\Sigma = 0 \qquad \qquad \Sigma = 1 \qquad \qquad \Sigma = 1$$

Gambar 43. Kernel

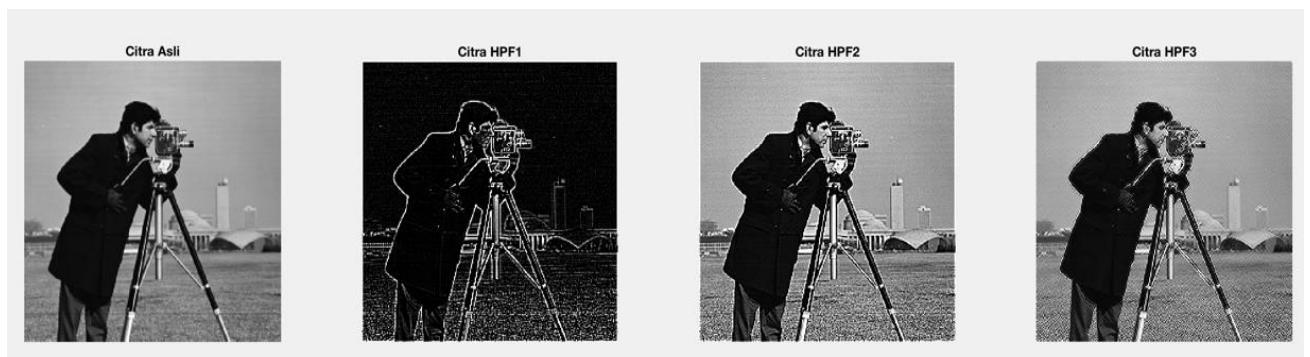
d. Alat dan Bahan

1. Kertas A4

2. Pulpen
 3. Laptop
 4. Citra (image)
 5. Aplikasi Matlab
- e. Prosedur Kerja
1. Siapkan citra, kemudian buka aplikasi matlab :
 2. Tuliskan kode berikut untuk mengimplementasikan Low Pass Filter :

```
I=imread('cameraman.tif');
hpf1=[-1 -1 -1;-1 8 -1;-1 -1 -1];
hpf2=[ 0 -1 0;-1 5 -1; 0 -1 0];
hpf3=[ 1 -2 1;-2 5 -2; 1 -2 1];
J1=uint8(conv2(double(I),hpf1,'same'));
J2=uint8(conv2(double(I),hpf2,'same'));
J3=uint8(conv2(double(I),hpf3,'same'));
```

3. Tampilkan citra asli, citra setelah dilakukan filtering 1,2, dan 3 sehingga tampilannya seperti berikut! (Penggunaan GUI sangat dianjurkan)



4. Lakukan Analisa !
- f. Hasil dan Pembahasan
1. Dokumentasi Tugas berupa laporan (definisi, proses dan penerapan low pass filtering)
 2. Dokumentasi Tugas berupa file ppt

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 27

Pokok Bahasan	:Median Filtering
Acara Praktikum/Praktik	:Minggu 7/3
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	:100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa dapat menjelaskan tentang Metode Median Filtering
2. Mahasiswa mampu mengimplementasi Metode Median Filtering

b. Indikator Penilaian

Kemampuan mahasiswa dalam memahami dan mengimplementasi Metode Median Filtering menggunakan matlab

c. Dasar Teori

Median filtering merupakan bagian filter non linier. *Median filtering* menghitung nilai dari piksel baru dimana nilai piksel pada pusat koordinat sliding window dengan nilai tengah (median) dari piksel di dalam window. Nilai tengah tersebut bergantung dengan ukuran sliding window. Ukuran window adalah ukuran dari citra tersebut.

Secara matematis, *median filtering* dinyatakan dengan persamaan rumus :

$$O(i,j) = \text{median}\{U(i+k-1, j+l-1), (k, l) \in W\}$$

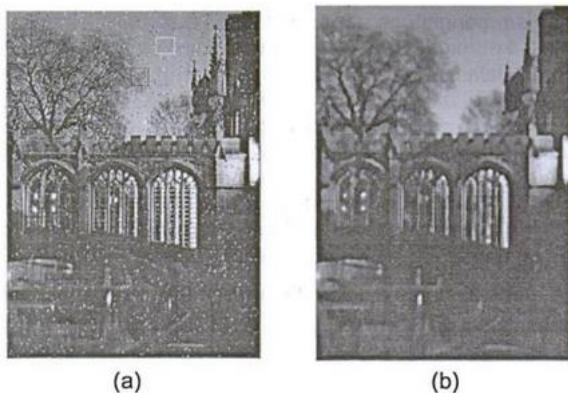
0 0 0 0 0	
0 1 1 1 0	
0 1 1 1 0	
0 1 1 1 0	
0 1 1 1 0	
0 1 9 1 0	0 1 0
0 1 1 1 0	1 1 1
0 0 0 1 0	1 1 1
	1 1 1
	1 1 1
	0 1 1

(a)

(b)

Gambar 44. (a) citra input (b) citra output median filtering 3 x 3

Nilai piksel 9 pada citra input dianggap derau (memiliki frekuensi tinggi) dengan *median filtering* derau tersebut hilang pada citra output, sedangkan contoh lain ditunjukkan pada Gambar 46.

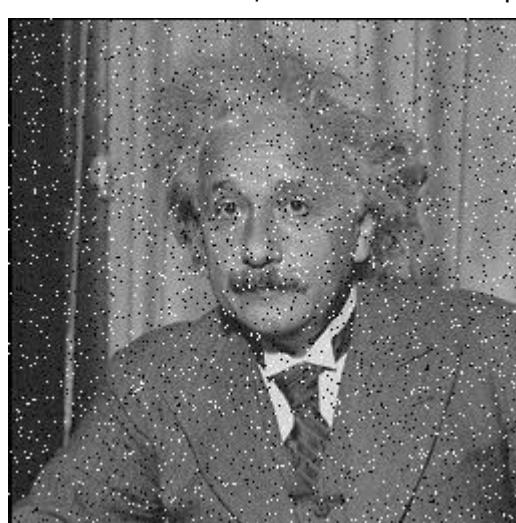


Gambar 45. (a) citra berderau dan (b) hasil median filtering

- d. Alat dan Bahan
 - 1. Kertas A4
 - 2. Pulpen
 - 3. Laptop
 - 4. Citra (image)
 - 5. Aplikasi Matlab

- e. Prosedur Kerja

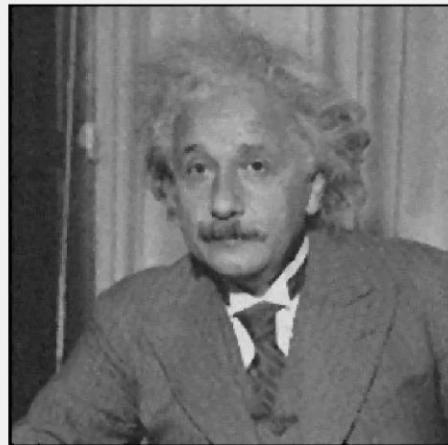
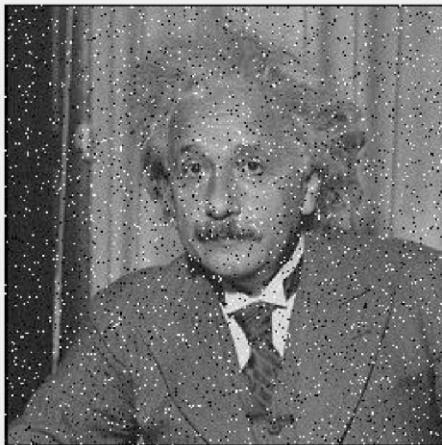
- 1. Siapkan citra berderau, kemudian buka aplikasi matlab :



- 2. Tuliskan kode berikut :

```
citra=imread('E:\1.Vani\BKPM PCV\instein.jpg');
filter=medfilt2(citra);
subplot(1,2,1), imshow(citra);
subplot(1,2,2), imshow(filter);
```

3. Tampilkan hasil median filter



4. Lakukan Analisa !

- f. Hasil dan Pembahasan
 - 1. Dokumentasi Tugas berupa laporan (definisi, proses dan penerapan low pass filtering)
 - 2. Dokumentasi Tugas berupa file ppt
- g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 28

Pokok Bahasan	: Gaussian Filtering
Acara Praktikum/Praktik	: Minggu 7/4
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa dapat menjelaskan tentang Metode Gaussian Filtering
2. Mahasiswa mampu mengimplementasi Metode Gaussian Filtering

b. Indikator Penilaian

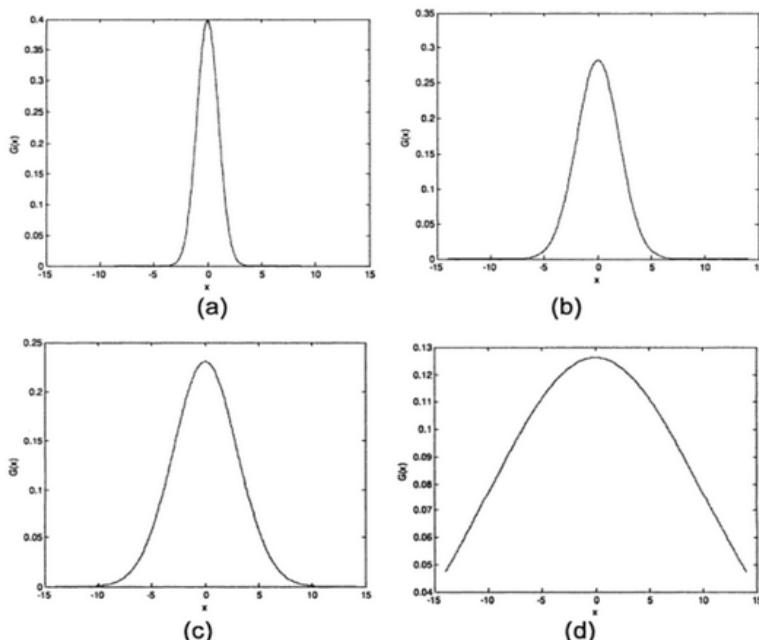
Kemampuan mahasiswa dalam memahami dan mengimplementasi Metode Gaussian Filtering menggunakan matlab

c. Dasar Teori

Gaussian filtering digunakan dalam bidang analis citra terutama untuk proses penghalusan (smoothing), pengaburan (bluring) dan menghilangkan derau (noise). Ukuran window adalah ukuran dari citra tersebut. Fungsi gaussian filtering 1 dimensi dinyatakan dengan persamaan rumus :

$$G(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{x^2}{2\sigma^2}\right)$$

Dengan σ menyatakan standar deviasi dari distribusi. Fungsi diatas diasumsikan memiliki mean 0 (pusat distribusi pada garis $x = 0$).



Gambar 46. Distribusi gaussian I-D berturut-turut distribusi gaussian dengan σ adalah 1,2,3 dan 10

Semakin besar nilai σ maka kurva distribusi gaussian semakin melebar dan puncaknya menurun.

0.000	0.004	0.054	0.242	0.399	0.342	0.054	0.004	0.000

0.038	0.092	0.171	0.249	0.282	0.249	0.171	0.092	0.038

0.095	0.140	0.184	0.218	0.230	0.218	0.184	0.140	0.095

0.117	0.121	0.124	0.126	0.126	0.126	0.124	0.121	0.117

Gambar 47. Kernel gaussian 1-D (a)-(b)-(c)-(d) dengan σ adalah 1,2,3 dan 10

Angka 0.399 pada Gambar 47(a) diperoleh dari :

$$G(0) = \frac{1}{\sqrt{(2)(3.14)(1)}} \exp \exp \left(-\frac{0^2}{(2)(1^2)} \right) = 0.3989$$

Bentuk 2-D dari fungsi gaussian adalah :

$$G(x) = \frac{1}{\sqrt{2\pi\sigma}} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$

0.0029	0.0131	0.0215	0.0131	0.0029
0.0131	0.0585	0.0965	0.0585	0.0131
0.0215	0.0965	0.1592	0.0965	0.0215
0.0131	0.0585	0.0965	0.0585	0.0131
0.0029	0.0131	0.0215	0.0131	0.0029

Gambar 48. Kernel distribusi gaussian 2-D dengan $\sigma = 1$ dan ukuran kernel 5 x 5

Angka 0.1592 pada Gambar 48 diperoleh dari :

$$G(0,0) = \frac{1}{\sqrt{(2)(3.14)(1^2)}} \exp \exp \left(-\frac{0^2 + 0^2}{(2)(1^2)} \right) = 0.1592$$

Proses penghalusan terhadap citra dapat dilakukan dengan proses konvolusi citra input dengan kernel gaussian. Tingkat atau derajat kehalusan citra hasil *gaussian filtering* dapat diatur dengan mengubah-ubah nilai σ .

d. Alat dan Bahan

1. Kertas A4
2. Pulpen
3. Laptop
4. Citra (image)
5. Aplikasi Matlab

e. Prosedur Kerja

1. Siapkan citra, kemudian buka aplikasi matlab :



2. Tuliskan kode berikut :

```
citra=imread('E:\1.Vani\BKPM PCV\gaussian.jpg');
filter1=imgaussfilt(citra);
filter2 = imgaussfilt(citra,2);
subplot(1,3,1), imshow(citra);
subplot(1,3,2), imshow(filter1);
subplot(1,3,3), imshow(filter2);
```

3. Run sourcecode tersebut, kemudian Analisa hasilnya serta berikan perbedaan antara dua sourcecode *gaussian filtering* !

f. Hasil dan Pembahasan

1. Dokumentasi Tugas berupa laporan (definisi, proses dan penerapan gaussian filtering)
2. Dokumentasi Tugas berupa file ppt

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 29

Pokok Bahasan	: Deteksi Tepi
Acara Praktikum/Praktik	: Minggu 9/1
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

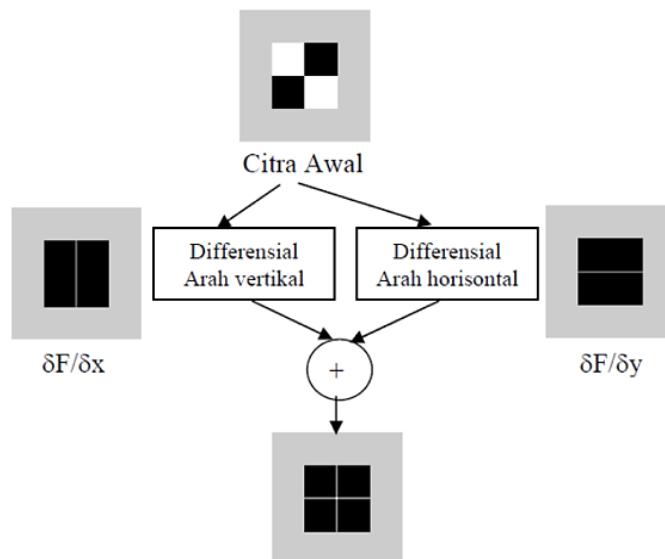
1. Mahasiswa dapat menjelaskan tentang Metode Deteksi Tepi Sobel
2. Mahasiswa mampu mengimplementasi Metode Deteksi Tepi Sobel

b. Indikator Penilaian

Kemampuan mahasiswa dalam memahami dan mengimplementasi Metode Deteksi Tepi menggunakan matlab

c. Dasar Teori

Deteksi tepi (Edge detection) adalah operasi yang dijalankan untuk mendeteksi garis tepi (edges) yang membatasi dua wilayah citra homogen yang memiliki tingkat kecerahan yang berbeda. Deteksi tepi memanfaatkan perubahan nilai intensitas yang drastis pada batas dua area. Defenisi tepi di sini adalah himpunan piksel yang terhubung yang terletak pada batas dua area. Perlu diketahui, tepi sesungguhnya mengandung informasi sangat penting. Informasi yang diperoleh dapat berupa bentuk maupun ukuran objek. Suatu titik (x,y) dikatakan sebagai tepi (edge) dari suatu citra bila titik tersebut mempunyai perbedaan yang tinggi dengan titik tetangganya. Gambar berikut menggambarkan bagaimana tepi suatu gambar diperoleh.

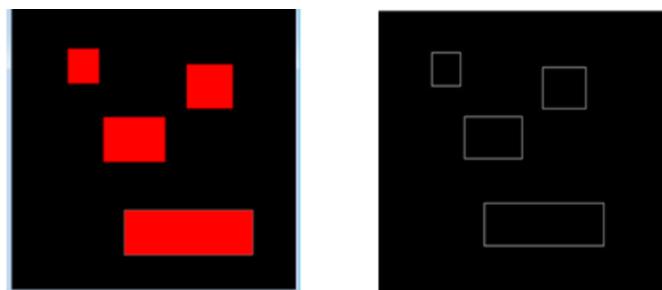


Gambar 49. Ilustrasi Proses Deteksi Tepi

Deteksi Tepi Sobel

Metode Sobel merupakan pengembangan metode Robert dengan menggunakan filter HPF yang diberi satu angka nol penyanga. Metode ini mengambil prinsip dari fungsi laplacian dan gaussian yang dikenal sebagai fungsi untuk membangkitkan HPF. Kelebihan dari metode Sobel ini adalah kemampuan untuk mengurangi noise sebelum melakukan perhitungan deteksi tepi. Biasanya operator Sobel menempatkan penekanan atau pembobotan pada piksel-piksel yang lebih dekat dengan titik pusat jendela, sehingga pengaruh piksel-piksel tetangga akan berbeda sesuai dengan letaknya terhadap titik di mana gradien dihitung. Dari susunan nilai-nilai pembobotan pada jendela juga terlihat bahwa perhitungan terhadap gradien juga merupakan gabungan dari posisi mendatar dan posisi vertikal.

Metode Sobel adalah sebuah metode pendektsian tepi yang tergolong kategori *gradient edge detection* (Murdianto, 2007), dimana metode ini akan mendeteksi nilai maksimum dan nilai minimum dari tepi sebuah gambar sesuai ditunjukkan pada gambar 50 di bawah ini:



Gambar 50. Pendektsian Tepi Gambar Asli (a) menjadi Pola (b)

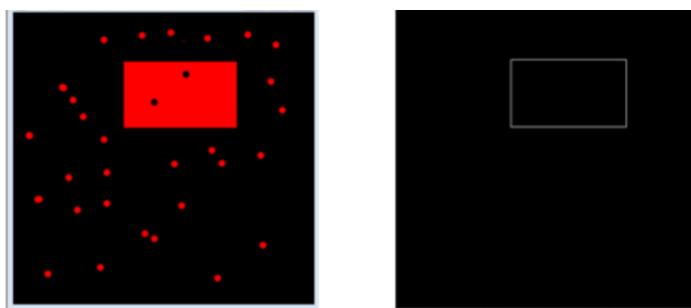
Apabila *threshold* (batas) dari sebuah obyek sudah sesuai dengan nilai yang ditetapkan, maka dapat ditentukan apakah nilai tersebut merupakan batas dari sebuah obyek atau bukan. Proses pendektsian ini dilakukan terhadap seluruh cell matrik dari sebuah citra 2 dimensi. Artinya sistem akan melakukan mapping terhadap sebuah citra sehingga diperoleh seluruh tepi dari obyek-obyek dalam citra tersebut.

Setelah seluruh tepi obyek dari citra tersebut telah dipetakan, proses berikutnya adalah melakukan penyaringan terhadap obyek yang perlu diolah dan obyek yang tidak perlu diolah seperti *noise* seperti ditunjukkan pada gambar 50 sebagai berikut. *Noise* pada gambar dapat muncul dari beberapa sebab, kualitas cahaya saat pengambilan gambar menggunakan kamera, *noise* pada obyek yang diambil seperti debu, batu atau obyek-obyek lain yang tidak seharusnya muncul saat gambar diambil atau bisa juga *noise* berasal dari kualitas kamera yang digunakan, misalnya kamera yang tidak memiliki fitur *auto focus* sehingga

gambar yang diambil memiliki “bayangan” atau duplikasi obyek. *Noise* pada gambar digital dapat diperbaiki dengan beberapa tools dari perangkat lunak pengolah citra seperti Adobe Photoshop dan lain-lain.

Pada Gambar 51 dijelaskan, walaupun pada citra asli (a) memiliki banyak *noise* baik *noise* yang berada di latar belakang maupun *noise* yang berada di dalam obyek, akan tetapi *noise-noise* tersebut memiliki dimensi yang kurang dari nilai yang ditetapkan sebagai sebuah obyek, maka *noise* tersebut akan dihilangkan dan terbentuklah pola yang tergambar pada gambar (b).

Noise adalah gangguan dalam citra yang dapat membuat kesalahan pendekripsi yang kemudian berujung pada kesalahan pengenalan pola dari citra yang diolah. *Noise* merupakan tantangan terbesar dalam konsep pengolahan citra. Sistem dapat mengira *noise* sebagai obyek (kesalahan negatif) maupun sebaliknya obyek sebagai *noise* (kesalahan positif).



Gambar 51. Pembersihan Noise dari Citra

Metode deteksi tepi sobel cukup baik dalam pendekripsi *noise* sebelum *noise* yang ada dalam citra tersebut memiliki toleransi yang cukup besar. Pada gambar 2 di atas, *noise* dipandang sebagai sebuah obyek yang memiliki dimensi (ukuran) tertentu yang kemudian “diabaikan” oleh sistem sehingga tidak akan diproses kemudian. Akan tetapi apabila dimensi *noise* ini melebihi ukuran yang diijinkan, maka *noise* tersebut akan dianggap sebagai sebuah obyek seperti pada gambar 51 di atas. *Noise* bisa juga muncul sebagai warna toleransi yang cukup mirip seperti ditunjukkan pada gambar 52 di bawah ini.



Gambar 52. Implementasi pada Gambar dengan Kemiripan Warna Tinggi

Metode deteksi tepi Sobel dilakukan dengan operasi matriks terhadap piksel-piksel di sekitar piksel yang sedang di-scan. Sebuah citra tersusun atas matriks 2

dimensi yang masing-masing sel matrik tersebut berisi kombinasi warna merah, hijau dan biru seperti ditunjukkan pada gambar 53 sebagai berikut.

[1][1]	[1][2]	[1][3]	[1][4]	..	[1][n]	RGB	RGB	RGB	RGB	RGB	RGB
[2][1]	[2][2]	[2][3]	[2][4]	..	[2][n]	RGB	RGB	RGB	RGB	RGB	RGB
[3][1]	[3][2]	[3][3]	[3][4]	..	[3][n]	RGB	RGB	RGB	RGB	RGB	RGB
[4][1]	[4][2]	[4][3]	[4][4]	..	[4][n]	RGB	RGB	RGB	RGB	RGB	RGB
..	RGB	RGB	RGB	RGB	RGB	RGB
[m][1]	[m][2]	[m][3]	[m][4]	..	[m][n]	RGB	RGB	RGB	RGB	RGB	RGB

Gambar 53. Matriks Warna dalam Piksel Gambar 2D

Matriks Sobel merupakan matriks 3x3 dengan koefisien nilai RGB yang telah ditentukan seperti pada gambar 54 sebagai berikut.

-1	0	+1	+1	+2	+1
-2	0	+2	0	0	0
-1	0	+1	-1	-2	-1

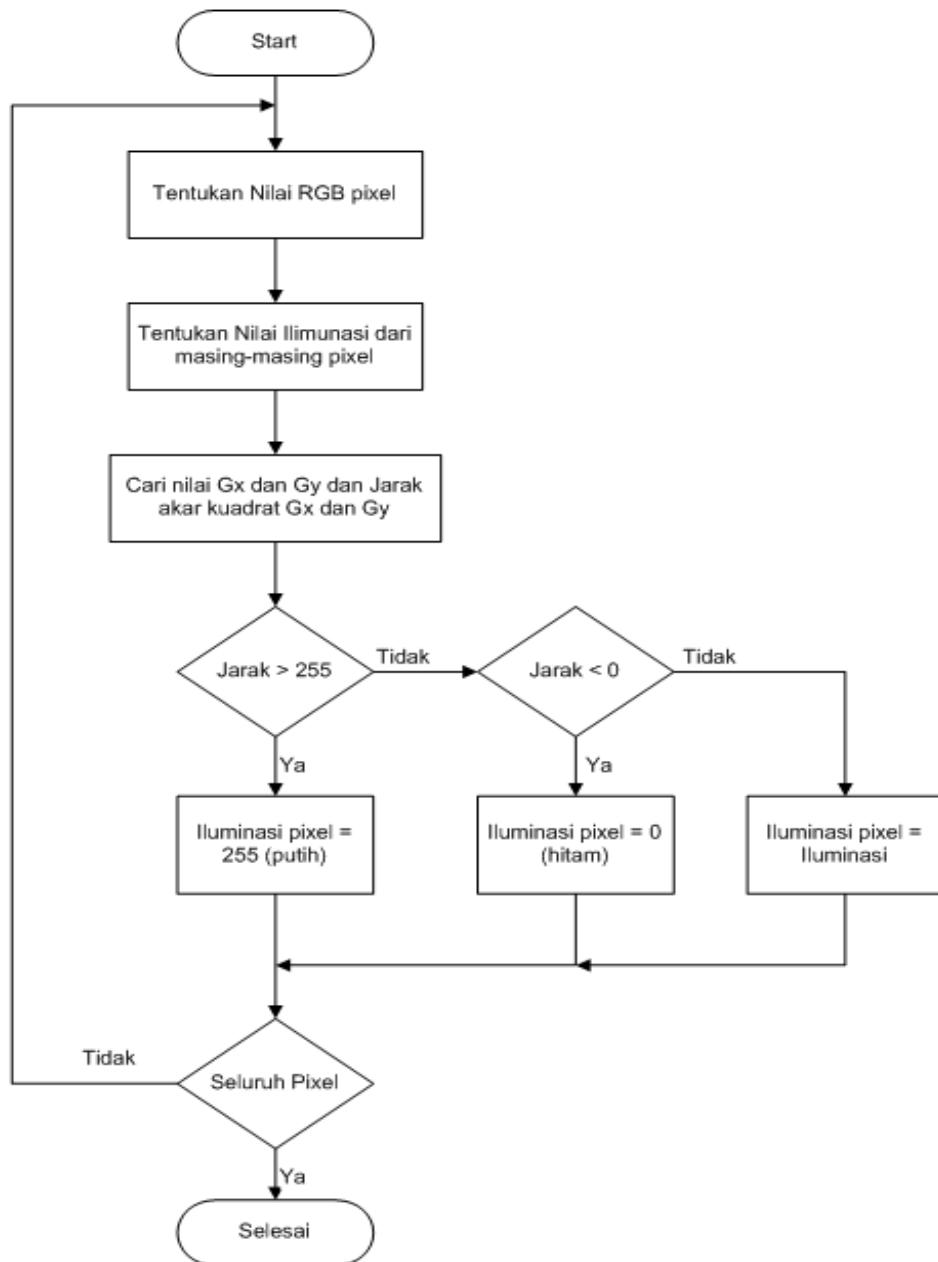
Gx

+1	+2	+1
0	0	0
-1	-2	-1

Gy

Gambar 54. Matriks sobel 3x3

Setiap sel dari matriks di gambar 54 akan diolah menggunakan koefisien matriks sehingga diperoleh jarak antar warna. Jika jarak antar warna tersebut kurang dari nilai yang ditentukan maka konversi nilai RGB pada pixel tersebut menjadi putih dan jika lebih dari nilai yang ditentukan, maka konversi nilai RGB pada pixel tersebut sebagai hitam sehingga diperoleh matriks warna baru. Pendekripsi tepi obyek menggunakan metode Sobel dijelaskan pada diagram alir gambar 55 di bawah ini:

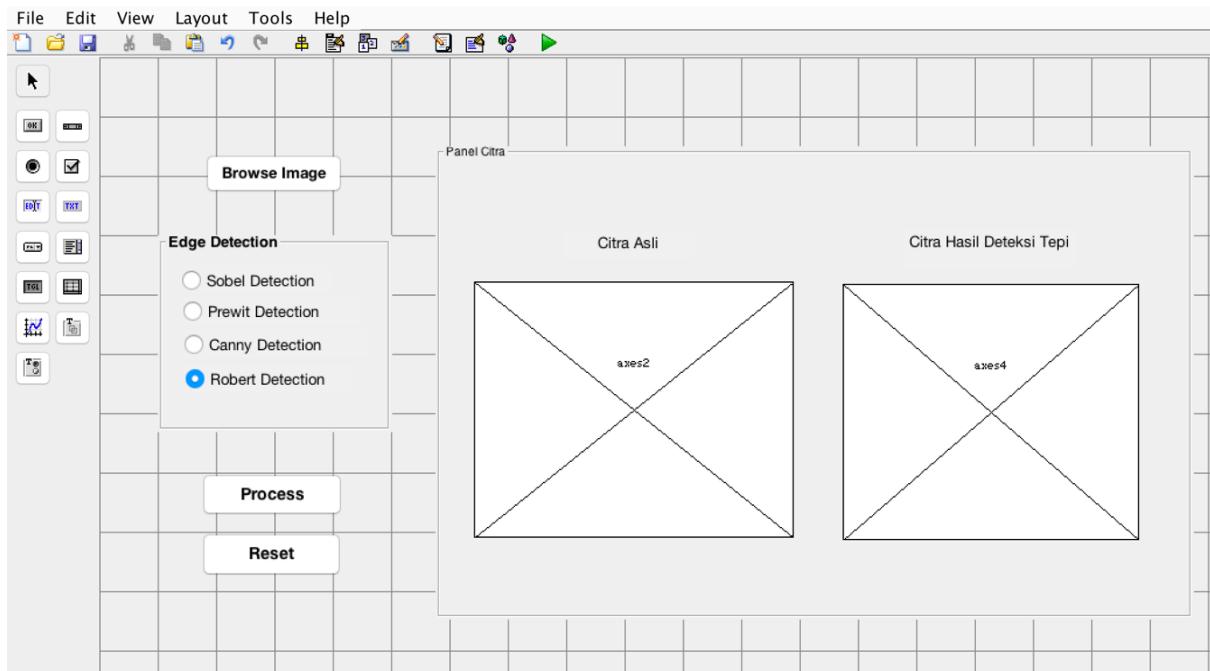


Gambar 55. Flowchart Metode Deteksi Tepi Sobel

- d. Alat dan Bahan
1. Kertas A4
 2. Pulpen
 3. Laptop
 4. Citra (image)
 5. Aplikasi Matlab

e. Prosedur Kerja

1. Buatlah tampilan GUI seperti dibawah ini :



Dimana ketentuan sebagai berikut:

- Terdapat button untuk *upload* gambar
- Terdapat list radio button untuk memilih jenis metode deteksi tepi yang akan diproses
- Terdapat button process untuk mengeksekusi program
- Terdapat button Reset untuk mengulang proses dari awal
- Terdapat 2 axes untuk menampilkan citra asli maupun citra hasil proses deteksi tepi

2. Tuliskan kode berikut :

```
citra_dtSobel=edge(citra_ori, 'sobel');
```

3. Jalankan script pada 5 citra berbeda dan lakukan Analisa!

f. Hasil dan Pembahasan

-

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 30

Pokok Bahasan	: Deteksi Tepi
Acara Praktikum/Praktik	: Minggu 9/2
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa dapat menjelaskan tentang Metode Deteksi Tepi Prewitt
2. Mahasiswa mampu mengimplementasi Metode Deteksi Tepi Prewitt

b. Indikator Penilaian

Kemampuan mahasiswa dalam memahami dan mengimplementasi Metode Deteksi Tepi menggunakan matlab

c. Dasar Teori

Metode Prewitt merupakan pengembangan metode Robert dengan menggunakan filter HPF (High Pass Filter) yang diberi satu angka nol penyanga. Metode ini mengambil prinsip dari fungsi laplacian yang dikenal sebagai fungsi untuk membangkitkan HPF (High Pass Filter). Kernel filter yang digunakan dalam metode Prewitt ini adalah:

$$H = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \text{ dan } V = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

d. Alat dan Bahan

1. Kertas A4
2. Pulpen
3. Laptop
4. Citra (image)
5. Aplikasi Matlab

e. Prosedur Kerja

1. Buatlah tampilan GUI seperti acara praktikum sebelumnya (Acara 29)
2. Tuliskan script untuk mengimplementasi metode deteksi tepi menggunakan metode Prewitt.

```
citra_dtPrewitt=edge(citra_ori, 'prewitt');
```

3. Jalankan script pada 5 citra berbeda dan lakukan Analisa!

f. Hasil dan Pembahasan

-

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 31

Pokok Bahasan	: Deteksi Tepi
Acara Praktikum/Praktik	: Minggu 9/3
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	: 100 menit

- a. Capaian Pembelajaran Mata Kuliah (CPMK)
 - 1. Mahasiswa dapat menjelaskan tentang Metode Deteksi Tepi Canny
 - 2. Mahasiswa mampu mengimplementasi Metode Deteksi Tepi Canny
- b. Indikator Penilaian
 - Kemampuan mahasiswa dalam memahami dan mengimplementasi Metode Deteksi Tepi menggunakan matlab
- c. Dasar Teori

Perancangan sebuah prosedur dengan menerapkan langkah-langkah metode Canny edge detection akan menghasilkan sebuah tampilan gambar yang berbeda dengan menampilkan efek relief di dalamnya. Efek relief adalah seperti sebuah tampilan batu kasar yang diukir, yaitu garis-garis kasar yang membentuk sebuah penggambaran objek di dalamnya. Efek relief terbentuk dari bayangan terang dan gelap. Kedua bayangan ini terjadi akibat adanya sorotan sinar mengenai gambar dari arah tertentu. Kelebihan dari metode Canny ini adalah kemampuan untuk mengurangi noise sebelum melakukan perhitungan deteksi tepi sehingga tepi-tepi yang dihasilkan lebih banyak. Deteksi Tepi Canny dapat mendeteksi tepian yang sebenarnya dengan tingkat error yang minimum dengan kata lain, operator Canny didesain agar menghasilkan citra tepian yang optimal.
- d. Alat dan Bahan
 - 1. Kertas A4
 - 2. Pulpen
 - 3. Laptop
 - 4. Citra (image)
 - 5. Aplikasi Matlab
- e. Prosedur Kerja
 - a. Buatlah tampilan GUI seperti acara praktikum sebelumnya (Acara 29)
 - b. Tuliskan script berikut untuk mengimplementasi metode deteksi tepi menggunakan metode Canny.

citra_dtCanny = edge(citra_ori, 'canny');

 - c. Jalankan script pada 5 citra berbeda dan lakukan Analisa!
- f. Hasil dan Pembahasan

-

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 32

Pokok Bahasan	: Deteksi Tepi
Acara Praktikum/Praktik	: Minggu 9/4
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa dapat menjelaskan tentang Metode Deteksi Tepi Robert
2. Mahasiswa mampu mengimplementasi Metode Deteksi Tepi Robert

b. Indikator Penilaian

Kemampuan mahasiswa dalam memahami dan mengimplementasi Metode Deteksi Tepi menggunakan matlab

c. Dasar Teori

Metode Robert adalah nama lain dari teknik differensial yang dikembangkan di atas, yaitu differensial pada arah horizontal dan differensial pada arah vertikal, dengan ditambahkan proses konversi biner setelah dilakukan differensial. Teknik konversi biner yang disarankan adalah konversi biner dengan meratakan distribusi warna hitam dan putih [5]. Metode Robert ini juga disamakan dengan teknik DPCM (Differential Pulse Code Modulation). Kernel filter yang digunakan dalam metode Robert ini adalah:

$$H = [-1 \quad 1] \text{ dan } V = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$$

Operator Robert Cross merupakan salah satu operator yang menggunakan jendela matrik 2x2, operator ini melakukan perhitungan dengan mengambil arah diagonal untuk melakukan perhitungan nilai gradiennya. Sebenarnya operator sedehana ini hanya memeriksa sebuah piksel tambahan pada satu arah gradient tetapi karena yang diperiksa adalah piksel dalam arah diagonal, maka secara keseluruhan piksel-piksel yang terlibat membentuk jendela matrik 2x2. Bentuk jendela yang demikian lebih menekankan pemeriksaan pada kedua arah diagonal, dari pada arah horizontal atau arah vertikal, sehingga perbedaan yang terletak pada sisi-sisi miring objek akan terdeteksi dengan lebih baik.

d. Alat dan Bahan

1. Kertas A4
2. Pulpen
3. Laptop
4. Citra (image)
5. Aplikasi Matlab

e. Prosedur Kerja

1. Buatlah tampilan GUI seperti pada praktikum Acara 29
2. Tuliskan script berikut untuk mengimplementasi metode deteksi tepi menggunakan metode Robert.

```
citra_dtRoberts=edge(citra_ori, 'roberts');
```

3. Jalankan script pada 5 citra berbeda dan lakukan Analisa!
4. Lakukan juga Analisa dari metode Sobel, Prewitt, Canny dan Robert!

f. Hasil dan Pembahasan

-

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 33-34

Pokok Bahasan	: Implementasi <i>Computer Vision</i> pada Bidang Pertanian
Acara Praktikum/Praktik : Minggu 10/1-2	
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	: 2 x 100 Menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

- 1) Mahasiswa mampu menjelaskan konsep computer vision pada bidang pertanian.
- 2) Mahasiswa mampu menerapkan teknik-teknik pengolahan citra dan algoritma cerdas untuk menyelesaikan kasus bidang pertanian.

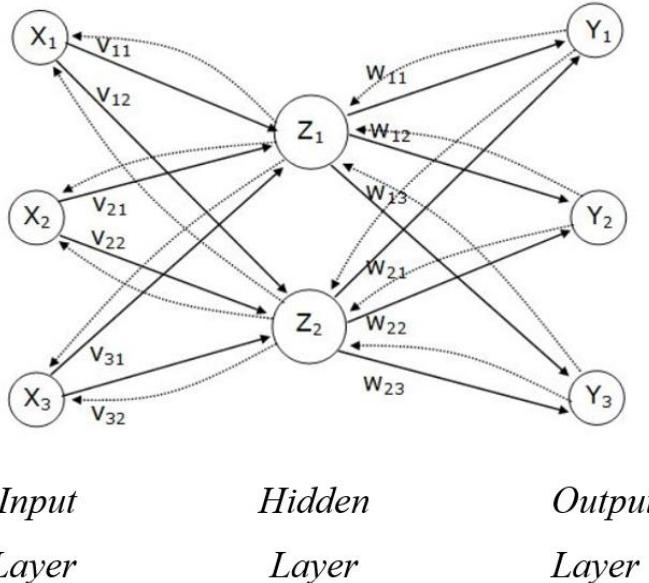
b. Indikator

Kemampuan mahasiswa dalam menjelaskan dan mengimplementasikan teknik-teknik pengolahan citra dalam kasus bidang pertanian.

c. Dasar Teori

Implementasi *computer vision* dalam bidang pertanian banyak dilakukan, hal ini merupakan bagian dari topik pertanian presisi. Contoh yang akan diberikan dalam BKPM ini adalah aplikasi *computer vision* untuk mengidentifikasi **bawang merah bagus** atau **busuk** berdasarkan ciri visual. Parameter ciri visual (fitur) yang digunakan yaitu tekstur dan warna. Untuk fitur tekstur diekstraksi menggunakan metode *Gray-Level Co-occurrence matrix* (GLCM) meliputi *Angular Second Moment* (ASM), *Inverse Different Moment* (IDM), kontras, dan korelasi masing-masing diambil dengan sudut 0° , 45° , 90° , 135° . Sedangkan fitur warna meliputi Red, Green, Blue. Sehingga jumlah total fitur adalah 19. Teori mengenai metode GLCM dapat dibaca kembali di BKPM **Acara 24** (Minggu ke 6).

Untuk proses klasifikasi datanya menggunakan model Jaringan Saraf Tiruan *Backpropagation*. Jaringan Saraf Tiruan (JST) adalah jaringan dari sekelompok unit pemroses kecil yang dimodelkan berlandaskan jaringan saraf manusia. JST adalah sebuah alat pemodelan data statistik non-linier. JST dapat digunakan untuk memodelkan hubungan yang kompleks selang input dan output untuk menemukan pola-pola pada data. *Backpropagation* merupakan JST *multilayer* karena terdiri atas 3 layer dalam proses pelatihannya antara lain: *input layer*, *hidden layer* dan *output layer* (Gambar 56) yang merupakan pengembangan dari single layer network JST. Karakteristik dari algoritma backpropagation yaitu meminimalkan error dengan cara menyesuaikan bobot berdasarkan perbedaan output dan target yang diinginkan.



Gambar 56. Arsitektur Model Backpropagation (Rizqy, 2012)

d. Alat dan Bahan

1. Laptop
2. Kertas A4 & Pulpen
3. Aplikasi *Matlab*
4. Aplikasi Ms. Excel

e. Prosedur Kerja

1. Buat *Project* baru di Matlab Anda, siapkan 1 folder berisi data citra yang terdiri dari bawang merah bagus dan bawang busuk. Data bisa diunduh pada tautan berikut <https://bit.ly/3b8YEHd>
2. Kode program untuk mengupload citra pada **satu folder**, *resize* citra, *remove background*, dan *grayscale*. *Resize* digunakan untuk merubah ukuran citra sesuai kebutuhan. *Remove background* silahkan lihat lagi materi di BKPM **Minggu 4**

```

for g=1:10
gambar1=imread(['D:\Data Latih\datalatih_',num2str(g),'.JPG']);

gambar = imresize(gambar1,[400,400]);
[BW, Img] = remove_bg(gambar);
red = Img(:,:,1);
green = Img(:,:,2);
blue = Img(:,:,3);

gray = rgb2gray(Img);
[M,N] = size(blue);
IFeature = zeros(M,N);

```

```

warna_red = mean(mean(red));
warna_green = mean(mean(green));
warna_blue = mean(mean(blue));
|
I = Img;
[tinggi, lebar] = size (I);

```

3. Menentukan sudut-sudut GLCM

```

% Bentuk GLCM
GLCM0 = zeros(256, 256);
total_piks0 =0;

GLCM45 = zeros(256, 256);
total_piks45 =0;

GLCM90 = zeros(256, 256);
total_piks90 =0;

GLCM135 = zeros(256, 256);
total_piks135 =0;

for y=2: tinggi-1
    for x=2: lebar-1
        %untuk sudut 0
        a = I(y,x);
        b = I(y,x+1);
        GLCM0(a+1, b+1) = GLCM0(a+1, b+1) + 1;
        total_piks0 = total_piks0 + 1;

        %untuk sudut 45
        a = I(y,x);
        b = I(y-1,x+1);
        GLCM45(a+1, b+1) = GLCM45(a+1, b+1) + 1;
        total_piks45 = total_piks45 + 1;

        %untuk sudut 90
        a = I(y,x);
        b = I(y-1,x);
        GLCM90(a+1, b+1) = GLCM90(a+1, b+1) + 1;
        total_piks90 = total_piks90 + 1;

        %untuk sudut 135
        a = I(y,x);
        b = I(y-1,x-1);
        GLCM135(a+1, b+1) = GLCM135(a+1, b+1) + 1;
        total_piks135 = total_piks135 + 1;
    end
end

```

```

GLCM0 = GLCM0 / total_piksello;
GLCM45 = GLCM45 / total_piksello45;
GLCM90 = GLCM90 / total_piksello90;
GLCM135 = GLCM135 / total_piksello135;

```

4. Menghitung Nilai ASM

```

%Hitung ASM
asm0 = 0.0;
asm45 = 0.0;
asm90 = 0.0;
asm135 = 0.0;

for a=0 : 255
    for b=0 : 255
        asm0 = asm0 + (GLCM0(a+1,b+1) * GLCM0(a+1,b+1));
        asm45 = asm45 + (GLCM45(a+1,b+1) * GLCM45(a+1,b+1));
        asm90 = asm90 + (GLCM90(a+1,b+1) * GLCM90(a+1,b+1));
        asm135 = asm135 + (GLCM135(a+1,b+1) * GLCM135(a+1,b+1));
    end
end

```

5. Menghitung Nilai IDM

```

%Hitung IDM
idm0 = 0.0;
idm45 = 0.0;
idm90 = 0.0;
idm135 = 0.0;

for a=0 : 255
    for b=0 : 255
        idm0 = idm0 + (GLCM0(a+1,b+1) / (1+(a-b)*(a-b)));
        idm45 = idm45 + (GLCM45(a+1,b+1) / (1+(a-b)*(a-b)));
        idm90 = idm90 + (GLCM90(a+1,b+1) / (1+(a-b)*(a-b)));
        idm135 = idm135 + (GLCM135(a+1,b+1) / (1+(a-b)*(a-b)));
    end
end

```

6. Menghitung Nilai Kontras

```

%Hitung Kontras
kontras0 = 0.0;
kontras45 = 0.0;
kontras90 = 0.0;
kontras135 = 0.0;

for a=0 : 255
    for b=0 : 255
        kontras0 = kontras0 + (a-b)*(a-b)* GLCM0(a+1,b+1);
        kontras45 = kontras45 + (a-b)*(a-b)* GLCM45(a+1,b+1);
        kontras90 = kontras90 + (a-b)*(a-b)* GLCM90(a+1,b+1);
        kontras135 = kontras135 + (a-b)*(a-b)* GLCM135(a+1,b+1);
    end
end

```

7. Menghitung Kovarian (PX) dan (PY)

```
%Hitung Kovarians
%Hitung px [] dan py [] dulu
korelasi0 =0.0;
px0 = 0;
py0 = 0;
reratax0 = 0.0;
reratay0 = 0.0;
stdevx0 = 0.0;
stdevy0 = 0.0;

korelasi45 =0.0;
px45 = 0;
py45 = 0;
reratax45 = 0.0;
reratay45 = 0.0;
stdevx45 = 0.0;
stdevy45 = 0.0;
```

```
korelasi90 =0.0;
px90 = 0;
py90 = 0;
reratax90 = 0.0;
reratay90 = 0.0;
stdevx90 = 0.0;
stdevy90 = 0.0;

korelasi135 =0.0;
px135 = 0;
py135 = 0;
reratax135 = 0.0;
reratay135 = 0.0;
stdevx135 = 0.0;
stdevy135 = 0.0;
```

```
for a=0 : 255
    for b=0 : 255
        px0 = px0 + a * GLCM0 (a+1, b+1);
        py0 = py0 + b * GLCM0 (a+1, b+1);

        px45 = px45 + a * GLCM45 (a+1, b+1);
        py45 = py45 + b * GLCM45 (a+1, b+1);

        px90 = px90 + a * GLCM90 (a+1, b+1);
        py90 = py90 + b * GLCM90 (a+1, b+1);

        px135 = px135 + a * GLCM135 (a+1, b+1);
        py135 = py135 + b * GLCM135 (a+1, b+1);
    end
end
```

8. Menghitung Deviasi Standart

```
%Hitung deviasi standar
for a=0 : 255
    for b=0 : 255
        stdevx0 = stdevx0 + (a-px0) * (a-px0) * GLCM0(a+1, b+1);
        stdevy0 = stdevy0 + (b-py0) * (b-py0) * GLCM0(a+1, b+1);

        stdevx45 = stdevx45 + (a-px45) * (a-px45) * GLCM45(a+1, b+1);
        stdevy45 = stdevy45 + (b-py45) * (b-py45) * GLCM45(a+1, b+1);

        stdevx90 = stdevx90 + (a-px90) * (a-px90) * GLCM90(a+1, b+1);
        stdevy90 = stdevy90 + (b-py90) * (b-py90) * GLCM90(a+1, b+1);

        stdevx135 = stdevx135 + (a-px135) * (a-px135) * GLCM135(a+1, b+1);
        stdevy135 = stdevy135 + (b-py135) * (b-py135) * GLCM135(a+1, b+1);

    end
end
```

9. Menghitung Korelasi

```
%Hitung Korelasi
for a=0 : 255
    for b=0 : 255
        korelasi0 = korelasi0 + ((a-px0)*(b-py0) * GLCM0(a+1, b+1)/(stdevx0*stdevy0));
        korelasi45 = korelasi45 + ((a-px45)*(b-py45) * GLCM45(a+1, b+1)/(stdevx45*stdevy45));
        korelasi90 = korelasi90 + ((a-px90)*(b-py90) * GLCM90(a+1, b+1)/(stdevx90*stdevy90));
        korelasi135 = korelasi135 + ((a-px135)*(b-py135) * GLCM135(a+1, b+1)/(stdevx135*stdevy135));

    end
end
```

10. Hasil ekstraksi fitur

```
%Hasil Fitur

Warna_red = warna_red;
Warna_green = warna_green;
Warna_blue = warna_blue;
```

```
G0asm = asm0;
G0idm = idm0;
G0kontras = kontras0;
G0korelasi = korelasi0;
```

```
G45asm = asm45;
G45idm = idm45;
G45kontras = kontras45;
G45korelasi = korelasi45;

G90asm = asm90;
G90idm = idm90;
G90kontras = kontras90;
G90korelasi = korelasi90;

G135asm = asm135;
G135idm = idm135;
G135kontras = kontras135;
G135korelasi = korelasi135;
```

11. Untuk penulisan kode (**fitur** sampai dengan **G135korelasi]**; harus didalam satu line)

```
fitur=[Warna_red ;Warna_green ;Warna_blue  
;G0asm ;G0idm ;G0kontras ;G0korelasi  
;G45asm ;G45idm ;G45kontras ;G45korelasi  
;G90asm ;G90idm ;G90kontras ;G90korelasi |  
;G135asm ;G135idm ;G135kontras ;G135korelasi];  
fitur_post(g,:)=fitur(1:19,:);
```

12. Kode program untuk **menyimpan** hasil ekstraksi fitur ke dalam bentuk exel dengan format *xlsx.

```
xlswrite('data_latih.xlsx',fitur_post);
```

f. Hasil dan Pembahasan

Dalam tahapan praktikum kali ini, Anda akan berhasil melakukan proses ekstraksi fitur tekstur dan menyimpannya dalam Ms. Excel.

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 35-36

Pokok Bahasan

**: Implementasi Computer Vision
pada Bidang Pertanian**

Acara Praktikum/Praktik : Minggu 10/3-4

Tempat

: Laboratorium Multimedia Cerdas

Alokasi Waktu

: 2 x 100 Menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

- 1) Mahasiswa mampu menjelaskan konsep computer vision pada bidang pertanian.
- 2) Mahasiswa mampu menerapkan teknik-teknik pengolahan citra dan algoritma cerdas untuk menyelesaikan kasus bidang pertanian.

b. Indikator

Kemampuan mahasiswa dalam menjelaskan dan mengimplementasikan teknik-teknik pengolahan citra dalam kasus bidang pertanian.

c. Dasar Teori

d. Alat dan Bahan

1. Laptop
2. Kertas A4 & Pulpen
3. Aplikasi *Matlab*
4. Aplikasi Ms. Excel

e. Prosedur Kerja

1. Lanjutkan *Project* di Matlab Anda yang sebelumnya.
2. Buat desain User Interface **Pelatihan Citra** seperti berikut

PELATIHAN CITRA BAWANG MERAH

The user interface for 'PELATIHAN CITRA BAWANG MERAH' includes:

- An input field labeled "Masukkan Data Excel" and a "Browse" button.
- A table with columns: Red, Green, Blue, G0.Asm, G0.Idm, G0.Kontras, G0.Korelasi, G45.Asm, G45.Idm, G45.Kontras, G45.Korelasi, G90.Asm, G90.Idm, G90.Kontras, G90.Korelasi. Rows 1, 2, 3, and 4 are present.
- A large image of a red onion.
- A small table labeled "Hasil" and "Target".
- An "Akurasi" input field.
- A legend box with "Keterangan":
 - 0 : Bagus
 - 1 : Busuk
- Buttons labeled "Proses" and "Reset".

3. Kode program Pelatihan citra untuk meupload data exel yang sudah diekstraksi fitur dan disimpan ke dalam bentuk exel dengan format *xlsx. Untuk pemanggilan

data excel pada sheet pertama dan xlRange membaca baris dan kolom. Data akan disimpan sebagai data latih dengan format *mat. **Input** adalah nilai hasil ekstraksi fitur pada kolom 1-19 dan **target** pada kolom ke20, nilai target kita tentukan sendiri. Digunakan pada “**Browser**”

```
[nama_file1, nama_path1]=uigetfile('*xlsx','Select xlsx file');
Path = [nama_path1 nama_file1];
sheet = 1;
xlRange = 'B2:U102';
save D:\Skripsi\Matlab\OtakAtik\data_latih.mat Path sheet xlRange

Data = xlsread(Path, sheet, xlRange);
input = Data(:,1:19);
target = Data(:,20);
[m,n] = size(input);

set(handles.uitable1,'Data',input)
set(handles.edit1,'String',Path)
```

4. Kode program Jaringan saraf tiruan digunakan pada tombol “**Proses**”

```
load 'data_latih.mat'

Data = xlsread(Path, sheet, xlRange);
input = Data(:,1:19)';
target = Data(:,20)';
[m,n] = size(input);

layer=10;
net = newff(input,target,layer,{'logsig'},'traingd');
net.performFcn = 'mse';
net.trainParam.epochs =500;
net.trainParam.goal = 0.1;
net.trainParam.lr=0.2;

% net=init(net);
[net,tr,Y,E] = train(net,input,target);
output = round(sim(net,input));

%bobot yang dihasilkan setelah training
IW=net.IW{1,1}; b1=net.b{1,1};
LW1=net.LW{2,1}; b2=net.b{2,1};
jumlah_iterasi = tr.num_epochs;
nilai_keluaran = Y;
nilai_error = E;
error_MSE = (1/n)*sum(nilai_error.^2);

save D:\Skripsi\Matlab\OtakAtik\pelatihan1.mat
```

5. Menghitung akurasi

```
% hitung akurasi
[m,n] = find(output==target);
akurasi = sum(m)/100*100;
set(handles.text1,'String',akurasi);

klasifikasi=[output;target];
set(handles.uitable3,'Data',klasifikasi);

save D:\Skripsi\Matlab\OtakAtik\klasifikasi.mat
```

6. Kode program untuk **mereset** digunakan pada tombol “Reset”

```
pproyek=guidata(pelatihan);
mau=questdlg(['Reset ' get(proyek.figure1,'Name') ''], ['Anda ingin Reset Aplikasi ini?'],'Ya','Tidak','Ya');
if strcmp(mau,'Tidak')
return;
else
    set(handles.edit1,'String','');
    set(handles.text1,'String','');
    set(handles.uitable1,'Data', cell(size(get(handles.uitable1,'Data'))));
    set(handles.uitable3,'Data', cell(size(get(handles.uitable3,'Data'))));
end
```

7. Tampilan GUI Pengujian Citra



8. Kode program untuk pengujian citra meupload data exel yang sudah diekstraksi fitur dan disimpan ke dalam bentuk exel dengan format *xlxs. Untuk pemanggilan data exel pada sheet pertama dan xlRange membaca baris dan kolom. Data akan disimpan sebagai data latih dengan format *mat. **Input** adalah nilai hasil ekstraksi fitur pada kolom 1-19 dan **target** pada kolom ke20, nilai target kita tentukan sendiri. Digunakan pada tombol “**Browser**”
9. Kode program pelatihan citra digunakan pada tombol “**Proses**”

```
[nama_file1, nama_path1]=uigetfile('.xlsx','Select xlsx file');
Path = [nama_path1 nama_file1];
sheet = 1;

% membaca excel data uji
xlRange = 'B2:U15';
Data = xlsread(Path, sheet, xlRange);
input = Data(:,1:19);
target = Data(:,20);

[m,n] = size(input);
save data_uji.mat Path sheet xlRange
set(handles.uitable6,'Data',input)
set(handles.edit2,'String',Path)
```

```
load klasifikasi.mat net
load data_uji.mat

Data = xlsread(Path, sheet, xlRange);
input = Data(:,1:19)';
target = Data(:,20)';
[m,n] = size(input);

output = round(sim(net,input));
klasifikasi=[output; target];
set(handles.uitable2,'Data',klasifikasi)

% persentase data uji
[m,n] = find(output==target);
akurasi = (sum(m)/14)*100;
set(handles.text1,'string',akurasi);
```

10. Tampilan Gui Identifikasi



11. Kode program untuk tombol “Pilih Citra”

```
global Im
[nama_file,nama_path]= uigetfile({'*.*'});

if ~isequal(nama_file,0)
    Im = imread(fullfile(nama_path,nama_file));
    axes(handles.axes2)
    imshow(Im)
    handles.Img=Im;
    guidata(hObject,handles)
```

```

axes(handles.axes3)
cla reset
set(gca,'XTick',[])
set(gca,'YTick',[])

axes(handles.axes4)
cla reset
set(gca,'XTick',[])
set(gca,'YTick',[])

else
    return
end

```

12. Kode program untuk tombol “**Ekstraksi Ciri**”

```

global Im ekstraksi_fitur
gambar = imresize(Im,[400,400]);
[BW, Img] = remove_bg(gambar);
red=Img(:,:,1);
green=Img(:,:,2);
blue=Img(:,:,3);
gray = rgb2gray(Img);

axes(handles.axes3)
imshow(Img)

axes(handles.axes4)
imshow(gray)

Red = mean(mean(red));
Green = mean(mean(green));
Blue = mean(mean(blue));
Gray = mean(mean(gray));

[M,N] = size(gray);
IFeature = zeros(M,N);
I = gray;
[tinggi, lebar] = size (I);

```

13. Menentukan sudut-sudut GLCM

```

% Bentuk GLCM
GLCM0 = zeros(256, 256);
total_piksel0 =0;

GLCM45 = zeros(256, 256);
total_piksel45 =0;

GLCM90 = zeros(256, 256);
total_piksel90 =0;

GLCM135 = zeros(256, 256);
total_piksel135 =0;

```

f. Hasil dan Pembahasan

Setelah dijalankan maka hasilnya seperti ini

The screenshot shows a software interface for image processing. At the top, there is a file selection bar with 'D:\Sampul\Matematika\Akhir Akhir\data_360x1000.jpg' and a 'Browse' button. Below this is a table with columns: Red, Green, Blue, G0.Asm, G0.Idm, G0.Kontras, G0.Korelasi, G45.Asm, G45.Idm, G45.Kontras, G45.Korelasi, G90.Asm, G90.Idm, G90.Kontras, and G90.Korelasi. The table contains 7 rows of numerical data. To the right of the table is a large image of a red onion. Below the table are two boxes: 'Akurasi' containing the value '92' and 'Keterangan' containing '0 : Bagus' and '1 : Busuk'. At the bottom are 'Proses' and 'Reset' buttons.

	Red	Green	Blue	G0.Asm	G0.Idm	G0.Kontras	G0.Korelasi	G45.Asm	G45.Idm	G45.Kontras	G45.Korelasi	G90.Asm	G90.Idm	G90.Kontras	G90.Korelasi
1	2.4642	1.6036	1.8890	0.9484	0.9790	16.6311	0.0058	0.9479	0.9780	21.1625	0.0057	0.9482	0.9798	17.1743	
2	2.2701	1.5223	1.7071	0.9522	0.9667	15.6686	0.0063	0.9518	0.9799	20.4995	0.0062	0.9521	0.9809	16.7175	
3	2.0069	1.9717	2.1522	0.9445	0.9769	16.4477	0.0048	0.9442	0.9763	19.7255	0.0047	0.9441	0.9779	15.7320	
4	3.1779	2.1181	2.1269	0.9353	0.9733	16.1410	0.0048	0.9349	0.9720	20.3181	0.0047	0.9353	0.9742	14.7888	
5	2.3840	1.4987	1.6341	0.9503	0.9695	12.9102	0.0065	0.9500	0.9797	15.0463	0.0065	0.9504	0.9814	11.1616	
6	2.0412	1.3760	1.3227	0.9574	0.9618	14.0482	0.0074	0.9571	0.9814	16.3731	0.0073	0.9575	0.9831	11.4965	
7	2.3449	1.6486	1.7022	0.9503	0.9789	14.5798	0.0061	0.9500	0.9781	17.9496	0.0061	0.9503	0.9795	13.4994	

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 37

Pokok Bahasan : Studi Kasus (Deteksi Kematangan Buah)

Acara Praktikum/Praktik : Minggu 11/1

Tempat : Politeknik Negeri Jember

Alokasi Waktu : 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami definisi dari pengolahan citra dan vision.
2. Mahasiswa mampu memberikan contoh penerapan pengolahan citra digital pada suatu studi kasus (Deteksi Kematangan Buah)

b. Indikator

Ketepatan dalam menjelaskan mengenai penerapan pengolahan citra digital pada suatu studi kasus (Deteksi Kematangan Buah)

c. Dasar Teori

Indonesia merupakan negara agraris yang mayoritas penghasilannya dari sector pertanian. Pada bidang pertanian dibagi menjadi 3 sub sector yaitu tanaman pangan, hortikultura dan perkebunan. Komoditas tanaman pangan terdiri dari beras, gandum, jagung, kacang tanah, kedelai, kacang hijau, ubi jalar segar, ubi kayu, dll. Komoditas hortikultura terbagi menjadi empat yaitu sayuran (bawang merah, bawang putih, bawang Bombay, kentang, bunga kol, brokoli, kubis dan cabai), buah-buahan (mangga, manggis, jeruk, anggur, papaya, apel, nanas, pir, pisang dan salak), tanaman hias (anggrek) dan tanaman obat (jahe dan temulawak). Komoditas perkebunan yaitu kelapa, karet, kelapa sawit, kopi, teh, lada, tembakau, kakao, cengkeh, kapas, pala, tebu, pinang, kacang mete, dll (BPS, 2019).

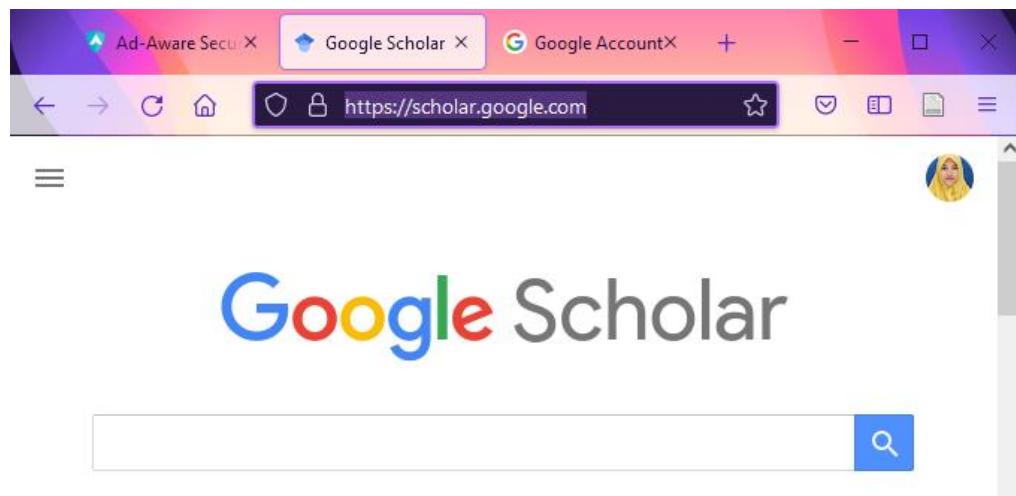
Jumlah produksi pada masing-masing komoditi unggulan bersifat fluktuatif setiap tahunnya, salah satu faktor penyebab menurunnya mutu buah adalah deteksi kematangan buah/sayur. Umumnya deteksi kematangan buah/sayur dilakukan secara manual sehingga penentuan tersebut bersifat subjektif. Untuk itu dibutuhkan sebuah sistem otomatis yang mampu membantu petani deteksi kematangan buah/sayur pada buah/sayuran tersebut sehingga dapat menentukan menjaga mutu dari buah dan sayur tersebut.

d. Alat dan Bahan

1. Laptop
2. Kertas A4
3. Pulpen
4. Website Google Chrome atau Mozilla Firefox

e. Prosedur Kerja

1. Buatlah satu kelompok yang terdiri dari 3 – 4 orang mahasiswa
2. Buka website kemudian ketik <https://scholar.google.com/>
3. Ketikkan kata kunci ‘pengolahan citra untuk deteksi kematangan buah/sayur’



Gambar 57. Tampilan website google scholar

3. Carilah paper terbaru terkait hal tersebut dengan rentang waktu 5 tahun terakhir.
4. Download dan lakukan resume/ringkas jurnal tersebut

f. Hasil dan Pembahasan

1. Dokumentasi Tugas berupa laporan (Resume Jurnal)
2. Dokumentasi Tugas berupa file ppt

g. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan dalam menjelaskan dari tugas ditunjang dengan bukti referensi	1	2	3	4
2	Ketepatan waktu dan ketepatan dalam menjelaskan dari tugas	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat dalam menjelaskan tugas	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam menjelaskan tugas	(1)	2	3	4
Jumlah skor					

Acara 38

Pokok Bahasan : Studi Kasus (Deteksi Kematangan Buah)

Acara Praktikum/Praktik : Minggu 11/2

Tempat : Politeknik Negeri Jember

Alokasi Waktu : 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami definisi dari pengolahan citra dan vision.
2. Mahasiswa mampu memberikan contoh penerapan pengolahan citra digital pada suatu studi kasus (Deteksi Kematangan Buah)

b. Indikator

Ketepatan dalam menjelaskan mengenai penerapan pengolahan citra digital pada suatu studi kasus (Deteksi Kematangan Buah)

c. Dasar Teori

Indonesia merupakan negara agraris yang mayoritas penghasilannya dari sector pertanian. Pada bidang pertanian dibagi menjadi 3 sub sector yaitu tanaman pangan, hortikultura dan perkebunan. Komoditas tanaman pangan terdiri dari beras, gandum, jagung, kacang tanah, kedelai, kacang hijau, ubi jalar segar, ubi kayu, dll. Komoditas hortikultura terbagi menjadi empat yaitu sayuran (bawang merah, bawang putih, bawang Bombay, kentang, bunga kol, brokoli, kubis dan cabai), buah-buahan (mangga, manggis, jeruk, anggur, papaya, apel, nanas, pir, pisang dan salak), tanaman hias (anggrek) dan tanaman obat (jahe dan temulawak). Komoditas perkebunan yaitu kelapa, karet, kelapa sawit, kopi, teh, lada, tembakau, kakao, cengkeh, kapas, pala, tebu, pinang, kacang mete, dll (BPS, 2019).

Jumlah produksi pada masing-masing komoditi unggulan bersifat fluktuatif setiap tahunnya, salah satu faktor penyebab menurunnya mutu buah adalah deteksi kematangan buah/sayur. Umumnya deteksi kematangan buah/sayur dilakukan secara manual sehingga penentuan tersebut bersifat subjektif. Untuk itu dibutuhkan sebuah sistem otomatis yang mampu membantu petani deteksi kematangan buah/sayur pada buah/sayuran tersebut sehingga dapat menentukan menjaga mutu dari buah dan sayur tersebut.

d. Alat dan Bahan

1. Laptop
2. Kertas A4
3. Pulpen
4. Website Google Chrome atau Mozilla Firefox

e. Prosedur Kerja

1. Buatlah satu kelompok yang terdiri dari 3 – 4 orang mahasiswa
2. Diskusikan topik atau obyek workshop yang akan digunakan
3. Cari dan kumpulkan data citra obyek yang akan digunakan
4. Buatlah rencana project workshop dalam bentuk draft makalah
5. Buatlah Bab 1 – Bab 3 yang terdiri dari :
 - a. Bab 1. Pendahuluan (Latarbelakang, Rumusan Masalah, Tujuan dan Manfaat)
 - b. Bab 2. Tinjauan Pustaka (Obyek Project Workshop, Teknik Pengolahan Citra Digital yang rencananya digunakan, Referensi Rujukan/State of the art)
 - c. Bab 3. Metode Pengerjaan (Alat dan bahan, metode pelaksanaan)

f. Hasil dan Pembahasan

1. Dokumentasi Tugas berupa draft makalah
2. Dokumentasi Tugas berupa file ppt

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 39

Pokok Bahasan : Studi Kasus (Deteksi Kematangan Buah)

Acara Praktikum/Praktik : Minggu 11/3

Tempat : Politeknik Negeri Jember

Alokasi Waktu : 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami definisi dari pengolahan citra dan vision.
2. Mahasiswa mampu memberikan contoh penerapan pengolahan citra digital pada suatu studi kasus (Deteksi Kematangan Buah)

b. Indikator

Ketepatan dalam menjelaskan mengenai penerapan pengolahan citra digital pada suatu studi kasus (Deteksi Kematangan Buah)

c. Dasar Teori

Indonesia merupakan negara agraris yang mayoritas penghasilannya dari sector pertanian. Pada bidang pertanian dibagi menjadi 3 sub sector yaitu tanaman pangan, hortikultura dan perkebunan. Komoditas tanaman pangan terdiri dari beras, gandum, jagung, kacang tanah, kedelai, kacang hijau, ubi jalar segar, ubi kayu, dll. Komoditas hortikultura terbagi menjadi empat yaitu sayuran (bawang merah, bawang putih, bawang Bombay, kentang, bunga kol, brokoli, kubis dan cabai), buah-buahan (mangga, manggis, jeruk, anggur, papaya, apel, nanas, pir, pisang dan salak), tanaman hias (anggrek) dan tanaman obat (jahe dan temulawak). Komoditas perkebunan yaitu kelapa, karet, kelapa sawit, kopi, teh, lada, tembakau, kakao, cengkeh, kapas, pala, tebu, pinang, kacang mete, dll (BPS, 2019).

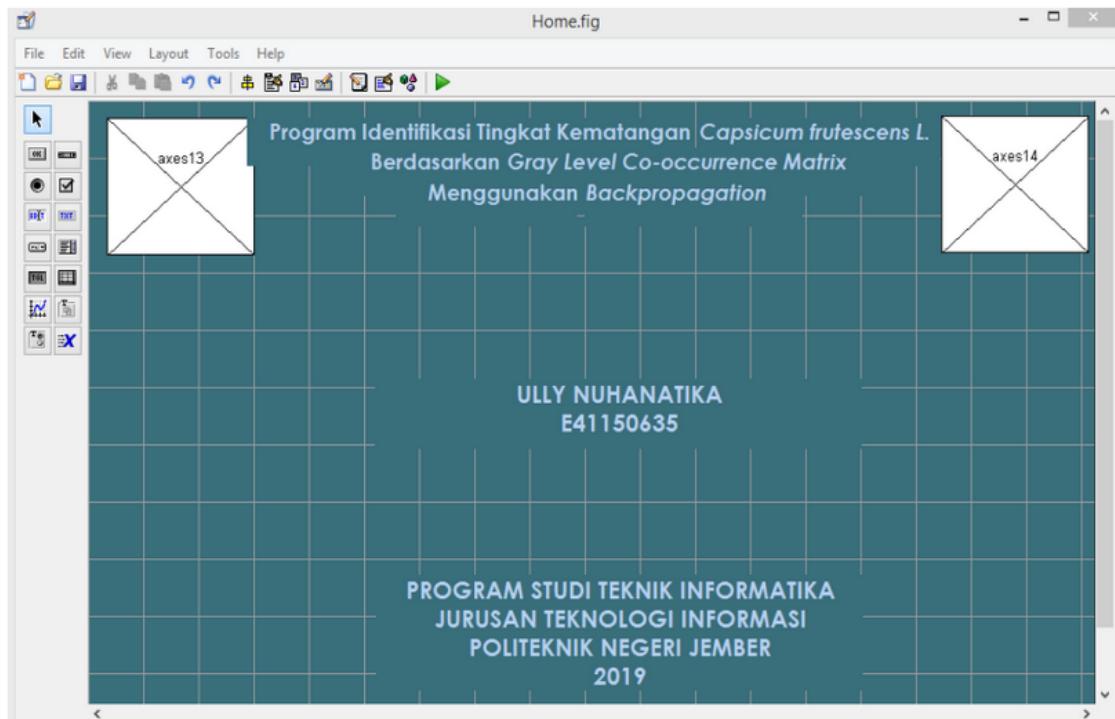
Jumlah produksi pada masing-masing komoditi unggulan bersifat fluktuatif setiap tahunnya, salah satu faktor penyebab menurunnya mutu buah adalah deteksi kematangan buah/sayur. Umumnya deteksi kematangan buah/sayur dilakukan secara manual sehingga penentuan tersebut bersifat subjektif. Untuk itu dibutuhkan sebuah sistem otomatis yang mampu membantu petani deteksi kematangan buah/sayur pada buah/sayuran tersebut sehingga dapat menentukan menjaga mutu dari buah dan sayur tersebut.

d. Alat dan Bahan

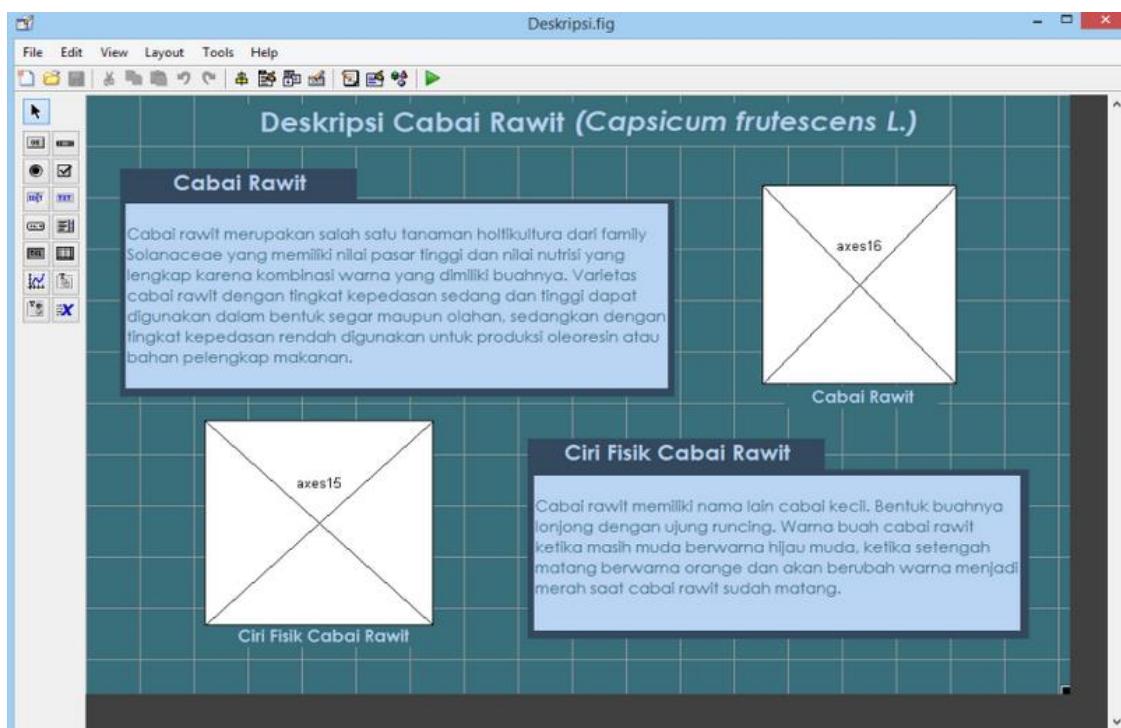
1. Laptop
2. Kertas A4
3. Pulpen
4. Matlab atau Python

e. Prosedur Kerja

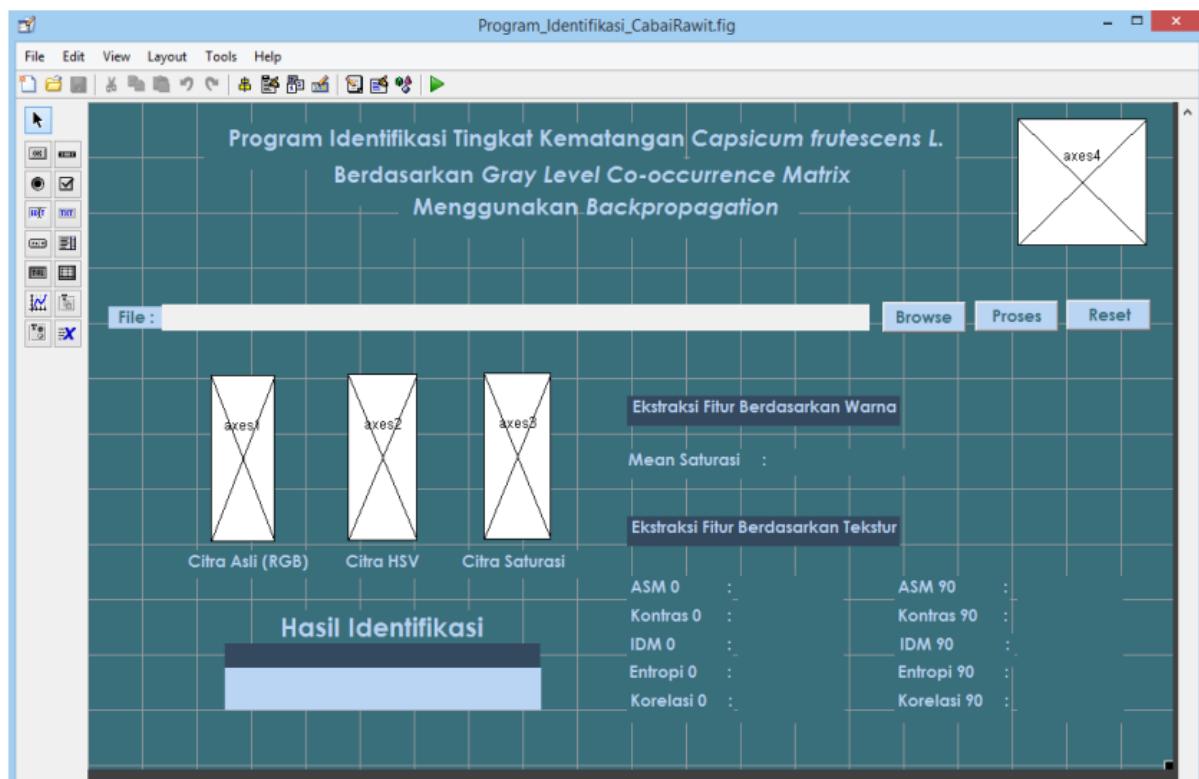
1. Buatlah satu kelompok yang terdiri dari 3 – 4 orang mahasiswa
2. Buatlah mock up design aplikasi yang kelompok anda rencanakan, contohnya mock up design menggunakan aplikasi matlab :



Gambar 58. Mock up halaman beranda



Gambar 59. Mock up halaman informasi terkait ciri fisik sayuran



Gambar 60. Mock up identifikasi penyakit daun pada tanaman mentimun

f. Hasil dan Pembahasan

Dokumentasi Tugas berupa mock up design aplikasi

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 40

Pokok Bahasan : Studi Kasus (Deteksi Kematangan Buah)

Acara Praktikum/Praktik : Minggu 11/4

Tempat : Politeknik Negeri Jember

Alokasi Waktu : 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami definisi dari pengolahan citra dan vision.
2. Mahasiswa mampu memberikan contoh penerapan pengolahan citra digital pada suatu studi kasus (Deteksi Kematangan Buah)

b. Indikator

Ketepatan dalam menjelaskan mengenai penerapan pengolahan citra digital pada suatu studi kasus (Deteksi Kematangan Buah)

c. Dasar Teori

Indonesia merupakan negara agraris yang mayoritas penghasilannya dari sector pertanian. Pada bidang pertanian dibagi menjadi 3 sub sector yaitu tanaman pangan, hortikultura dan perkebunan. Komoditas tanaman pangan terdiri dari beras, gandum, jagung, kacang tanah, kedelai, kacang hijau, ubi jalar segar, ubi kayu, dll. Komoditas hortikultura terbagi menjadi empat yaitu sayuran (bawang merah, bawang putih, bawang Bombay, kentang, bunga kol, brokoli, kubis dan cabai), buah-buahan (mangga, manggis, jeruk, anggur, papaya, apel, nanas, pir, pisang dan salak), tanaman hias (anggrek) dan tanaman obat (jahe dan temulawak). Komoditas perkebunan yaitu kelapa, karet, kelapa sawit, kopi, teh, lada, tembakau, kakao, cengkeh, kapas, pala, tebu, pinang, kacang mete, dll (BPS, 2019).

Jumlah produksi pada masing-masing komoditi unggulan bersifat fluktuatif setiap tahunnya, salah satu faktor penyebab menurunnya mutu buah adalah deteksi kematangan buah/sayur. Umumnya deteksi kematangan buah/sayur dilakukan secara manual sehingga penentuan tersebut bersifat subjektif. Untuk itu dibutuhkan sebuah sistem otomatis yang mampu membantu petani deteksi kematangan buah/sayur pada buah/sayuran tersebut sehingga dapat menentukan menjaga mutu dari buah dan sayur tersebut.

d. Alat dan Bahan

1. Laptop
2. Kertas A4
3. Pulpen
4. Matlab atau Python
5. Powerpoint

1. Prosedur Kerja

1. Buatlah satu kelompok yang terdiri dari 3 – 4 orang mahasiswa
2. Membuat aplikasi terkait Deteksi Kematangan Buah

2. Hasil dan Pembahasan

1. Dokumentasi Tugas berupa makalah
2. Dokumentasi Tugas berupa file ppt
3. Dokumentasi Tugas berupa aplikasi yang telah dibuat

3. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 41

Pokok Bahasan : Studi Kasus (Deteksi Penyakit/Hama Tanaman)

Acara Praktikum/Praktik : Minggu 12/1

Tempat : Politeknik Negeri Jember

Alokasi Waktu : 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami definisi dari pengolahan citra dan vision.
2. Mahasiswa mampu memberikan contoh penerapan pengolahan citra digital pada suatu studi kasus (deteksi penyakit/hama tanaman)

b. Indikator

Ketepatan dalam menjelaskan mengenai penerapan pengolahan citra digital pada suatu studi kasus (deteksi penyakit/hama tanaman)

c. Dasar Teori

Indonesia merupakan negara agraris yang mayoritas penghasilannya dari sector pertanian. Pada bidang pertanian dibagi menjadi 3 sub sector yaitu tanaman pangan, hortikultura dan perkebunan. Komoditas tanaman pangan terdiri dari beras, gandum, jagung, kacang tanah, kedelai, kacang hijau, ubi jalar segar, ubi kayu, dll. Komoditas hortikultura terbagi menjadi empat yaitu sayuran (bawang merah, bawang putih, bawang Bombay, kentang, bunga kol, brokoli, kubis dan cabai), buah-buahan (mangga, manggis, jeruk, anggur, papaya, apel, nanas, pir, pisang dan salak), tanaman hias (anggrek) dan tanaman obat (jahe dan temulawak). Komoditas perkebunan yaitu kelapa, karet, kelapa sawit, kopi, teh, lada, tembakau, kakao, cengkeh, kapas, pala, tebu, pinang, kacang mete, dll (BPS, 2019).

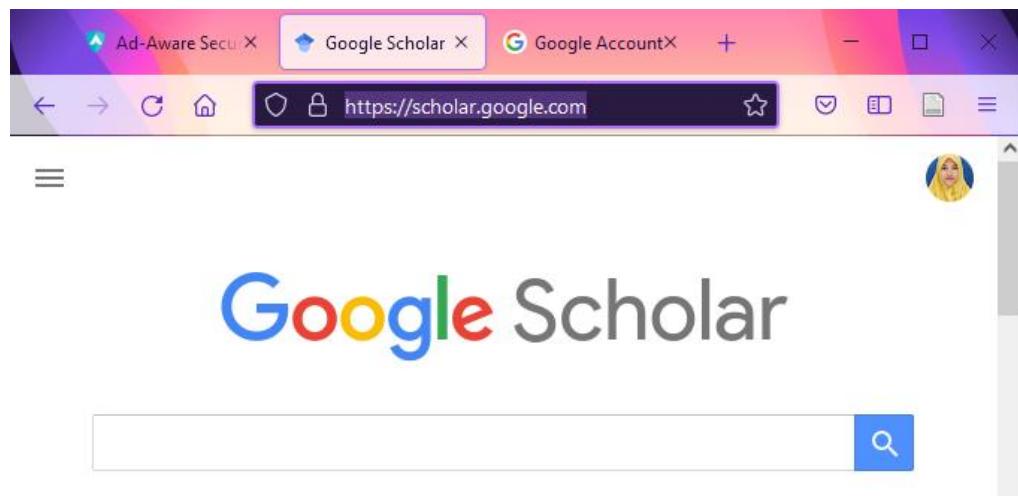
Jumlah produksi pada masing-masing komoditi unggulan bersifat fluktuatif setiap tahunnya, salah satu faktor penyebabnya adalah serangan hama dan penyakit. Gejala penyakit pada tanaman dapat dilihat dari bagian tubuh tanaman, seperti daun, buah, batang dan akar. Umumnya identifikasi hama dan penyakit pada tanaman dilakukan secara manual berdasarkan pengalaman dari petani sehingga penentuan tersebut bersifat subjektif. Untuk itu dibutuhkan sebuah sistem otomatis yang mampu membantu petani mengidentifikasi jenis penyakit dan hama pada tanaman tersebut sehingga dapat menentukan solusi yang tepat.

d. Alat dan Bahan

1. Laptop
2. Kertas A4
3. Pulpen
4. Website Google Chrome atau Mozilla Firefox

e. Prosedur Kerja

1. Buatlah satu kelompok yang terdiri dari 3 – 4 orang mahasiswa
2. Buka website kemudian ketik <https://scholar.google.com/>
3. Ketikkan kata kunci ‘pengolahan citra untuk identifikasi penyakit tanaman’



Gambar 61. Tampilan website google scholar

3. Carilah paper terbaru terkait hal tersebut dengan rentang waktu 5 tahun terakhir.
4. Download dan lakukan resume/ringkas jurnal tersebut

f. Hasil dan Pembahasan

1. Dokumentasi Tugas berupa laporan (Resume Jurnal)
2. Dokumentasi Tugas berupa file ppt

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 42

Pokok Bahasan : Studi Kasus (Deteksi Penyakit/Hama Tanaman)

Acara Praktikum/Praktik : Minggu 12/2

Tempat : Politeknik Negeri Jember

Alokasi Waktu : 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami definisi dari pengolahan citra dan vision.
2. Mahasiswa mampu memberikan contoh penerapan pengolahan citra digital pada suatu studi kasus (deteksi penyakit/hama tanaman)

b. Indikator

Ketepatan dalam menjelaskan mengenai penerapan pengolahan citra digital pada suatu studi kasus (deteksi penyakit/hama tanaman)

c. Dasar Teori

Indonesia merupakan negara agraris yang mayoritas penghasilannya dari sector pertanian. Pada bidang pertanian dibagi menjadi 3 sub sector yaitu tanaman pangan, hortikultura dan perkebunan. Komoditas tanaman pangan terdiri dari beras, gandum, jagung, kacang tanah, kedelai, kacang hijau, ubi jalar segar, ubi kayu, dll. Komoditas hortikultura terbagi menjadi empat yaitu sayuran (bawang merah, bawang putih, bawang Bombay, kentang, bunga kol, brokoli, kubis dan cabai), buah-buahan (mangga, manggis, jeruk, anggur, papaya, apel, nanas, pir, pisang dan salak), tanaman hias (anggrek) dan tanaman obat (jahe dan temulawak). Komoditas perkebunan yaitu kelapa, karet, kelapa sawit, kopi, teh, lada, tembakau, kakao, cengkeh, kapas, pala, tebu, pinang, kacang mete, dll (BPS, 2019).

Jumlah produksi pada masing-masing komoditi unggulan bersifat fluktuatif setiap tahunnya, salah satu faktor penyebabnya adalah serangan hama dan penyakit. Gejala penyakit pada tanaman dapat dilihat dari bagian tubuh tanaman, seperti daun, buah, batang dan akar. Umumnya identifikasi hama dan penyakit pada tanaman dilakukan secara manual berdasarkan pengalaman dari petani sehingga penentuan tersebut bersifat subjektif. Untuk itu dibutuhkan sebuah sistem otomatis yang mampu membantu petani mengidentifikasi jenis penyakit dan hama pada tanaman tersebut sehingga dapat menentukan solusi yang tepat.

d. Alat dan Bahan

1. Laptop
2. Kertas A4
3. Pulpen
4. Website Google Chrome atau Mozilla Firefox

e. Prosedur Kerja

1. Buatlah satu kelompok yang terdiri dari 3 – 4 orang mahasiswa
2. Diskusikan topik atau obyek workshop yang akan digunakan
3. Cari dan kumpulkan data citra obyek yang akan digunakan
4. Buatlah rencana project workshop dalam bentuk draft makalah
5. Buatlah Bab 1 – Bab 3 yang terdiri dari :
 - a. Bab 1. Pendahuluan (Latarbelakang, Rumusan Masalah, Tujuan dan Manfaat)
 - b. Bab 2. Tinjauan Pustaka (Obyek Project Workshop, Teknik Pengolahan Citra Digital yang rencananya digunakan, Referensi Rujukan/State of the art)
 - c. Bab 3. Metode Pengerjaan (Alat dan bahan, metode pelaksanaan)

f. Hasil dan Pembahasan

1. Dokumentasi Tugas berupa draft makalah
2. Dokumentasi Tugas berupa file ppt

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 43

Pokok Bahasan : Studi Kasus (Deteksi Penyakit/Hama Tanaman)

Acara Praktikum/Praktik : Minggu 12/3

Tempat : Politeknik Negeri Jember

Alokasi Waktu : 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami definisi dari pengolahan citra dan vision.
2. Mahasiswa mampu memberikan contoh penerapan pengolahan citra digital pada suatu studi kasus (deteksi penyakit/hama tanaman)

b. Indikator

Ketepatan dalam menjelaskan mengenai penerapan pengolahan citra digital pada suatu studi kasus (deteksi penyakit/hama tanaman)

c. Dasar Teori

Indonesia merupakan negara agraris yang mayoritas penghasilannya dari sector pertanian. Pada bidang pertanian dibagi menjadi 3 sub sector yaitu tanaman pangan, hortikultura dan perkebunan. Komoditas tanaman pangan terdiri dari beras, gandum, jagung, kacang tanah, kedelai, kacang hijau, ubi jalar segar, ubi kayu, dll. Komoditas hortikultura terbagi menjadi empat yaitu sayuran (bawang merah, bawang putih, bawang Bombay, kentang, bunga kol, brokoli, kubis dan cabai), buah-buahan (mangga, manggis, jeruk, anggur, papaya, apel, nanas, pir, pisang dan salak), tanaman hias (anggrek) dan tanaman obat (jahe dan temulawak). Komoditas perkebunan yaitu kelapa, karet, kelapa sawit, kopi, teh, lada, tembakau, kakao, cengkeh, kapas, pala, tebu, pinang, kacang mete, dll (BPS, 2019).

Jumlah produksi pada masing-masing komoditi unggulan bersifat fluktuatif setiap tahunnya, salah satu faktor penyebabnya adalah serangan hama dan penyakit. Gejala penyakit pada tanaman dapat dilihat dari bagian tubuh tanaman, seperti daun, buah, batang dan akar. Umumnya identifikasi hama dan penyakit pada tanaman dilakukan secara manual berdasarkan pengalaman dari petani sehingga penentuan tersebut bersifat subjektif. Untuk itu dibutuhkan sebuah sistem otomatis yang mampu membantu petani mengidentifikasi jenis penyakit dan hama pada tanaman tersebut sehingga dapat menentukan solusi yang tepat.

d. Alat dan Bahan

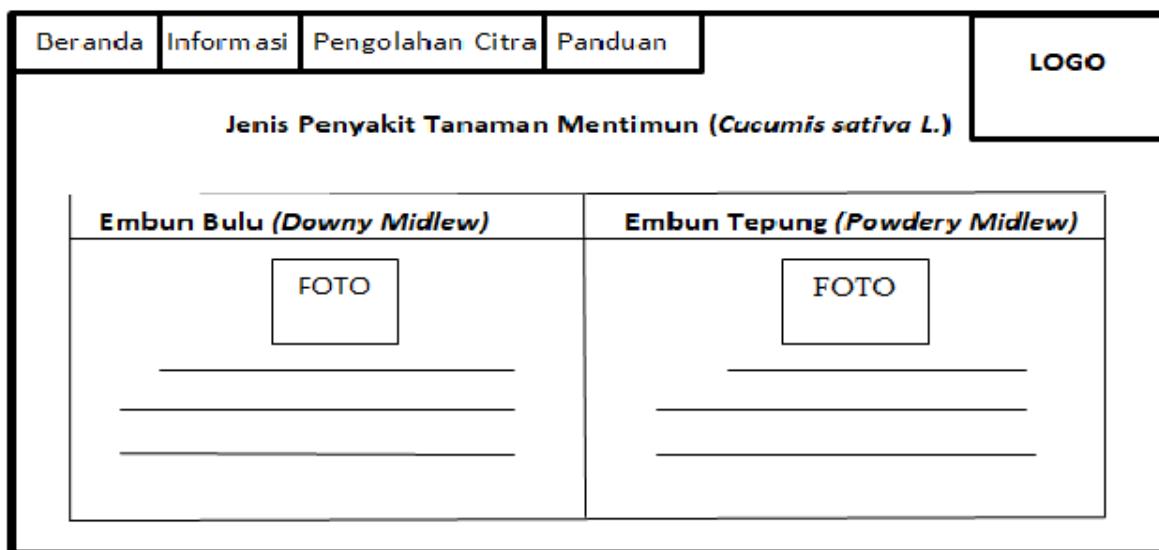
1. Laptop
2. Kertas A4
3. Pulpen
4. Matlab atau Python

e. Prosedur Kerja

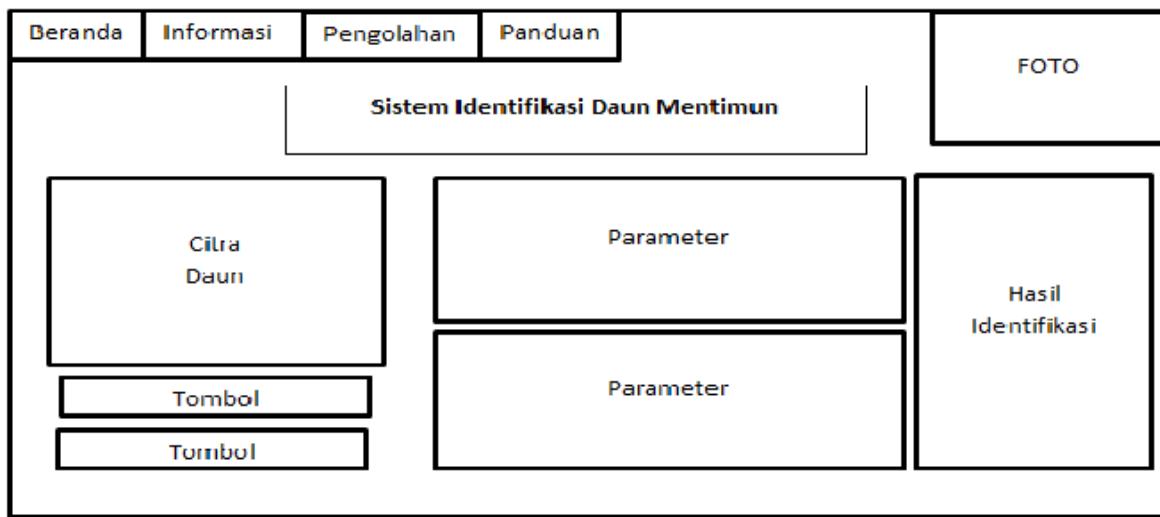
1. Buatlah satu kelompok yang terdiri dari 3 – 4 orang mahasiswa
2. Buatlah mock up design aplikasi yang kelompok anda rencanakan, contohnya mock up design menggunakan aplikasi matlab :



Gambar 62. Mock up halaman beranda



Gambar 63. Mock up halaman informasi terkait jenis penyakit



Gambar 64. Mock up identifikasi penyakit daun pada tanaman mentimun

f. Hasil dan Pembahasan

Dokumentasi Tugas berupa mock up design aplikasi

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 44

Pokok Bahasan : Studi Kasus (Deteksi Penyakit/Hama Tanaman)

Acara Praktikum/Praktik : Minggu 12/4

Tempat : Politeknik Negeri Jember

Alokasi Waktu : 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami definisi dari pengolahan citra dan vision.
2. Mahasiswa mampu memberikan contoh penerapan pengolahan citra digital pada suatu studi kasus (deteksi penyakit/hama tanaman)

b. Indikator

Ketepatan dalam menjelaskan mengenai penerapan pengolahan citra digital pada suatu studi kasus (deteksi penyakit/hama tanaman)

c. Dasar Teori

Indonesia merupakan negara agraris yang mayoritas penghasilannya dari sector pertanian. Pada bidang pertanian dibagi menjadi 3 sub sector yaitu tanaman pangan, hortikultura dan perkebunan. Komoditas tanaman pangan terdiri dari beras, gandum, jagung, kacang tanah, kedelai, kacang hijau, ubi jalar segar, ubi kayu, dll. Komoditas hortikultura terbagi menjadi empat yaitu sayuran (bawang merah, bawang putih, bawang Bombay, kentang, bunga kol, brokoli, kubis dan cabai), buah-buahan (mangga, manggis, jeruk, anggur, papaya, apel, nanas, pir, pisang dan salak), tanaman hias (anggrek) dan tanaman obat (jahe dan temulawak). Komoditas perkebunan yaitu kelapa, karet, kelapa sawit, kopi, teh, lada, tembakau, kakao, cengkeh, kapas, pala, tebu, pinang, kacang mete, dll (BPS, 2019).

Jumlah produksi pada masing-masing komoditi unggulan bersifat fluktuatif setiap tahunnya, salah satu faktor penyebabnya adalah serangan hama dan penyakit. Gejala penyakit pada tanaman dapat dilihat dari bagian tubuh tanaman, seperti daun, buah, batang dan akar. Umumnya identifikasi hama dan penyakit pada tanaman dilakukan secara manual berdasarkan pengalaman dari petani sehingga penentuan tersebut bersifat subjektif. Untuk itu dibutuhkan sebuah sistem otomatis yang mampu membantu petani mengidentifikasi jenis penyakit dan hama pada tanaman tersebut sehingga dapat menentukan solusi yang tepat.

d. Alat dan Bahan

1. Laptop
2. Kertas A4
3. Pulpen
4. Matlab atau Python

5. Power Point

e. Prosedur Kerja

1. Buatlah satu kelompok yang terdiri dari 3 – 4 orang mahasiswa
2. Membuat aplikasi terkait deteksi penyakit/hama tanaman

f. Hasil dan Pembahasan

1. Dokumentasi Tugas berupa makalah
2. Dokumentasi Tugas berupa file ppt
3. Dokumentasi Tugas berupa aplikasi yang telah dibuat

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 45-48

Pokok Bahasan : *Computer Vision untuk Pengenalan Batik*

Acara Praktikum/Praktik : Minggu 13/1-4

Tempat : Laboratorium Multimedia Cerdas

Alokasi Waktu : 4 x 100 Menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

- 1) Mahasiswa mampu menjelaskan konsep pengenalan motif batik.
- 2) Mahasiswa mampu menerapkan teknik ekstraksi fitur citra batik dan algoritma *random forest* untuk klasifikasi.

b. Indikator

Kemampuan mahasiswa dalam menjelaskan dan mengimplementasikan teknik ekstraksi fitur citra batik dan algoritma *random forest* untuk klasifikasi.

c. Dasar Teori

Random Forest merupakan sebuah metode *ensemble* dimana metode ini merupakan salah satu cara mengkombinasikan metode klasifikasi untuk meningkatkan akurasi dari sebuah metode klasifikasi. Tahapan yang dilakukan untuk metode *random forest* diawali dengan teknik dasar *data mining* yaitu *decision tree*. *Random Forest* terdiri dari sekumpulan *decision tree*, dimana kumpulan *decision tree* tersebut merupakan data yang digunakan dalam pengklasifikasian suatu kelas. Metode *random forest* memerlukan *data training* untuk melakukan pengklasifikasian. Maka dari itu *Random Forest* juga dapat disebut sebagai metode yang *supervised learning*. Sesuai seperti namanya (*random forest*), metode ini menciptakan sebuah hutan (*forest*) dari kumpulan pohon-pohon (*tree*) didalamnya. Dengan kata lain, semakin banyak pohon (*tree*) yang terdapat dalam hutan (*forest*) tersebut, maka akan semakin kuat hutan tersebut. Pada konteks metode ini, semakin banyak jumlah *tree* yang didapat dari pengklasifikasian (*decision tree*), maka semakin besar pula akurasi yang bisa didapatkan. Berikut merupakan tahapan dari algoritma/metode *Random Forest* untuk proses klasifikasi citra.

1. Menentukan estimasi jumlah perulangan dalam pembuatan pohon keputusan (*tree*).
2. Penentuan pembagian dataset sebanyak jumlah estimasi perulangan dari *data training*. Dataset diambil secara *random* dalam lingkup *data training*.
3. Penghitungan *Information Gain* dari setiap atribut.
4. Pemilihan atribut yang memiliki nilai *information gain* terbesar.
5. Pembentukan simpul yang berisi atribut tersebut.

6. Ulangi proses perhitungan *information gain* sampai semua data telah termasuk dalam kelas yang sama. Atribut yang telah dipilih tidak diikutkan lagi dalam perhitungan nilai *information gain*.
7. Saat pohon keputusan telah terbuat, maka ulangi langkah ke-dua sampai langkah ke-enam sampai jumlah n-estimasi terpenuhi.
8. Setelah pohon keputusan sejumlah n-estimasi telah terbuat, maka dilakukan *majority vote / voting* suara target kelas terbanyak.
9. Jumlah target kelas terbanyak merupakan hasil akhir dari klasifikasi *random forest* yang dilakukan.

d. Alat dan Bahan

1. Laptop, Kertas A4 & Pulpen
2. PyCharm & Python Version

e. Prosedur Kerja

1. Pada praktikum ini diberikan contoh aplikasi computer vision untuk mengenali motif batik. Fitur yang digunakan adalah fitur tekstur dengan metode ekstraksi GLCM (telah di bahas sebelumnya).

CATATAN :

Didalam BKPM 14 ini menggunakan contoh 32 fitur GLCM citra yang terdiri dari : Contrast, Dissimilarity, Homogeneity, ASM, Energy, Correlation, IDM, Entropy. Masing-masing dilihat dari sudut 0^0 , 45^0 , 90^0 serta 135^0 .

Mahasiswa diperbolehkan hanya mengambil **12 Fitur** saja,
Misal 3 fitur GLCM dengan 4 sudut tersebut

2. Buat **Project** baru di **Pycharm** Anda, lalu siapkan 1 folder berisi data citra yang didalamnya terdiri dari beberapa folder terpisah dari Batik Banyuwangi, Batik Jember, Batik Bondowoso dan Batik Situbondo. Data citra batik bisa diunduh pada tautan berikut : <https://bit.ly/dataBatik>
3. Setelah membuat project baru, import beberapa library di **main.py** yang akan digunakan dalam "**Aplikasi Pengenalan Batik**", seperti gambar dibawah :

```
from xlsxwriter.workbook import Workbook
import mysql.connector
from datetime import datetime
```

```
import tkinter
from tkinter import *
from tkinter import filedialog
from tkinter import messagebox as msgbox
from tkinter import ttk
```

```

from skimage.feature import greycoprops
from skimage import io, color, img_as_ubyte
import pandas as pd

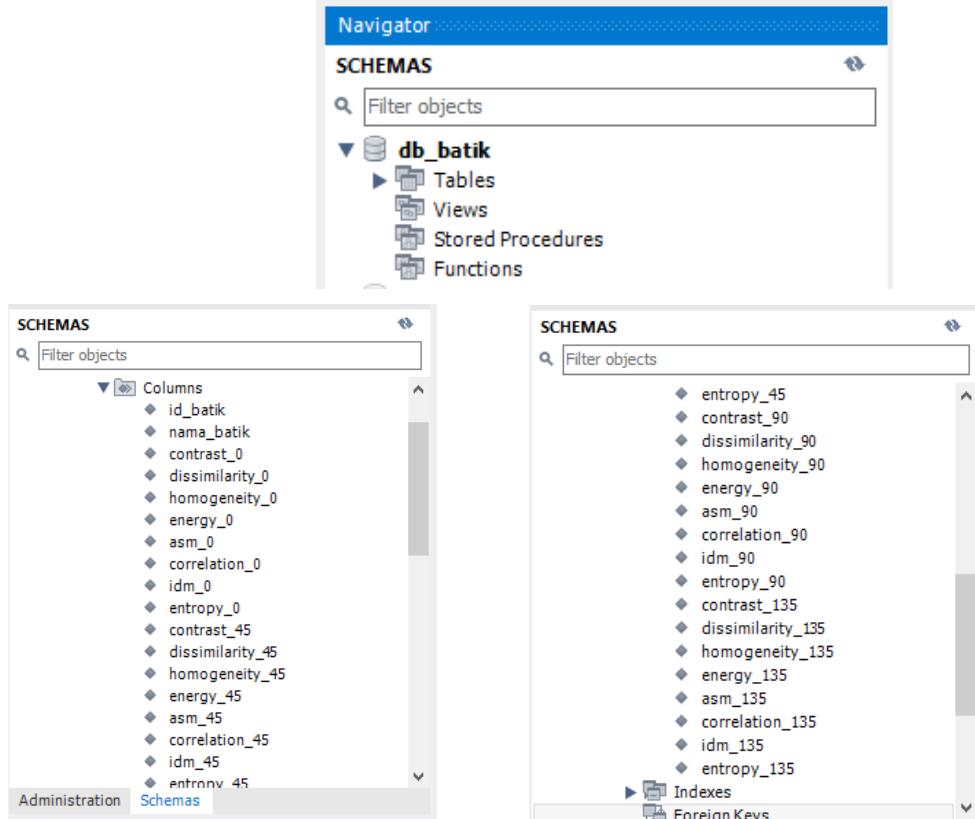
from PIL import Image
from PIL import ImageTk
import numpy as np
from cv2 import *
from sklearn.ensemble import RandomForestClassifier

```

CATATAN :

Jika Interpreter **pandas** dan **sklearn** belum ditambahkan,
silahkan menambahkan seperti dalam langkah BKPM 13
File Project → Settings → Python Interpreter
Install Package **pandas** dan **sklearn**

4. Aplikasi ini menggunakan **MySQL Workbench** sebagai databasenya. Maka dari itu, sebelumnya buatlah database di MySQL Workbench. Contohnya **db_batik** dengan 2 tabel didalamnya, yaitu **tbl_data_testing** dan **tbl_data_training**. Serta buatlah beberapa atribut di tiap tabel seperti pada gambar dibawah.



5. Dalam file yang sama seperti sebelumnya di **Pycharm**, tuliskan kode untuk menghubungkan antara **Database** dengan **Sistem**. Sesuaikan **host**, **user**, **password** dan **nama database** sesuai dengan milik Anda.

```
#####
host = 'localhost'
user = 'root'
passwd = 'Jember311098'
database = 'db_batik'

# Connect to MySql Database
mydb = mysql.connector.connect(
    host=host,
    user=user,
    passwd=passwd,
    database=database,
)

# Create a cursor
my_cursor = mydb.cursor()
```

6. Dalam membuat “**Aplikasi Pengenalan Batik**” ini memerlukan interface sebagai jembatan penghubung antara user dengan sistem. Beberapa tab dalam aplikasi ini pembuatannya opsional tergantung dengan kebutuhan masing-masing. Buatlah beberapa tampilan UI berikut :

a) Tampilan Home

Tampilan Home bersifat opsional, dapat digunakan sebagai landing page aplikasi. Untuk membuat tampilan seperti dibawah, maka dapat menggunakan code seperti gambar dibawah ini.



```
#####
Tkinter Design #####
root = Tk()
root.title("Aplikasi Pengenalan Motif Citra Batik")
# root.iconbitmap()
root.resizable(True, True) # root.resizeable(Width, Height)
root.columnconfigure(0, weight=1)
root.rowconfigure(0, weight=1)

# Centered Width and Height of Apps in Monitor
app_width = 1200
app_height = 650
screen_width = root.winfo_screenwidth()
screen_height = root.winfo_screenheight()
x = (screen_width / 2) - (app_width / 2)
y = (screen_height / 2) - (app_height / 2)
root.geometry(f"{app_width}x{app_height}+{int(x)}+{int(y) - 50}")

# Create Menu
my_menu = Menu(root)
```

```
# Create Tabs=Notebook
my_tabs = ttk.Notebook(root)
my_tabs.pack()

# Create a Frame
my_home_frame = Frame(my_tabs, width=app_width, height=app_height)
my_training_frame = Frame(my_tabs, width=app_width, height=app_height)
my_testing_frame = Frame(my_tabs, width=app_width, height=app_height)
my_about_frame = Frame(my_tabs, width=app_width, height=app_height)

my_home_frame.pack(fill="both", expand=1)
my_training_frame.pack(fill="both", expand=1)
my_testing_frame.pack(fill="both", expand=1)
my_about_frame.pack(fill="both", expand=1)

my_tabs.add(my_home_frame, text="Dashboard")
my_tabs.add(my_training_frame, text="Data Training")
my_tabs.add(my_testing_frame, text="Data Testing")
my_tabs.add(my_about_frame, text="About")
```

```
##### Home Frame Component #####
# Image Background
bg_1 = PhotoImage(file="image/bgbatik1.png")
bg_root = Label(my_home_frame, image=bg_1)
bg_root.place(x=0, y=0, relwidth=1, relheight=1)

...
my_canvas = Canvas(my_home_frame, width=app_width, height=app_height)
my_canvas.pack(fill="both", expand=True)

# Add a Image
my_canvas.create_image(0, 0, image=bg_1, anchor="nw")

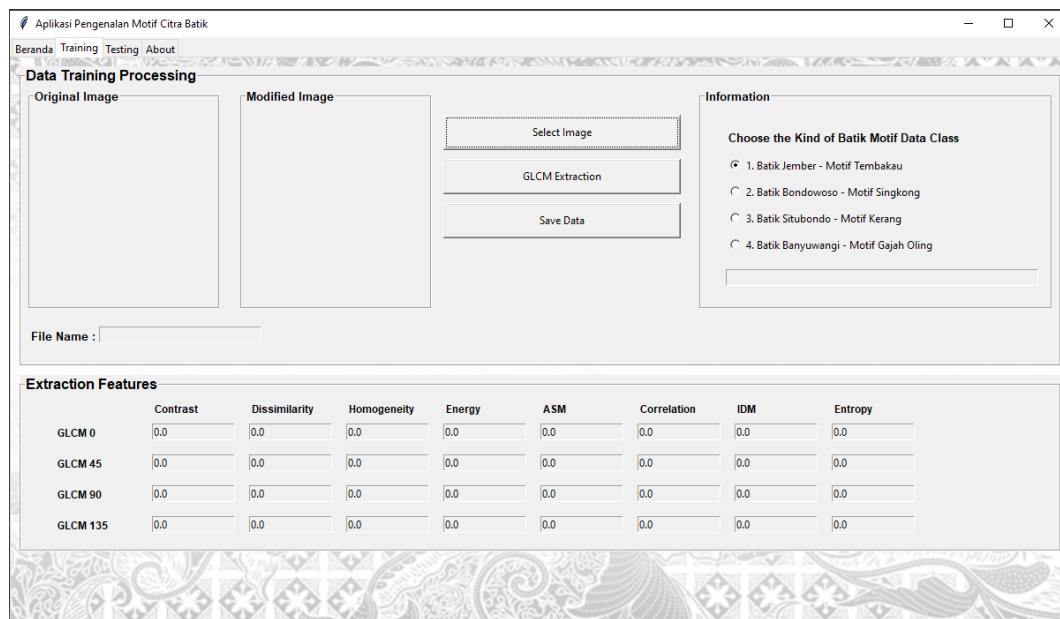
# Add a Label
my_canvas.create_text(app_width / 2, (app_height / 2) - 100, text="Aplikasi Pengenalan Batik", font="Segoi 36 bold")
my_canvas.create_text(app_width / 2, (app_height / 2) - 40, text="(Batik Recognition Application)", font="Segoi 40 bold")
my_canvas.create_text(app_width / 2, (app_height / 2) + 30, text="Rakhmat Fadilah", font="Calibri 16")
my_canvas.create_text(app_width / 2, (app_height / 2) + 55, text="Teknik Informatika, Teknologi Informasi, Politeknik Negeri Jember", font="Calibri 16")
```

Setelah menuliskan kode diatas, jangan lupa untuk menuliskan kode berikut pada baris paling bawah kode Anda.

```
root.mainloop()
```

b) Tampilan Data Training

Tampilan Data Training digunakan untuk beberapa proses seperti Pengambilan Image Batik, Ekstraksi Fitur GLCM dan Menyimpan Data Hasil Ekstraksi Citra.



Untuk membuat tampilan seperti diatas, maka dapat menggunakan code seperti gambar dibawah ini.

BKPM - (NAMA BPKM)

```
# Image Background
bg_2 = PhotoImage(file="image/bgbatik1.png")
bg_root = Label(my_training_frame, image=bg_2)
bg_root.place(x=0, y=0, relwidth=1, relheight=1)

title_frame = LabelFrame(my_training_frame, text="Data Training Processing", font="Segoi 12 bold")
title_frame.place(width=1180, height=340, x=10, y=10)

original_image_frame = LabelFrame(my_training_frame, text="Original Image", font="Segoi 10 bold")
original_image_frame.place(width=217, height=250, x=20, y=35)

original_image_training = Label(my_training_frame)
original_image_training.place(x=25, y=70)

txt_modified_image = LabelFrame(my_training_frame, text="Modified Image", font="Segoi 10 bold")
txt_modified_image.place(width=217, height=250, x=260, y=35)

modified_image_training = Label(my_training_frame)
modified_image_training.place(x=265, y=70)

txt_classification = LabelFrame(my_training_frame, text="Information", font="Segoi 10 bold")
txt_classification.place(width=400, height=250, x=780, y=35)
```

```
# Button
btnSelectImageTraining = Button(my_training_frame, text="Select Image", width=37, command=select_image)
btnSelectImageTraining.place(x=490, y=65, height=40)

btnGrayscale = Button(my_training_frame, text="GLCM Extraction", width=37, command=grayscale)
btnGrayscale.place(x=490, y=115, height=40)

btnAddData = Button(my_training_frame, text="Save Data", width=37, command=add_data)
btnAddData.place(x=490, y=165, height=40)

# Create Extraction Label Frame
extraction_training_frame = LabelFrame(my_training_frame, text="Extraction Features", font="Segoi 12 bold")
extraction_training_frame.place(width=1180, height=200, x=10, y=360)
```

```
## LABEL FOR ENTRY BOXES ##

label_contrast = Label(my_training_frame, text="Contrast", font="Segoi 9 bold")
label_contrast.place(x=160, y=388) # 10, 415

label_dissimilarity_value = Label(my_training_frame, text="Dissimilarity", font="Segoi 9 bold")
label_dissimilarity_value.place(x=270, y=388) # 450
```

Untuk membuat label, dapat menggunakan code seperti pada contoh gambar diatas. Untuk membuat Tampilan Data Training seperti pada gambar diatas, buatlah beberapa label seperti **Contrast, Dissimilarity, Homogeneity, ASM, Energy, Correlation, IDM, Entropy, GLCM 0, GLCM 45, GLCM 90 dan GLCM 135.**

```
radiobutton1 = Radiobutton(my_training_frame, text="1. Batik Jember - Motif Tembakau", variable=r,
                           value="Jember", command=radio_button)
radiobutton1.place(x=810, y=110)
```

Untuk membuat **Radio Button**, dapat menggunakan code seperti pada contoh gambar diatas. Untuk membuat Tampilan Data Training seperti pada gambar diatas, buatlah beberapa radio button seperti Batik Jember Motif Tembakau, Batik Bondowoso Motif Singkong, Batik Situbondo Motif Kerang dan Batik Banyuwangi Motif Gajah Oling.

```
txt_radio_text = tkinter.StringVar()
txt_radio = Entry(my_training_frame, width=50, state="readonly", textvariable=txt_radio_text, font="Segoeui 9 bold",
                  justify="center").place(x=810, y=240)

# Text Box with GLCM 0 Value

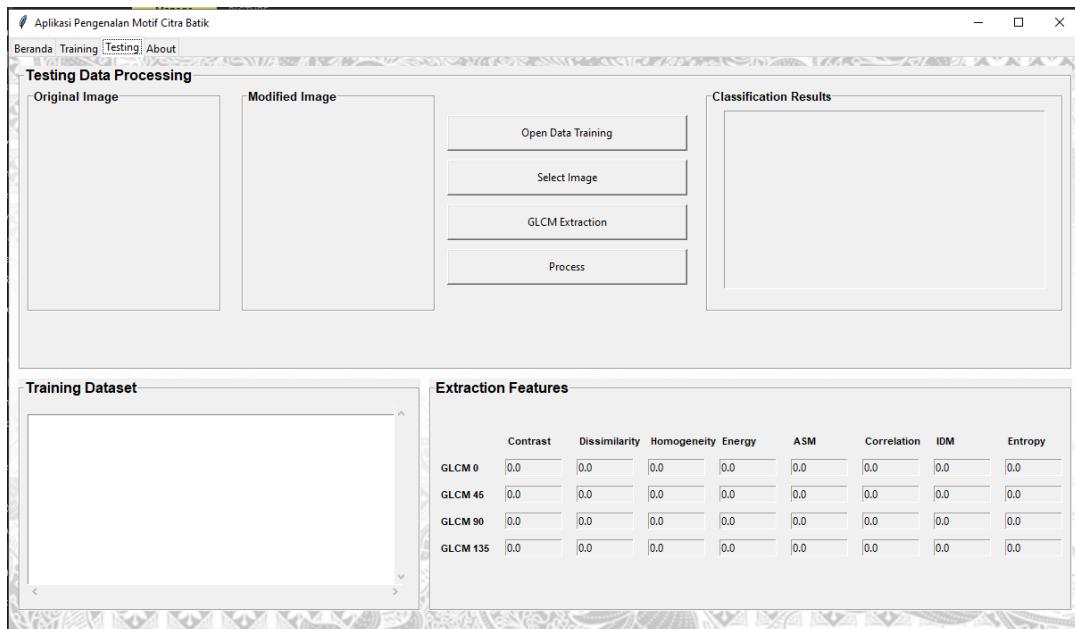
contrast_0_value = DoubleVar()
txt_contrast_0_value = Entry(my_training_frame, width=15, textvariable=contrast_0_value, state="readonly")
txt_contrast_0_value.place(x=160, y=415)

dissimilarity_0_value = DoubleVar()
txt_dissimilarity_0_value = Entry(my_training_frame, width=15, textvariable=dissimilarity_0_value, state="readonly")
txt_dissimilarity_0_value.place(x=270, y=415)
```

Untuk membuat **Entry Box**, dapat menggunakan code seperti pada contoh gambar diatas.

c) Tampilan Data Testing

Tampilan Data Testing digunakan untuk beberapa proses seperti Membuka Database Data Training, Pengambilan Image Batik, Ekstraksi Fitur GLCM dan Proses Klasifikasi Batik.



Untuk membuat tampilan seperti diatas, maka dapat menggunakan code seperti gambar dibawah ini.

```
##### Testing Frame Component #####
# Image Background
bg_3 = PhotoImage(file="image/bqbatik1.png")
bg_root = Label(my_testing_frame, image=bg_3)
bg_root.place(x=0, y=0, relwidth=1, relheight=1)

# Create Extraction Label Frame
extraction_testing_frame = LabelFrame(my_testing_frame, text="Training Dataset", font="Segoe 12 bold")
extraction_testing_frame.place(width=450, height=260, x=10, y=360)

# Create Extraction Label Frame
extraction_testing_data_frame = LabelFrame(my_testing_frame, text="Extraction Features", font="Segoe 12 bold")
extraction_testing_data_frame.place(width=720, height=260, x=470, y=360)

# Create Treeview
my_tree = ttk.Treeview(my_testing_frame)
# my_tree.pack()
my_tree.place(width=420, height=200, x=20, y=400)

vsb1 = ttk.Scrollbar(my_testing_frame, orient="vertical", command=my_tree.yview)
vsb1.place(x=430, y=390, height=200)

my_tree.configure(yscrollcommand=vsb1.set)

vsb2 = ttk.Scrollbar(my_testing_frame, orient="horizontal", command=my_tree.xview)
vsb2.place(x=20, y=590, width=420)

my_tree.configure(xscrollcommand=vsb2.set)

tittle_frame = LabelFrame(my_testing_frame, text="Testing Data Processing", font="Segoe 12 bold")
tittle_frame.place(width=1180, height=340, x=10, y=10)
```

Untuk membuat kotak / box putih pada pojok kiri bawah pada Tampilan Data Testing yaitu dengan membuat **Treeview** dengan kode seperti gambar diatas.

```
txt_classification_entry = tkinter.StringVar()
txt_classification = Entry(my_testing_frame, width=30, state="readonly", textvariable=txt_classification_entry,
                           font="Segoe 30 bold", justify="center")
txt_classification.place(x=800, y=60, width=360, height=200)

original_image_testing = Label(my_testing_frame)
original_image_testing.place(x=25, y=70)

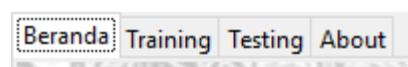
modified_image_testing = Label(my_testing_frame)
modified_image_testing.place(x=205, y=70)
```

Untuk membuat **Label**, **Entry Box** maupun **Button** yang sesuai seperti pada Tampilan Data Testing, dapat dilihat pada gambar sebelumnya.

7. Setelah **Tampilan Sistem / GUI** sudah terlihat, maka selanjutnya ada beberapa kode **Function** yang harus dituliskan agar **Proses** dalam **Aplikasi Batik Recognition** dapat dijalankan.

a) Tampilan Home/Beranda

Dalam Tampilan Home/Beranda ada fungsi yang dijalankan yaitu Tab Navigator.



- 1) Tab Navigator

Berikut merupakan source code'nya.

```
def about_tab():
    my_tabs.select(3)
def testing_tab():
    my_tabs.select(2)
def training_tab():
    my_tabs.select(1)
```

b) Tampilan Data Training

Dalam Tampilan Data Training terdapat beberapa fungsi yang dijalankan yaitu Tab Navigator, Button Select Image, Button GLCM Extraction, Button Save Data dan Radio Button Motif Batik.

1) Tab Navigator

Untuk membuat fungsi Tab Navigator dapat dilihat pada gambar sebelumnya.

2) Button Select Image

Untuk membuat fungsi pada Button Select Image dapat menggunakan kode dibawah ini.

```
# Select Image Training
def select_image():
    # Variable Path Menginisialisasi Variable Global : Path
    global path
    global name

    width = 200
    height = 200
    dim = (width, height)

    path = filedialog.askopenfilename()
    folder_path, name = os.path.split(path) # Get Name File

    # Jika Gambar Ada
    if (len(path) > 0):
        read_image = cv2.imread(path)
        # Picture RGB
        original = cv2.cvtColor(read_image, cv2.COLOR_BGR2RGB)
        # Resize Image
        resized = cv2.resize(original, dim, interpolation=cv2.INTER_AREA)
        # Set Ke Label Image
        setOriginalImage(resized)

    # Set Name File to a Label
    txt_name_file_train_text.set(name)

    else:
        messagebox.showwarning("Warning", "No File Selected. Please Select File!")
```

Dalam gambar kode fungsi *Select Image* diatas, terdapat beberapa fungsi lain seperti Membaca Citra, Konversi Citra Original ke RGB, Resize Citra dan Set Citra ke Label. Untuk menampilkan citra yang dipilih ke label *Original Image* menggunakan kode dibawah ini.

```
# Set Original Image
def setOriginalImage(images):
    # Operasi
    images = Image.fromarray(images)
    images = ImageTk.PhotoImage(images)

    # Set Image Ke Label
    original_image_training.configure(image=images)
    original_image_training.image = images
    # original_image_training.configure(width=200, height=200)
```

3) Button GLCM Extraction

Untuk membuat fungsi pada Button GLCM Extraction dapat dimulai dengan membuat kode dibawah ini dahulu.

```
def grayscale():
    width = 200
    height = 200
    dim = (width, height)

    if (len(path) > 0):
        dt_predict.clear()
        # Contrast 0
        contrast_0_ = contrast_0_()
        contrast_0_value.set(contrast_0_)

        # Contrast 45
        contrast_45_ = contrast_45_()
        contrast_45_value.set(contrast_45_)

# IDM 135
IDM_135_ = IDM_135_()
IDM_135_value.set(IDM_135_)

# convert image to grayscale image
img = cv2.imread(path)
original_image = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
gray_image = cv2.cvtColor(original_image, cv2.COLOR_BGR2GRAY)

# Resize image
resized = cv2.resize(gray_image, dim, interpolation=cv2.INTER_AREA)

# Set to Label
setModifiedImage(resized)

msgbox.showwarning("Success", "Gambar Telah Diekstraksi")

else:
    msgbox.showwarning("Warning", "No File Selected. Please Select File!")
```

Dalam gambar kode fungsi *GLCM Extraction* diatas, terdapat beberapa fungsi lain seperti Konversi Citra RGB ke *Grayscale*, *Resize* dan Set Gambar ke Label *Modified Image*. Untuk kode diatas, selain Contrast 0, Contrast 45 dan IDM 135, dituliskan juga kode untuk **ekstraksi fitur lainnya**. Untuk menampilkan citra grayscale ke label Modified Image menggunakan kode dibawah ini.

```
# set Modified Image
def setModifiedImage(images):
    # Operasi
    images = Image.fromarray(images)
    images = ImageTk.PhotoImage(images)
    # Set Image Ke Label
    modified_image_training.configure(image=images)
    modified_image_training.image = images
    modified_image_training.configure(width=200, height=200)
```

Setelah dilakukan proses mengubah citra berwarna RGB menjadi citra berwarna Grayscale, selanjutnya yaitu mengekstraksi / mengambil ciri-ciri dari citra batik. Terdapat 8 ciri fitur GLCM (Gray Level Co-occurrence Matrices) dengan masing-masing ciri bersudut 0, 45, 90 dan 135 yaitu total 32 ciri fitur GLCM (**untuk Mahasiswa bisa hanya mengambil 12 fitur saja**).

Ciri fitur yang diekstraksi yaitu Contrast 0, Contrast 45, Dissimilarity 0, Dissimilarity 45 dan seterusnya. Untuk kodennya bisa dilihat pada gambar dibawah ini.

i. Contrast 0

```
# Contrast 0
def contrast_0():
    width = 200
    height = 200
    dim = (width, height)
    if (len(path) > 0):
        img = cv2.imread(path)
        original_image = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
        gray_image = cv2.cvtColor(original_image, cv2.COLOR_BGR2GRAY)

        # Resize image
        resized = cv2.resize(gray_image, dim, interpolation=cv2.INTER_AREA)

        img = img_as_ubyte(resized)
        bins = np.array([0, 16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 176, 192, 208, 224, 240, 255]) # 16-bit
        inds = np.digitize(img, bins)

        # Looping for Value in Pixels
        max_value = inds.max() + 1
```

```
# 0=0, np.pi/4=45, np.pi/2=90, 3*np.pi/4=135
matrix_cocurrence = greycomatrix(inds, [1], [0], levels=max_value,
                                 normed=False, symmetric=False)

# Operasi Contrast
contrast_0 = greycoprops(matrix_cocurrence, 'contrast')

# Set to Label
# setModifiedImage(contrast_0)

# Insert Value to Data Prediction
dt_predict.insert(0, contrast_0[0][0])

return round(contrast_0[0][0], 7)
```

```
dt_predict = []
```

Untuk fitur yang sama dengan sudut yang berbeda, Anda dapat mengubah kode nilainya dengan melihat gambar dibawah ini.

Dalam kode dibawah ini, **untuk sudut 0 menggunakan nilai 0, sudut 45 menggunakan nilai np.pi/4, sudut 90 menggunakan nilai np.pi/2 dan sudut 135 menggunakan 3*np.pi/4**

```
# 0=0, np.pi/4=45, np.pi/2=90, 3*np.pi/4=135
matrix_coocurrence = greycomatrix(inds, [1], [0], levels=max_value,
                                 normed=False, symmetric=False)
```

ii. Contrast 45, 90, 135

Dalam fitur Contrast 45,90,135 , penulisan kode hampir sama dengan Fitur Contrast 0, hanya saja dilakukan perubahan pada beberapa hal seperti pada gambar dibawah.

```
# 0=0, np.pi/4=45, np.pi/2=90, 3*np.pi/4=135
matrix_coocurrence = greycomatrix(inds, [1], [np.pi / 4], levels=max_value,
                                 normed=False, symmetric=False)

# Operasi Contrast
contrast_45 = greycoprops(matrix_coocurrence, 'contrast')

# Insert Value to Data Prediction
dt_predict.insert(8, contrast_45[0][0])

return round(contrast_45[0][0], 7)
```

iii. IDM 0

Dalam fitur IDM 0, dilakukan perubahan pada beberapa hal seperti pada gambar dibawah.

```
# 0=0, np.pi/4=45, np.pi/2=90, 3*np.pi/4=135
matrix_coocurrence = greycomatrix(inds, [1], [0], levels=max_value,
                                 normed=True, symmetric=True)

# Operasi IDM
(num_level, num_level2, num_dist, num_angle) = matrix_coocurrence.shape
i, j = np.ogrid[0:num_level, 0:num_level]
IDM_0 = 1 / (1 + (i - j) * (i - j))
weights = IDM_0.reshape(num_level, num_level, 1, 1)
results = np.apply_over_axes(np.sum, (matrix_coocurrence * weights), axes=(0, 1))[0, 0]

# Insert Value to Data Prediction
dt_predict.insert(6, results[0][0])

return round(results[0][0], 7)
```

iv. IDM 45, 90, 135

Dalam fitur IDM 45, 90, 135 dilakukan perubahan pada beberapa hal seperti pada gambar dibawah.

```
# 0=0, np.pi/4=45, np.pi/2=90, 3*np.pi/4=135
matrix_cocurrence = greycomatrix(nds, [1], [np.pi / 4], levels=max_value,
                                 normed=True, symmetric=True)

# Operasi IDM
(num_level, num_level2, num_dist, num_angle) = matrix_cocurrence.shape
i, j = np.ogrid[0:num_level, 0:num_level]
IDM_45 = 1 / (1 + (i - j) * (i - j))
weights = IDM_45.reshape((num_level, num_level, 1, 1))
results = np.apply_over_axes(np.sum, (matrix_cocurrence * weights), axes=(0, 1))[0, 0]

# Insert Value to Data Prediction
dt_predict.insert(14, results[0][0])

return round(results[0][0], 7)
```

v. Entropy 0

Dalam fitur Entropy 0, dilakukan perubahan pada beberapa hal seperti pada gambar dibawah.

```
# 0=0, np.pi/4=45, np.pi/2=90, 3*np.pi/4=135
matrix_cocurrence = np.squeeze(greycomatrix(nds, [1], [0], levels=max_value,
                                             normed=True, symmetric=True))

# Operasi Entropy
entropy_0 = -np.sum(matrix_cocurrence * np.log2(matrix_cocurrence + (matrix_cocurrence == 0)))

# Insert Value to Data Prediction
dt_predict.insert(7, entropy_0)

return round(entropy_0, 7)
```

vi. Entropy 45, 90, 135

Dalam fitur Entropy 45, 90, 135 dilakukan perubahan pada beberapa hal seperti pada gambar dibawah.

```
# 0=0, np.pi/4=45, np.pi/2=90, 3*np.pi/4=135
matrix_cocurrence = np.squeeze(greycomatrix(nds, [1], [np.pi / 4], levels=max_value,
                                             normed=True, symmetric=True))

# Operasi Entropy
entropy_45 = -np.sum(matrix_cocurrence * np.log2(matrix_cocurrence + (matrix_cocurrence == 0)))

# Insert Value to Data Prediction
dt_predict.insert(15, entropy_45)

return round(entropy_45, 7)
```

vii. Dissimilarity 0

Dalam fitur Dissimilarity 0, dilakukan perubahan pada beberapa hal seperti pada gambar dibawah.

```
# 0=0, np.pi/4=45, np.pi/2=90, 3*np.pi/4=135
matrix_cocurrence = greycomatrix(nds, [1], [0], levels=max_value,
                                 normed=False, symmetric=False)

# Operasi Dissimilarity
dissimilarity_0 = greycoprops(matrix_cocurrence, 'dissimilarity')

# Set to Label
setModifiedImage(dissimilarity_0)

# Insert Value to Data Prediction
dt_predict.insert(1, dissimilarity_0[0][0])

return round(dissimilarity_0[0][0], 7)
```

viii. Dissimilarity 45, 90, 135

Dalam fitur Dissimilarity 45, 90, 135 dilakukan perubahan pada beberapa hal seperti pada gambar dibawah.

```
# 0=0, np.pi/4=45, np.pi/2=90, 3*np.pi/4=135
matrix_coocurrence = greycomatrix(inds, [1], [np.pi / 4], levels=max_value,
                                 normed=False, symmetric=False)

# Operasi Dissimilarity
dissimilarity_45 = greycoprops(matrix_coocurrence, 'dissimilarity')

# Set to Label
setModifiedImage(dissimilarity_45)

# Insert Value to Data Prediction
dt_predict.insert(9, dissimilarity_45[0][0])

return round(dissimilarity_45[0][0], 7)
```

ix. Homogeneity 0

Dalam fitur Homogeneity 0, dilakukan perubahan pada beberapa hal seperti pada gambar dibawah.

```
# 0=0, np.pi/4=45, np.pi/2=90, 3*np.pi/4=135
matrix_coocurrence = greycomatrix(inds, [1], [0], levels=max_value,
                                 normed=False, symmetric=False)

# Operasi Homogeneity
homogeneity_0 = greycoprops(matrix_coocurrence, 'homogeneity')

# Set to Label
setModifiedImage(homogeneity_0)

# Insert Value to Data Prediction
dt_predict.insert(2, homogeneity_0[0][0])

return round(homogeneity_0[0][0], 7)
```

x. Homogeneity 45, 90, 135

Dalam fitur Homogeneity 45, 90, 135 dilakukan perubahan pada beberapa hal seperti pada gambar dibawah.

```
# 0=0, np.pi/4=45, np.pi/2=90, 3*np.pi/4=135
matrix_coocurrence = greycomatrix(inds, [1], [np.pi / 4], levels=max_value,
                                 normed=False, symmetric=False)

# Operasi Homogeneity
homogeneity_45 = greycoprops(matrix_coocurrence, 'homogeneity')

# Set to Label
setModifiedImage(homogeneity_45)

# Insert Value to Data Prediction
dt_predict.insert(10, homogeneity_45[0][0])

return round(homogeneity_45[0][0], 7)
```

xi. Energy 0

Dalam fitur Energy 0, dilakukan perubahan pada beberapa hal seperti pada gambar dibawah.

```
# 0=0, np.pi/4=45, np.pi/2=90, 3*np.pi/4=135
matrix_coocurrence = greycomatrix(inds, [1], [0], levels=max_value,
                                 normed=False, symmetric=False)

# Operasi Energy
energy_0 = greycoprops(matrix_coocurrence, 'energy')

# Set to Label
setModifiedImage(energy_0)

# Insert Value to Data Prediction
dt_predict.insert(3, energy_0[0][0])

return round(energy_0[0][0], 7)
```

xii. Energy 45, 90, 135

Dalam fitur Energy 45, 90, 135 dilakukan perubahan pada beberapa hal seperti pada gambar dibawah.

```
# 0=0, np.pi/4=45, np.pi/2=90, 3*np.pi/4=135
matrix_coocurrence = greycomatrix(inds, [1], [np.pi / 4], levels=max_value,
                                 normed=False, symmetric=False)

# Operasi Energy
energy_45 = greycoprops(matrix_coocurrence, 'energy')

# Set to Label
setModifiedImage(energy_45)

# Insert Value to Data Prediction
dt_predict.insert(11, energy_45[0][0])

return round(energy_45[0][0], 7)
```

xiii. ASM 0

Dalam fitur ASM 0, dilakukan perubahan pada beberapa hal seperti pada gambar dibawah.

```
# 0=0, np.pi/4=45, np.pi/2=90, 3*np.pi/4=135
matrix_coocurrence = greycomatrix(inds, [1], [0], levels=max_value,
                                 normed=False, symmetric=False)

# Operasi ASM
ASM_0 = greycoprops(matrix_coocurrence, 'ASM')

# Set to Label
setModifiedImage(ASM_0)

# Insert Value to Data Prediction
dt_predict.insert(4, ASM_0[0][0])

return round(ASM_0[0][0], 7)
```

xiv. ASM 45, 90, 135

Dalam fitur ASM 45, 90, 135 dilakukan perubahan pada beberapa hal seperti pada gambar dibawah.

```
# 0=0, np.pi/4=45, np.pi/2=90, 3*np.pi/4=135
matrix_coocurrence = greycomatrix(inds, [1], [np.pi / 4], levels=max_value,
                                    normed=False, symmetric=False)

# Operasi ASM
ASM_45 = greycoprops(matrix_coocurrence, 'ASM')

# Set to Label
setModifiedImage(ASM_45)

# Insert Value to Data Prediction
dt_predict.insert(12, ASM_45[0][0])

return round(ASM_45[0][0], 7)
```

xv. Correlation 0

Dalam fitur Correlation 0, dilakukan perubahan pada beberapa hal seperti pada gambar dibawah.

```
# 0=0, np.pi/4=45, np.pi/2=90, 3*np.pi/4=135
matrix_coocurrence = greycomatrix(inds, [1], [0], levels=max_value,
                                    normed=False, symmetric=False)

# Operasi Correlation
correlation_0 = greycoprops(matrix_coocurrence, 'correlation')

# Set to Label
setModifiedImage(correlation_0)

# Insert Value to Data Prediction
dt_predict.insert(5, correlation_0[0][0])

return round(correlation_0[0][0], 7)
```

xvi. Correlation 45, 90, 135

Dalam fitur Correlation 45, 90, 135 dilakukan perubahan pada beberapa hal seperti pada gambar dibawah.

```
# 0=0, np.pi/4=45, np.pi/2=90, 3*np.pi/4=135
matrix_coocurrence = greycomatrix(inds, [1], [np.pi / 4], levels=max_value,
                                    normed=False, symmetric=False)

# Operasi Correlation
correlation_45 = greycoprops(matrix_coocurrence, 'correlation')

# Set to Label
setModifiedImage(correlation_45)

# Insert Value to Data Prediction
dt_predict.insert(13, correlation_45[0][0])

return round(correlation_45[0][0], 7)
```

4) Button Save Data

Untuk membuat fungsi pada Button Save Data dapat menggunakan kode dibawah ini.

5) Radio Button Motif Batik

Untuk membuat fungsi pada Radio Button dapat menggunakan kode dibawah ini.

```
def radio_button():
    # Set Name File to a Label
    txt_radio_text.set(r.get())
```

c) Tampilan Data Testing

Dalam Tampilan Data Testing terdapat beberapa fungsi yang dijalankan yaitu Tab Navigator, Open Data Training, Button Select Image, Button GLCM Extraction dan Button Process.

1) Tab Navigator

Untuk membuat fungsi Tab Navigator dapat dilihat pada gambar sebelumnya.

2) Button Select Image

Untuk membuat fungsi Select Image dapat dilihat pada gambar sebelumnya. Anda dapat melihat kodennya seperti dibawah ini.

```
def select_image_test():
    # Variable Path Menginisialisasi Variable Global : Path
    global path
    global name

    width = 200
    height = 200
    dim = (width, height)

    path = filedialog.askopenfilename()
    folder_path, name = os.path.split(path) # Get Name File
    # Jika Gambar Ada
    if (len(path)) > 0:

        read_image = cv2.imread(path)
        # Picture RGB
        original = cv2.cvtColor(read_image, cv2.COLOR_BGR2RGB)
        # Resize Image
        resized = cv2.resize(original, dim, interpolation=cv2.INTER_AREA)
        # Set Ke Label Image
        setOriginalImageTest(resized)

    else:
        msgbox.showwarning("Warning", "No File Selected. Please Select File!")
```

Untuk menampilkan citra yang dipilih ke label Original Image menggunakan kode dibawah ini.

```
# Set Original Image
def setOriginalImageTest(images):
    # Operasi
    images = Image.fromarray(images)
    images = ImageTk.PhotoImage(images)

    # Set Image Ke Label
    original_image_testing.configure(image=images)
    original_image_testing.image = images
    original_image_testing.configure(width=200, height=200)
```

3) Open Data Training

Training Dataset					
No.	Nama	Contras	Dissimilarity 0	Homogeneity 0	
1	1_Jember (1).jpg	4.22771	1.16681	0.651832	
2	1_Jember (2).jpg	6.27829	1.51618	0.586911	
3	1_Jember (3).jpg	5.59525	1.5252	0.547467	
4	1_Jember (4).jpg	4.3553	1.2744	0.601624	
5	1_Jember (5).jpg	3.86256	1.15668	0.632506	
6	1_Jember (6).jpg	5.75337	1.6004	0.52244	
7	1_Jember (7).jpg	4.46957	1.27661	0.606205	
8	1_Jember (8).jpg	3.67477	1.02216	0.696203	

Untuk membuka dataset nilai dari data training seperti pada gambar diatas, maka dituliskan kode seperti pada gambar berikut.

```

def open_data_training():
    ...
    SQL_Query = pd.read_sql_query("""
        '''select
        *
        from tbl_data_training_t'''', mydb)
    df = pd.DataFrame(SQL_Query,
                      columns=['nama_batik', 'contrast_0', 'dissimilarity_0', 'homogeneity_0', 'energy_0', 'asm_0',
                                'correlation_0', 'idm_0', 'entropy_0', 'contrast_45', 'dissimilarity_45',
                                'homogeneity_45', 'energy_45', 'asm_45', 'correlation_45', 'idm_45', 'entropy_45',
                                'contrast_90', 'dissimilarity_90', 'homogeneity_90', 'energy_90', 'asm_90',
                                'correlation_90', 'idm_90', 'entropy_90', 'contrast_135', 'dissimilarity_135',
                                'homogeneity_135', 'energy_135', 'asm_135', 'correlation_135', 'idm_135',
                                'entropy_135', 'Target'])
    clear_tree()
    # Set Up A New Treeview
    my_tree["column"] = list(df.columns)
    my_tree["show"] = "headings"

    # Loop Thru Columns List For Headers
    for column in my_tree["column"]:
        my_tree.heading(column, text=column)

    # Put Data in Treeview
    df_rows = df.to_numpy().tolist()
    for row in df_rows:
        my_tree.insert("", "end", values=row)

    my_tree.place(width=420, height=200, x=20, y=400)

# Delete the Old Tree
def clear_tree():
    my_tree.delete(*my_tree.get_children())

```

def clear_tree berfungsi untuk mereset tampilan dalam **Treeview** ketika dataset dibuka kembali

4) Button GLCM Extraction

Dalam GLCM Extraction, terdapat beberapa fungsi lain seperti Konversi Citra RGB ke Grayscale, Resize dan Set Gambar ke Label Modified Image

```

def grayscale_test():
    width = 200
    height = 200
    dim = (width, height)

    if (len(path) > 0):

        # Read Image As Grayscale
        # grayscale = cv2.imread(path, 0)

        # convert image to grayscale image
        img = cv2.imread(path)
        original_image = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
        gray_image = cv2.cvtColor(original_image, cv2.COLOR_BGR2GRAY)

        # Resize image
        resized = cv2.resize(gray_image, dim, interpolation=cv2.INTER_AREA)

        # Set to Label
        setModifiedImageTest(resized)

        dt_predict.clear()

```

```
# IDM 90
IDM_90_test = IDM_90()
IDM_90_value_test.set(IDM_90_test)

# IDM 135
IDM_135_test = IDM_135()
IDM_135_value_test.set(IDM_135_test)

msgbox.showwarning("Success", "Gambar Telah Diekstraksi")
else:
    msgbox.showwarning("Warning", "No File Selected. Please Select File!")
```

Untuk kode diatas, selain IDM 90 dan IDM 135, dituliskan juga kode untuk ekstraksi fitur Contrast 0, Contrast 45, Dissimilarity 90, Dissimilarity 135, dst. Untuk menampilkan citra grayscale ke label Modified Image menggunakan kode dibawah ini.

```
# set Modified Image
def setModifiedImageTest(images):
    # Operasi
    images = Image.fromarray(images)
    images = ImageTk.PhotoImage(images)
    # Set Image Ke Label
    modified_image_testing.configure(image=images)
    modified_image_testing.image = images
    modified_image_testing.configure(width=200, height=200)
```

5) Button Process

Dalam button process ini terdapat fungsi utama yaitu Mengklasifikasikan Citra ke dalam kelas-kelas tertentu yaitu Batik Banyuwangi, Batik Jember, Batik Bondowoso dan Batik Situbondo.

```
def classification(dp):
    df_p = dp
    # Remove Replace Warning
    pd.options.mode.chained_assignment = None
    # read from db
    # mycursor = mydb.cursor()
    # my_cursor.execute("SELECT * FROM tbl_data_training_t")
    #
    try:
        SQL_Query = pd.read_sql_query(
            '''select
            *
            from tbl_data_training_t''', mydb)

        df = pd.DataFrame(SQL_Query, columns=['nama_batik', 'contrast_0', 'dissimilarity_0', 'homogeneity_0',
                                               'energy_0', 'asm_0', 'correlation_0', 'idm_0', 'entropy_0',
                                               'contrast_45', 'dissimilarity_45', 'homogeneity_45', 'energy_45',
                                               'asm_45', 'correlation_45', 'idm_45', 'entropy_45', 'contrast_90',
                                               'dissimilarity_90', 'homogeneity_90', 'energy_90', 'asm_90',
                                               'correlation_90', 'idm_90', 'entropy_90', 'contrast_135',
                                               'dissimilarity_135', 'homogeneity_135', 'energy_135', 'asm_135',
                                               'correlation_135', 'idm_135', 'entropy_135', 'Target'])
```

```

except:
    print("Error: unable to convert the data")
...
df.drop(["nama_batik"], axis=1, inplace=True)
# print(df.head())

# Convert Non-Numeric Data to Numeric Data
df.Target[df.Target == "Jember"] = 1
df.Target[df.Target == "Banyuwangi"] = 2
df.Target[df.Target == "Bondowoso"] = 3
df.Target[df.Target == "Situbondo"] = 4
# print(df.head())

# Define Dependent Variable
Y = df['Target'].values
Y = Y.astype('int')

# Define Independent Variable
X = df.drop(labels=['Target'], axis=1)

# Random Forest Classifier
model = RandomForestClassifier(n_estimators=200, criterion='entropy', random_state=20, max_depth=None)
target = ['', 'Jember', 'Banyuwangi', 'Bondowoso', 'Situbondo']
model.fit(X, Y)

# Prediction
prediction_test = model.predict([df_p]).reshape(-1, 1)

dt_predict.clear()
hasil_split = name.split('.')
L = list(hasil_split[0])
print(name.split('.'))
print(L[0])
print('Hasil Klasifikasi : ', target[prediction_test[0][0]])
# print('Accuracy : ', metrics.accuracy_score([int(L[0])], prediction_test))

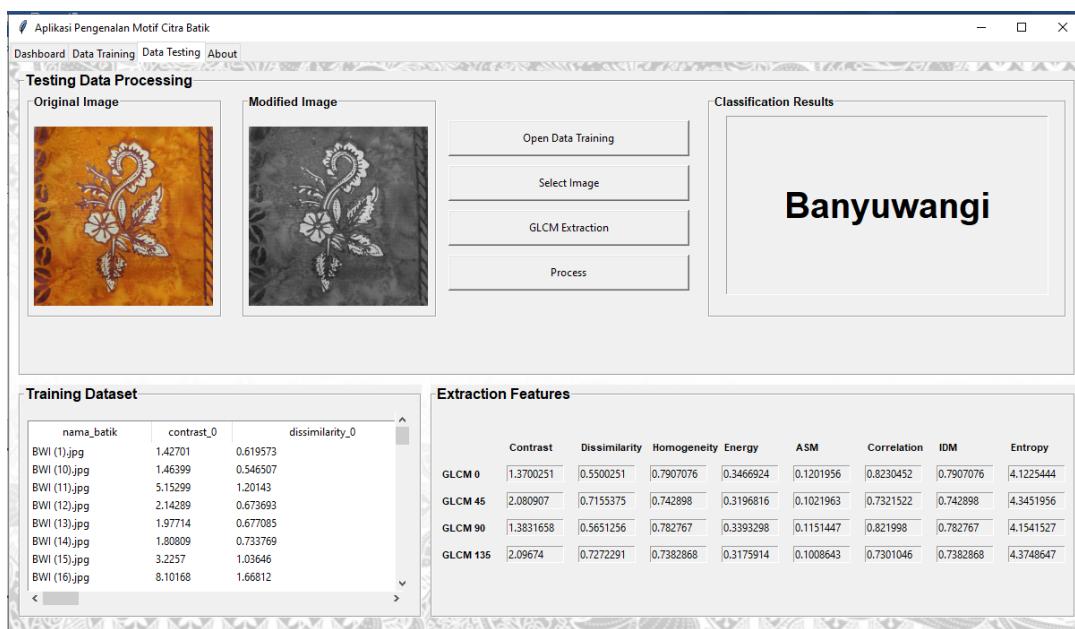
txt_classification_entry.set(target[prediction_test[0][0]])

msgbox.showwarning("Success", "Gambar Telah Terkласifikasi")

```

f. Hasil dan Pembahasan

Berikut merupakan contoh tampilan bila proses klasifikasi telah berhasil dilakukan.



g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 49-52

Pokok Bahasan : Studi Kasus (Deteksi Keaslian Uang Kertas)

Acara Praktikum/Praktik : Minggu 14/1-4

Tempat : Politeknik Negeri Jember

Alokasi Waktu : 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami definisi dari pengolahan citra dan vision.
2. Mahasiswa mampu memberikan contoh penerapan pengolahan citra digital pada suatu studi kasus (Deteksi Keaslian Uang Kertas)

b. Indikator

Ketepatan dalam menjelaskan mengenai penerapan pengolahan citra digital pada suatu studi kasus (Deteksi Keaslian Uang Kertas)

c. Dasar Teori

Kejadian pemalsuan uang saat ini semakin meningkat di Indonesia[1]. Pengertian uang palsu atau rupiah palsu berdasarkan Undang-Undang Nomor 7 Tahun 2011 tentang mata uang, rupiah palsu adalah suatu benda yang bahan, ukuran, warna, gambar, dan/atau desainnya menyerupai rupiah yang dibuat, dibentuk, dicetak, digandakan, diedarkan, atau digunakan sebagai alat pembayaran secara melawan hukum[2]. Uang kertas rupiah memiliki ciri-ciri keaslian yang dapat dilihat dengan dengan 3D (dilihat, diraba, diterawang)[3]. Namun hal tersebut masih kurang efektif, terbukti dengan masih banyaknya terjadi kasus penipuan uang kertas palsu di kalangan masyarakat. Penentuan keaslian uang kertas menggunakan panca indra dinilai kurang efektif dikarenakan penilaian setiap orang bersifat subjektif. Selain ciri-ciri yang terlihat secara langsung, otentifikasi keaslian uang kertas juga dapat dilakukan dengan melihat *invisible ink* pada uang asli yang hanya tampak ketika uang kertas disinari sinar *ultraviolet*.

Namun cara tersebut juga berdampak buruk bagi kesehatan, karena pemaparan sinar *ultraviolet* (UV) yang berkepanjangan dapat menyebabkan kanker kulit, kerusakan mata dan menurunkan kekebalan tubuh[4]. Untuk mengurangi dampak dari proses pengecekan *invisible ink* menggunakan pengamatan langsung maka dibutuhkan sebuah sistem yang dapat mempelajari pola keaslian uang kertas berdasarkan *invisible ink* sehingga deteksi keaslian menjadi lebih baik dan akurat.

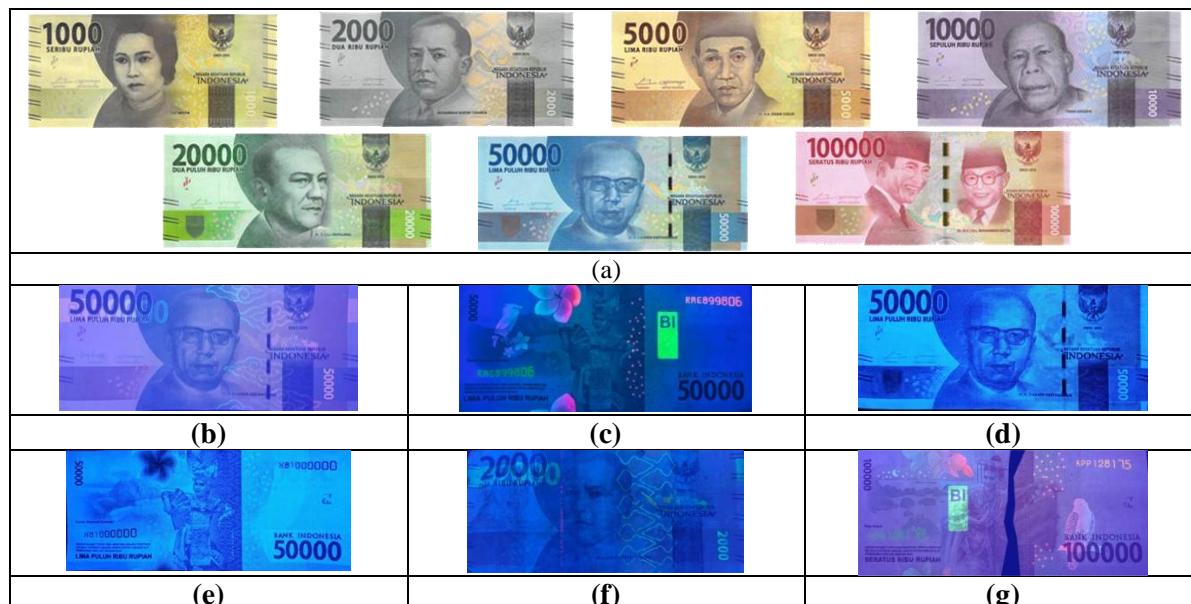
Sistem tersebut menerapkan *image processing* (pengolahan citra) agar sistem dapat meniru mata manusia dalam menganalisa keaslian citra uang yang akan dideteksi.

d. Alat dan Bahan

1. Laptop
2. Kertas A4
3. Pulpen
4. Matlab

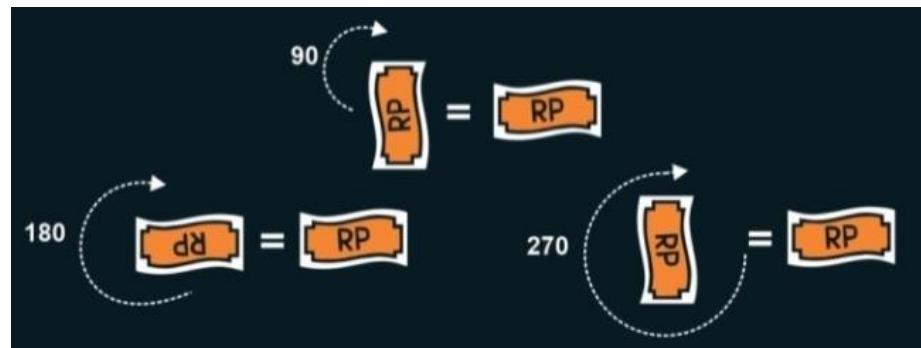
e. Prosedur Kerja

1. Kumpulkan citra uang kertas, proses pengambilan data menggunakan kamera smartphone resolusi 13 MP berlatarbelakang gelap di ruangan tertutup. Untuk pencahayaan, menggunakan lampu ultraviolet (UV) Neon T5 8 Watt. Jarak pengambilan foto antara lensa kamera dengan uang kertas yaitu ± 13 cm dan uang yang kami gunakan terdiri dari uang asli sisi depan dan belakang, uang palsu sisi depan dan belakang, uang lusuh, serta uang terpotong hasil editing (Gambar 65).



Gambar 65. (a) uang kertas rupiah, (b) Uang Asli Tampak Depan (c) Uang Asli Tampak Belakang (d) Uang Palsu Tampak Depan (e) Uang Palsu Tampak Belakang (f) Uang Lusuh (g) Uang Terpotong

2. Lakukan rotasi seperti gambar 66.

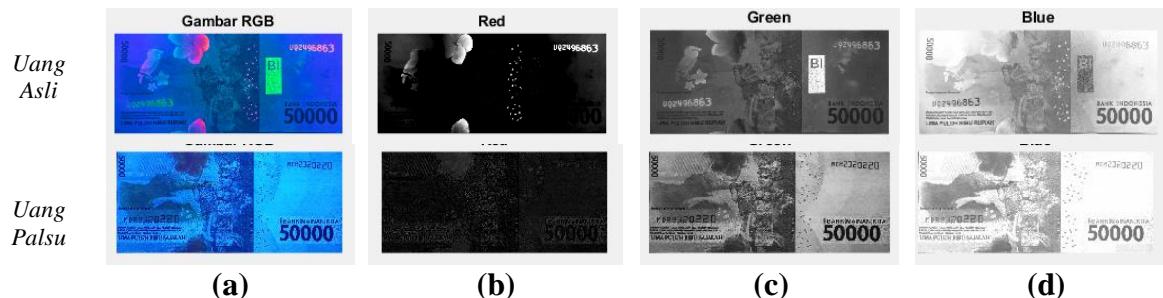


Gambar 66. Gambaran Rotasi Image

3. Kemudian lakukan cropping citra dan resize citra untuk proses normalisasi data.



4. Lakukan proses pemecahan komponen RGB



Gambar 67. (a) Citra RGB (b) Citra Red (c) Citra Green dan (d) Citra Blue

3. Lakukan ekstraksi fitur tekstur menggunakan GLCM pada materi sebelumnya menggunakan aplikasi pemrograman matlab atau python. Kemudian hasil ekstraksi fitur tersebut simpan kedalam file excel !

4. kemudian implementasi menggunakan sistem cerdas yang anda pahami !

f. Hasil dan Pembahasan

1. Dokumentasi Tugas berupa laporan (Resume Jurnal)
2. Dokumentasi Tugas berupa file ppt

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 53-54

Pokok Bahasan	: Implementasi <i>Computer Vision</i> dengan <i>Python code</i>
Acara Praktikum/Praktik : Minggu 15/1-2	
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	: 2 x 100 Menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

- 1) Mahasiswa mampu menjelaskan tahapan-tahapan meyiapkan (*install*) perangkat lunak untuk menulis kode program *Python*.
- 2) Mahasiswa mampu meng-*install* beberapa perangkat lunak yang dibutuhkan untuk mengembangkan aplikasi *computer vision* dengan *Python* .

b. Indikator

Kemampuan mahasiswa dalam menjelaskan dan mengimplementasikan *Python code* sebagai alternatif mengembangkan aplikasi *computer vision*.

c. Dasar Teori

Python adalah sebuah bahasa pemrograman tingkat tinggi berorientasi objek yang diciptakan oleh *Guido van Rossum*. *Python* memiliki sintaks yang sederhana dan mudah, sehingga membuatnya menjadi bahasa yang sangat cocok untuk pemula yang ingin belajar pemrograman komputer. *Python* adalah bahasa pemrograman interpretatif yang dapat digunakan di **berbagai platform** dengan filosofi perancangan yang berfokus pada tingkat keterbacaan kode dan merupakan salah satu bahasa populer yang berkaitan dengan *Data Science*, *Machine Learning*, *Computer vision*, dan *Internet of Things* (IoT). Keunggulan *Python* juga banyak digunakan untuk *prototyping*, *scripting* dalam pengelolaan infrastruktur, hingga pembuatan *website* berskala besar.

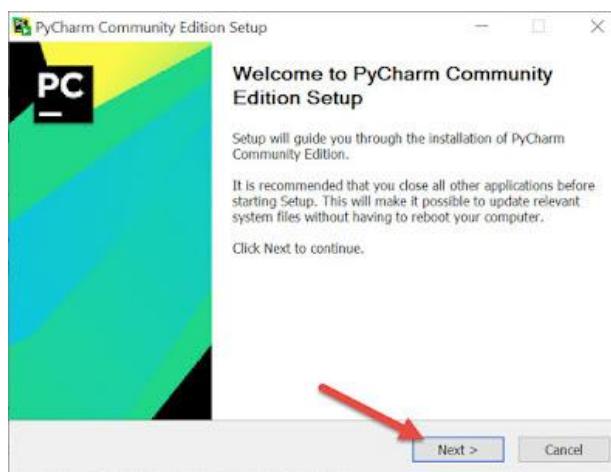
- Bahasa *Python* menjadi keharusan untuk Anda yang ingin mempelajari dasar-dasar scripting dan pengolahan data atau machine learning.
- Bahasa *Python* digunakan secara luas, masuk dalam 3 besar bahasa pemrograman yang digunakan dalam beberapa tahun belakangan.
- Pustaka (*Library*) yang luas, memungkinkan Anda mengembangkan ke bidang-bidang lainnya. Beberapa *library* atau *framework* terpopuler data science dan machine learning menggunakan *Python* antara lain: *Scikit-Learn*, *TensorFlow*, *PyTorch*.
- Bahasa *Python* memiliki kurva pembelajaran (*learning-curve*) yang sangat landai, cocok untuk dipelajari sebagai bahasa pemrograman pertama - dengan kemudahan pembacaan dan kemudahan mempelajari sintaksisnya..

d. Alat dan Bahan

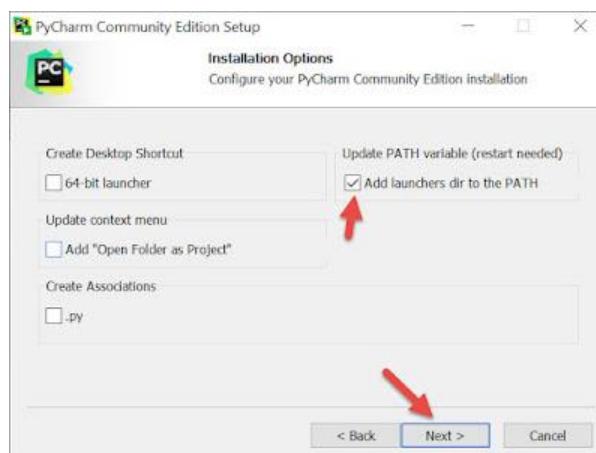
1. Laptop
2. Kertas A4 & Pulpen
3. PyCharm Community Edition version 2021.2.2 (atau yang tersedia)
4. Python Version 3.7.9 – 64 bit (harus versi 3.7.x atau diatasnya)

e. Prosedur Kerja

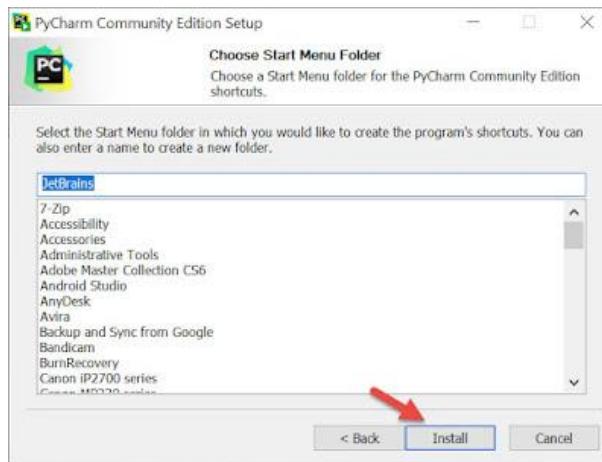
1. Sebelum meng-*install* beberapa perangkat lunak yang dibutuhkan dalam proses kerja nantinya, sebaiknya diperhatikan *minimum System Requirements*:
 - 64-bit versions of Microsoft Windows 10, 8
 - 2 GB RAM minimum, 8 GB RAM recommended
 - 2.5 GB hard disk space, SSD recommended
 - 1024x768 minimum screen resolution
2. Tahapan instalasi *software pycharm*, download Installer Pycharm dari situs resminya : <https://www.jetbrains.com/pycharm/download>
3. Jalankan **Installer PyCharm**, kemudian klik **Next** untuk melanjutkan installasi seperti pada contoh gambar di bawah ini.



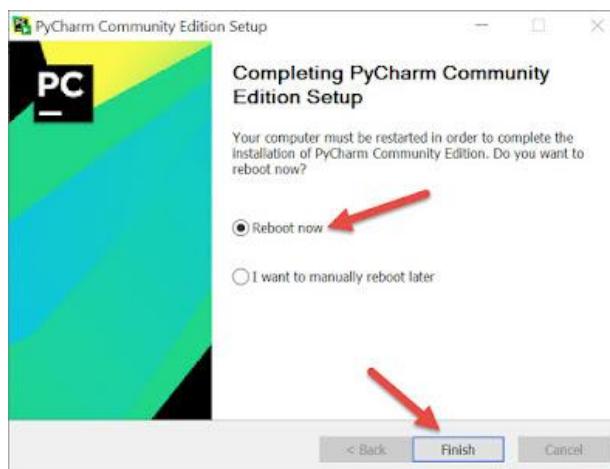
4. Tentukan **destination folder** jika diperlukan, atau dibiarkan default kemudian klik **Next**.
5. Checklist pada "**Add launchers dir to PATH**" kemudian klik **Next**.



6. Klik **Install** untuk melanjutkan,



7. Tunggu proses installasi **PyCharm** sampai selesai.
8. Setelah proses installasi PyCharm selesai, klik **Reboot now**, lalu klik **Finish**.



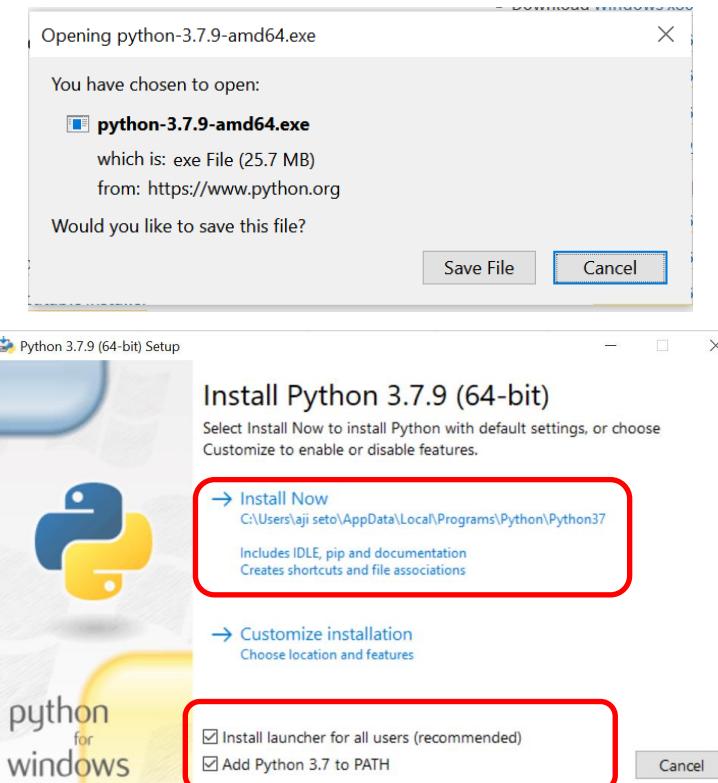
9. Tahapan **instalasi python**, Unduh instalasi Python yang berbentuk executable di <https://www.python.org/downloads/windows/>. Pilih **versi 3.7** atau diatasnya. Python ini digunakan sebagai base interpreter virtual environment/ irtualenv/ venv.

A screenshot of a web browser displaying the Python website at https://www.python.org/downloads/windows/. The page title is 'Python Releases for Windows'. It shows two sections: 'Stable Releases' and 'Pre-releases'. Under 'Stable Releases', there are links for 'Latest Python 3 Release - Python 3.10.0' and 'Latest Python 2 Release - Python 2.7.18'. Under 'Pre-releases', there is a link for 'Python 3.11.0a1 - Oct. 5, 2021'. Each release section includes download links for 'Windows embeddable package (32-bit)' and 'Windows embeddable package (64-bit)'. The browser address bar shows the URL: https://www.python.org/downloads/windows/.

BKPM - (NAMA BPKM)

- [Python 3.5.10 - Sept. 5, 2020](#)
Note that Python 3.5.10 cannot be used on Windows XP or earlier.
 - No files for this release.
- [Python 3.7.9 - Aug. 17, 2020](#)
Note that Python 3.7.9 cannot be used on Windows XP or earlier.
 - Download [Windows help file](#)
 - Download [Windows x86-64 embeddable zip file](#)
 - Download [Windows x86 executable installer](#) 
 - Download [Windows x86-64 web-based installer](#)
 - Download [Windows x86 embeddable zip file](#)
 - Download [Windows x86 executable installer](#)
 - Download [Windows x86 web-based installer](#)
- [Python 3.7.8rc1 - June 17, 2020](#)
 - Download [Windows help file](#)
 - Download [Windows x86-64 embeddable zip file](#)
 - Download [Windows x86-64 executable installer](#)
 - Download [Windows x86-64 web-based installer](#)
 - Download [Windows x86 embeddable zip file](#)
 - Download [Windows x86 executable installer](#)
 - Download [Windows x86 web-based installer](#)
- [Python 3.6.11rc1 - June 17, 2020](#)

10. Buka file installer yang telah di download



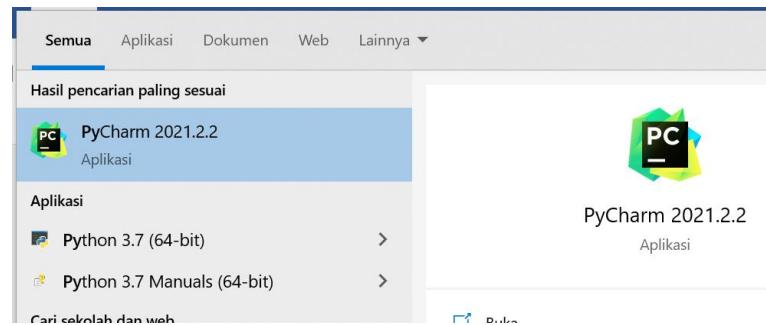
11. Centang **Install launcher for all user** untuk mengaktifkan python pada semua user Windows dan centang **Python 3.7 to PATH** untuk menambah path command Python. Kemudian klik **Install Now**. Klik **Yes** saat muncul notifikasi User Account Control. **Tunggu proses install hingga selesai**.

12. Opsi "**Disable path length limit**"

Disini, dapat mengaktifkan atau menonaktifkan batasan lintasan direktori Python. Untuk pengguna Linux batasan ini tidak mempunyai pengaruh yang besar, karena Linux menyimpan python pada direktori yang pendek. Berbeda halnya dengan Windows, yang menyimpan direktori lintasan agak jauh dari direktori utama partisi. Disarankan untuk menonaktifkan batasan ini dengan klik "**Disable path length limit**". Klik **Yes** saat muncul notifikasi User Account Control.

f. Hasil dan Pembahasan

Setelah melakukan proses install Pycharm dan Python 3.7 maka akan terlihat kedua perangkat lunak itu di komputer Anda

**g. Rubrik Penilaian**

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 55-56

Pokok Bahasan	: Hand Tracking menggunakan Python dan Mediapipe
Acara Praktikum/Praktik : Minggu 15/3-4	
Tempat	: Laboratorium Multimedia Cerdas
Alokasi Waktu	: 2 x 100 Menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

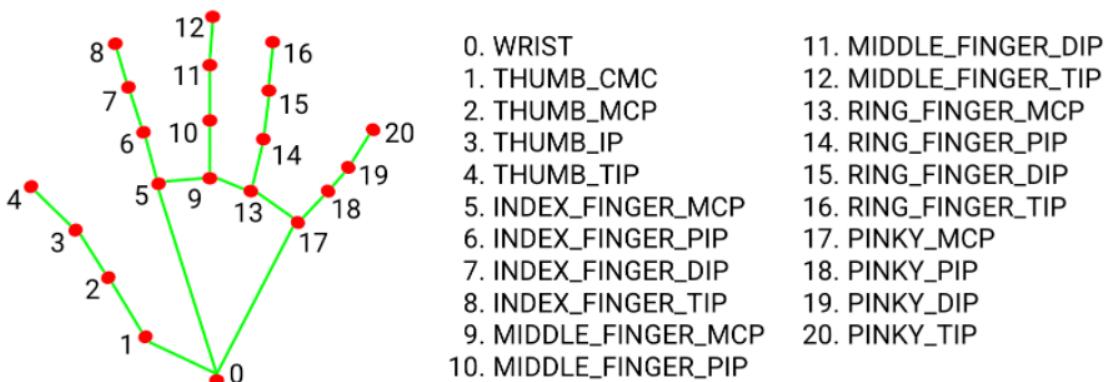
- 1) Mahasiswa mampu menjelaskan konsep *Hand Tracking*.
- 2) Mahasiswa mampu mengembangkan *Hand tracking System* dengan *Python* dan *Mediapipe*.

b. Indikator

Kemampuan mahasiswa dalam menjelaskan dan mengimplementasikan *Hand tracking System* dengan *Python* dan *Mediapipe*.

c. Dasar Teori

Teknik *hand tracking* merupakan teknik *motion tracking* yang lebih berfokus pada organ tangan. Teknik ini banyak digunakan dalam industri film (animasi), aplikasi permainan (*game*), robotika ataupun untuk pengembangan HCI (Human Computer Interaction). Model *Landmark Tangan*



Sumber: <https://google.github.io/mediapipe/solutions/hands.html>

MediaPipe

MediaPipe adalah *open-source framework* yang dirilis Google dan digunakan untuk membangun audio, video, atau berbagai *time series data*. Beberapa aplikasi utama MediaPipe: (1) Pelacakan Multi-tangan, (2) Deteksi wajah, (3) Deteksi dan Pelacakan Objek, (4) Objectron: Deteksi dan Pelacakan Objek 3D, dan (5) AutoFlip: Pipa pemotongan video otomatis dll. Pada dasarnya, *MediaPipe* menggunakan *single-shot*

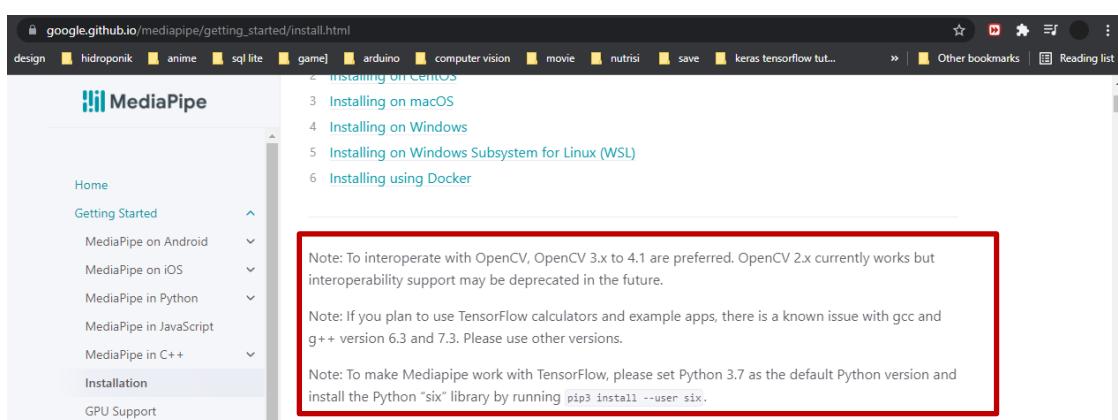
palm detection model serta menerapkan teknik pembelajaran mesin (*supervised*) untuk mengidentifikasi 21 titik kunci 3D dari satu bingkai gambar tangan.

d. Alat dan Bahan

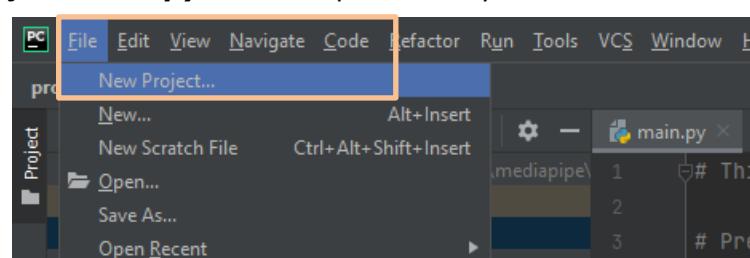
1. Laptop
2. Kertas A4 & Pulpen
3. Perangkat Lunak **PyCharm**
4. Perangkat Lunak **Python**
5. Library Numpy, OpenCV, Mediapipe

e. Prosedur Kerja

1. Pastikan dulu Bawa Anda telah meg-install Pycharm dan Python 3.7 (sesuai praktikum 45-46 sebelumnya).
2. Sebelum install **mediapipe** disarankan membaca dokumentasi secara singkat untuk lebih memahami di link berikut :
<https://google.github.io/mediapipe/>
https://google.github.io/mediapipe/getting_started/install.html

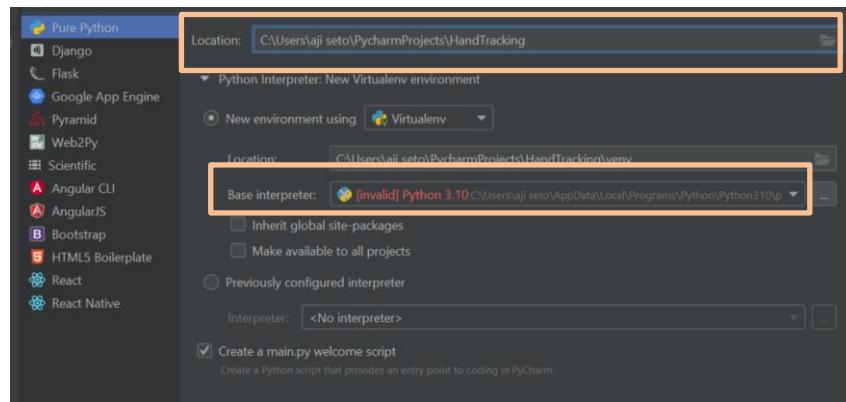


3. Perlu diingat bahwa **MediaPipe Python** secara resmi hanya bisa berjalan pada Sistem Operasi **64-bit** dan Python **versi 3.7** atau diatasnya dan OpenCV Versi **4++**.
4. Membuat project baru di **pycharm**: Open tools Pycharm setelah itu klik *new project*

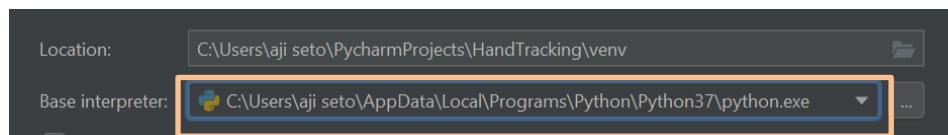


BKPM - (NAMA BPBM)

5. Beri nama, misal **HandTracking**

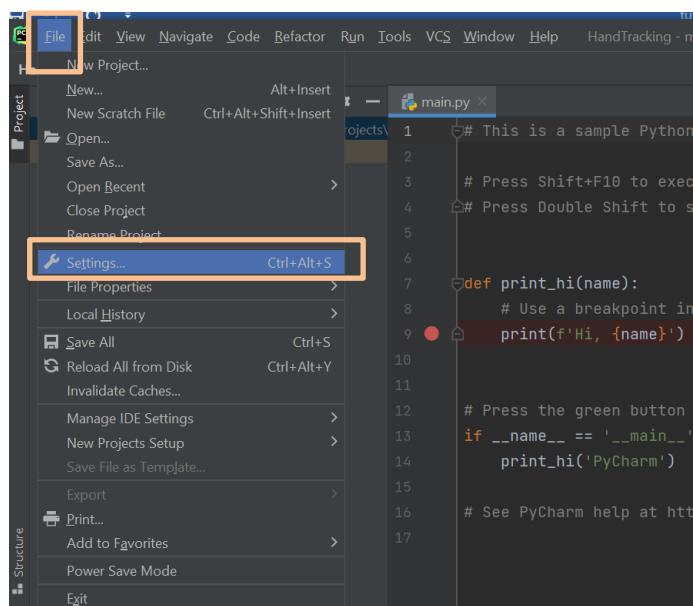


6. Dibagian **Base Interpreter**, pastikan seperti ini (merujuk lokasi python 37 – sesuai versi)

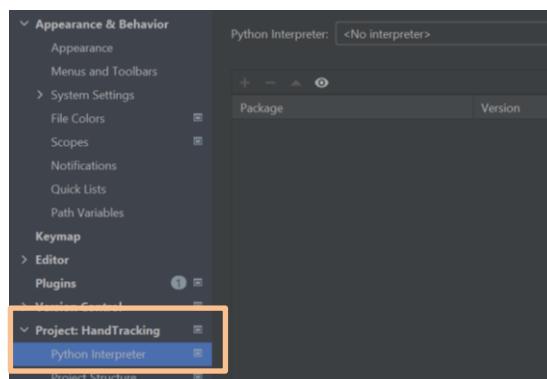


7. Lalu Klik Create di kanan bawah, tunggu pycharm membuat **virtual environment**

8. Menginstal OpenCV, **File→ Settings**

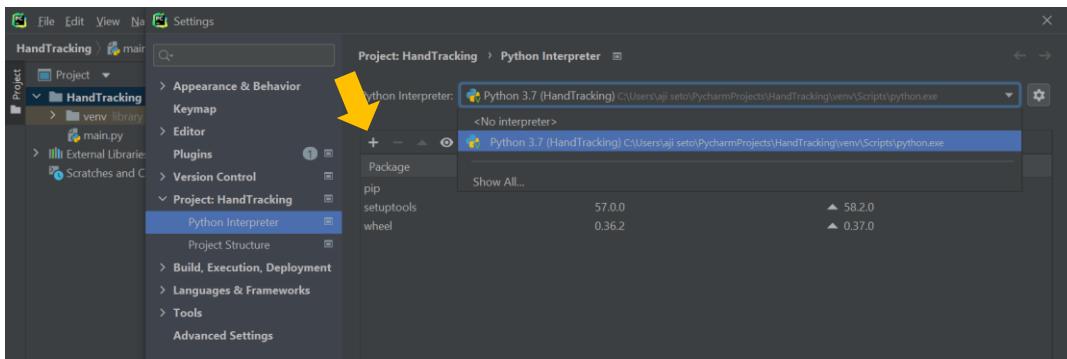


9. Pilih nama Project : **HandTracking**, kemudian pilih **Python Interpreter**



BKPM - (NAMA BPBM)

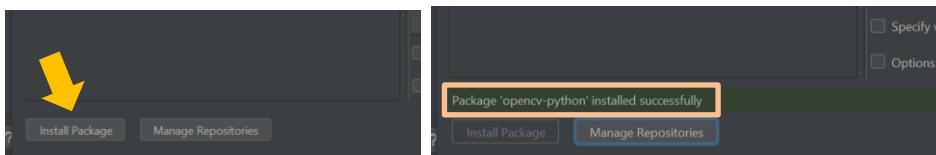
10. Klik Tanda +



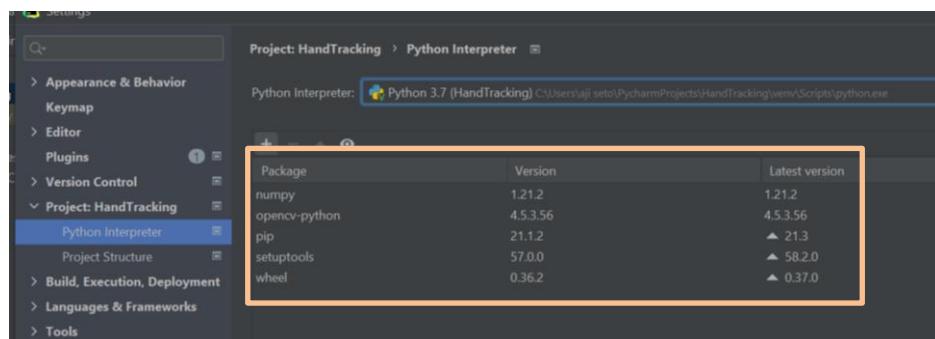
11. Ketikkan opencv-python di kolom searching



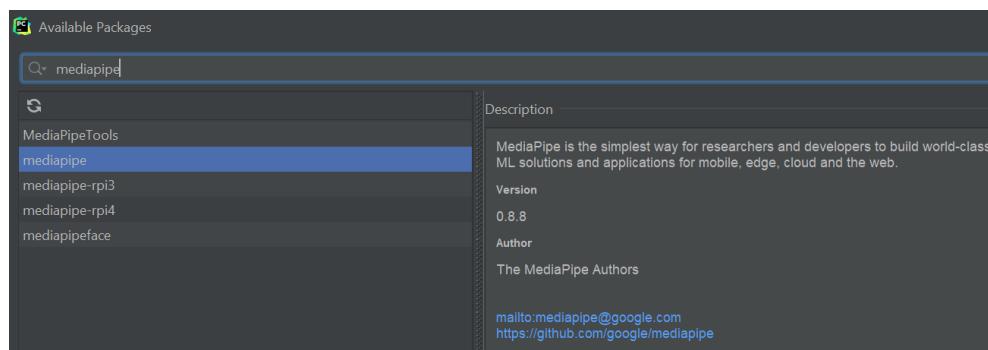
12. Klik Install Package, tunggu hingga Sukses



13. Maka akan ada penambahan Library, numpy dan opencv

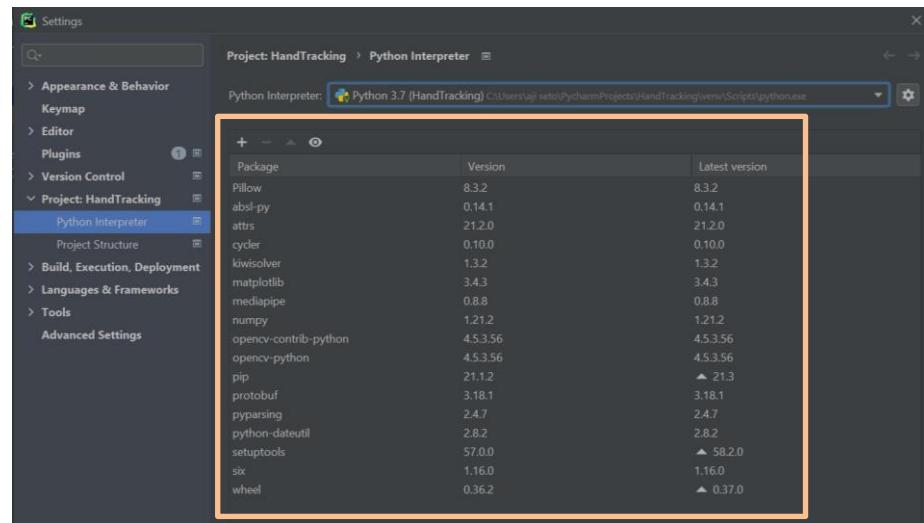


14. Lakukan hal yang sama langkah 10-13, menambah interpreter/library dengan Klik tanda + lalu ketikkan mediapipe di kolom pencarian, lakukan **Install Package**

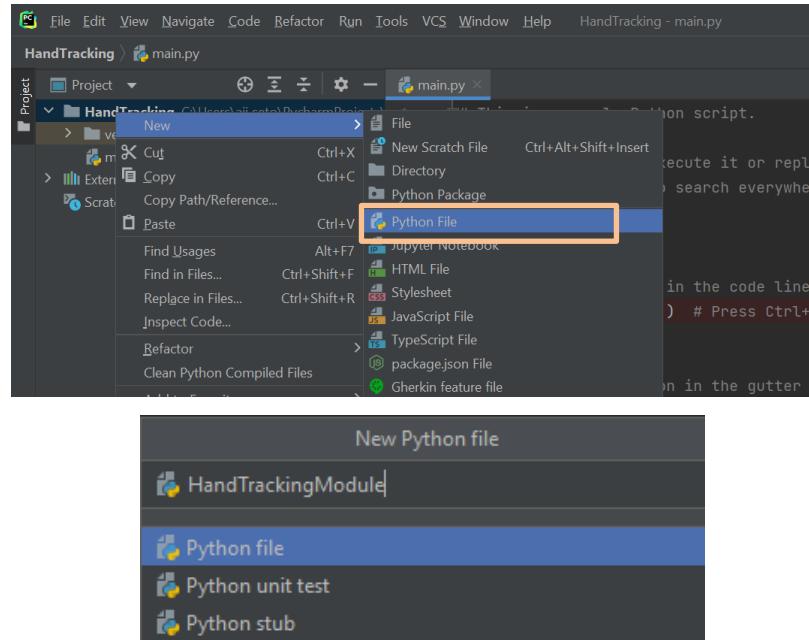


BKPM - (NAMA BPBM)

15. Maka akan bertambah beberapa library



16. Buat file python, Klik Kanan pada Project **HandTracking**, Klik **New** → **Python File**, Beri nama misal **HandTrackingModule**



17. Import opencv dan mediapipe kedalam source code

```
1 import cv2
2
3 import mediapipe as mp
4
5 import time
```

18. Implementasi opencv (show from webcam), silahkan tulis kode program berikut

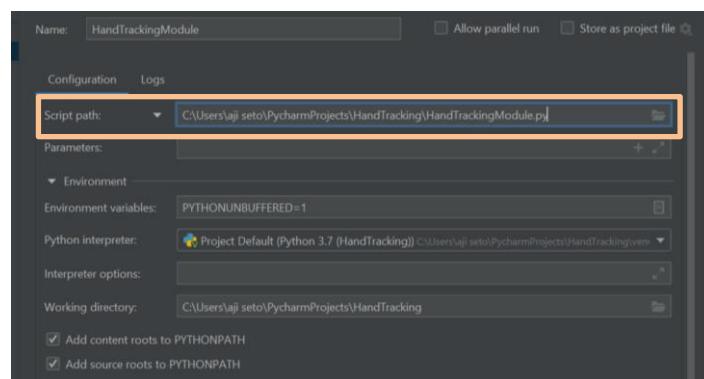
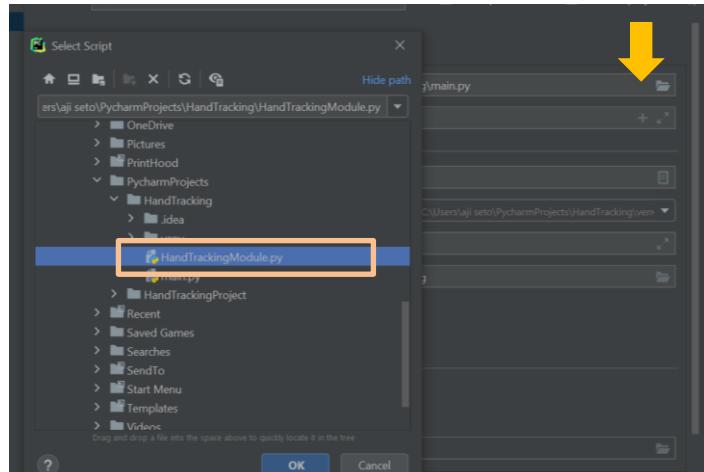
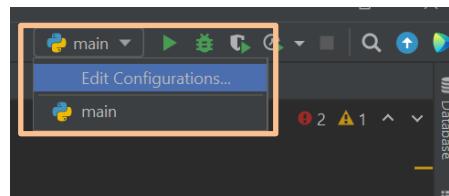
```
5 ccap = cv2.VideoCapture(0)
6
7 # time to count fps
8 pTime = 0 # previous
9 cTime = 0 # current
10
11 while True:
12     success, img = ccap.read() # read the image
13     imgRGB = cv2.cvtColor(img, cv2.COLOR_BGR2RGB) # convert BGR -> RGB
```

```

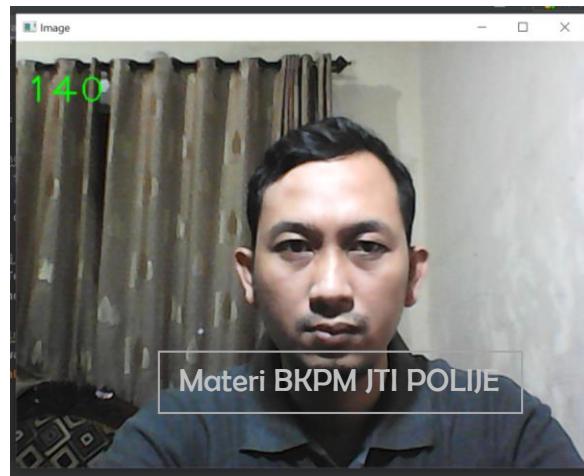
15     #counting fps
16     cTime = time.time()
17     fps = 1 / (cTime-pTime)
18     pTime = cTime
19
20     #drawing and show img
21     cv2.putText(img, str(int(fps)), (10, 70), cv2.FONT_HERSHEY_PLAIN, 3, (255, 0, 255), 2)
22     cv2.imshow('Image', img)
23
24     #delay 1ms and check to quit the looping when press 'q' on keyboard
25     if cv2.waitKey(1) & 0xff == ord("q"):
26         break

```

19. Kemudian untuk menjalankan program (run), dibagian kanan atas, arahkan eksekusi ke file **HandTrackingModule**, dengan cara berikut ini



20. Kemudian Klik tanda segitiga hijau (run), jika tidak ada kesalahan penulisan kode program, maka akan keluar tampilan sebagai berikut. Disebelah kiri atas tampilan webcam muncul angka yang menunjukkan **fps**. Untuk merubah jenis, ukuran dan warna font, silahkan rubah sintaks di **baris 21**



21. Implementasi **Mediapipe Show HandLandMark**, tambahkan beberapa baris kode program berikut ini ke program sebelumnya

```
5     ccap = cv2.VideoCapture(0)

6

7     mpHands = mp.solutions.hands # declaration before using mediapipe
8     hands = mpHands.Hands(min_detection_confidence=0.8,
9                           min_tracking_confidence=0.8) # module for hand tracking and detection
10    mpDraw = mp.solutions.drawing_utils # module for drawing landmark connection
11

12    # time to count fps
13    pTime = 0 # previous
14    cTime = 0 # current
15
16    while True:
17        success, img = ccap.read() # read the image
18        imgRGB = cv2.cvtColor(img, cv2.COLOR_BGR2RGB) # convert BGR -> RGB
19
20        results = hands.process(imgRGB) # preform the hand detection
21        # print(results.multi_hand_landmarks)
22

23
24         #detect if there is hand or not
25         if results.multi_hand_landmarks:
26             # detect multiple hands
27             for handLms in results.multi_hand_landmarks:
28                 # detect index ,position (ratio) landmark in image
29                 for id, lm in enumerate(handLms.landmark):
30                     # print(id, lm)
31                     h, w, c = img.shape
32                     cx, cy = int(lm.x * w), int(lm.y * h)
33                     print(id, cx, cy)
34                     if id == 0:
35                         cv2.circle(img, (cx, cy), 25, (0, 0, 255), cv2.FILLED)
36
37             # drawing connection landmark
38             mpDraw.draw_landmarks(img, handLms, mpHands.HAND_CONNECTIONS)
```

f. Hasil dan Pembahasan

Setelah dijalankan maka hasilnya seperti ini, untuk 1 atau 2 tangan



g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	

Acara 57-60

Pokok Bahasan : Proyek Akhir Workshop PCV

Acara Praktikum/Praktik : Minggu 16/1-4

Tempat : Laboratorium Multimedia Cerdas

Alokasi Waktu : 4 x 100 Menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

- 1) Mahasiswa mampu mendokumentasikan perencanaan, implementasi dan pengujian proyek computer vision.
- 2) Mahasiswa mampu mendemonstrasikan dan mempresentasikan hasil kerja workshop.

b. Indikator

Kemampuan mahasiswa dalam merencanakan, mengimplementasikan, menguji, melaporkan dan mempresentasikan hasil kerja workshop pengolahan citra dan vision.

c. Dasar Teori

Proyek Akhir Workshop Pengolahan Citra Vision merupakan produk aplikasi yang diharapkan bisa menjadi solusi bagi persoalan dikehidupan masyarakat. Produk aplikasi yang menerapkan teknik-teknik pengolahan citra dan metode/algoritma kecerdasan buatan sehingga aplikasi menjadi sebuah sistem *computer vision* yang mampu mengolah data masukan (citra/video), memproses serta mengambil kesimpulan secara benar. Setiap produk Proyek akhir yang dikerjakan mahasiswa secara berkelompok ini nantinya akan dipamerkan dalam agenda rutin TIF Exhibition.

d. Alat dan Bahan

1. Laptop
2. Kertas A4 & Pulpen
3. Aplikasi *Matlab* atau Pycharm
4. Aplikasi Ms. Word
5. Aplikasi Ms. Powerpoint

e. Prosedur Kerja

1. Mahasiswa membentuk kelompok 4-5 orang (disesuaikan dengan instruksi dosen pengampu)
2. Memilih permasalahan yang akan diangkat dalam proyek akhir workshop dan melakukan bimbingan kepada dosen pengampu.
3. Proses mengembangkan aplikasi *computer vision*.

4. Aplikasi yang sudah jadi didemonstrasikan kepada tim dosen pengampu.
5. Setiap kelompok juga wajib menyusun laporan dengan format sebagai berikut:
 - a) Ditulis dalam dokumen ukuran **A4, Spasi 1,5**
 - b) **Margin** dokumen atas 4 cm, kiri 4 cm, bawah dan kanan = 3cm
 - c) Halaman Judul (**Cover**), berisi
 - Judul Aplikasi [Font Arial, 14, Bold]
 - Logo Polije berwarna
 - Nama Ketua, Anggota Tim beserta NIM [Font Arial, 12, Bold]
 - Tulisan [Font Arial, 12, Bold]

Proyek Workshop Sistem Cerdas
Program Studi D4 Teknik Informatika
Jurusan Teknologi Informasi
Politeknik Negeri Jember
(Tahun)

- d) Bab 1 Pendahuluan [Font Arial, 11, Bold, Capital]
 - 1.1 Latar Belakang [Font Arial, 11, Bold]

(Buat Deskripsi min 3 paragraf ttg mengapa aplikasi Anda harus dibuat tuliskan masalah yg akan diselesaikan bisa diperkuat dengan data/fakta dari artikel/majalah/buletin/ dll yg bisa di pertanggung jawabkan, ide solusi boleh merujuk jurnal/buku) → [Font Arial, 11, spasi 1,5]
 - 1.2 Pustaka [Font Arial, 11, Bold]

(Bagian ini berisi beberapa Teori dasar yg mendukung proyek Kalian, Misal Teori ttg studi kasusnya/masalah, teori ttg metode/Algoritma nya) Boleh di dukung dg Gambar dan Rumus → [Font Arial, 11, spasi 1,5]
- e) Bab 2 Rancangan [Font Arial, 11, Bold, Capital]
 - 2.1 Data

(bagian ini sajikan ttg sumber data, jumlah data, bisa menggunakan grafik untuk menyajikan data, dan contoh data image) → [Font Arial, 11, spasi 1,5]
 - 2.2 Rancangan Sistem

(Bagian ini berisi penjelasan mengenai input-proses-output ; Deskripsi kan inputnya seperti apa, variabel brapa dn apa saja, Proses nya menggunakan Algoritma/Metode apa, dan hasil/output yg diharapkan bagaimana). Bisa menggunakan model Blok Diagram atau Flowchart System + deskripsi → [Font Arial, 11, spasi 1,5]

- f) Bab 3 Hasil [Font Arial, 11, Bold, Capital]
(pemberian nomor sub-bab bebas 3.1-3.n)
(Bagian ini diawali menjelaskan spesifikasi software yang dipakai; Sajikan tangkapan layar/capture User interface, diikuti source code lengkap, disajikan per bagian/fungsi/method - Disertai Penjelasan; Hasil pengujian, Berikan deskripsi/penjelasan yang baik
→ [Font Arial, 11, spasi 1,5]
- g) Daftar Pustaka [Font Arial, 11, Bold, Capital]
Sertakan semua sumber referensi berupa Artikel/Buku/Buletin/Majalah/Url Web (termasuk jika gambar/obyek ada yang Anda ambil dari source orang lain) → [Font Arial, 11, spasi 1,5]

f. Hasil dan Pembahasan

Hasil proyek akhir berupa : Aplikasi *computer vision*, laporan, file presentasi, serta bukti keikutsertaan dalam pameran (TIF Exhibition)

g. Rubrik Penilaian

No	Item Penilaian	Kriteria Penilaian	Bobot Penilaian	Penilaian
1.	Kecakapan Pembelajaran	Mengaplikasikan pengetahuan yang sudah didapatkan dengan mengidentifikasi permasalahan utama dan mempertimbangkan kerumitan yang ada	30%	
2.	Literasi	Sumber referensi dalam mengimplementasikan proyek	10%	
3.	Produk	Capaian akhir produk sesuai target proyek	40%	
4.	Presentasi & Laporan	Menyajikan informasi yang runtut, lengkap dan jelas dalam bentuk laporan serta penyampaian waktu presentasi	20%	
Total			100%	