

**IDENTIFIKASI TINGKAT KEMATANGAN PADI MENGGUNAKAN METODE
LINEAR DISCRIMINANT ANALYSIS**

TUGAS AKHIR

Sebagai syarat untuk memperoleh gelar sarjana S-1 di Program Studi Informatika, Jurusan
Informatika, Fakultas Teknik Industri Universitas Pembangunan Nasional “Veteran”
Yogyakarta



Disusun Oleh :
Nabila Nurul Aini
123160002

**PROGRAM STUDI INFORMATIKA
JURUSAN INFORMATIKA
FAKULTAS TEKNIK INDUSTRI
UNIVERSITAS PEMBANGUNAN NASIONAL “VETERAN”
YOGYAKARTA
2022**

HALAMAN PENGESAHAN PEMBIMBING

IDENTIFIKASI TINGKAT KEMATANGAN PADI
MENGUNAKAN METODE *LINEAR DISCRIMINANT ANALYSIS*



Dr. Heriyanto, A.Md., S.Kom., M.Cs.
NIDN. 0508067703

HALAMAN PENGESAHAN PENGUJI

IDENTIFIKASI TINGKAT KEMATANGAN PADI
MENGUNAKAN METODE *LINEAR DISCRIMINANT ANALYSIS*



Dr. Heriyanto, A.Md, S.Kom, M.Cs.
NIDN. 0508067703

Mangaras Yanu Florestiyanto, S.T., M.T.
NIDN. 0521018201

SURAT PERNYATAAN KARYA ASLI TUGAS AKHIR

Sebagai mahasiswa Program Studi Informatika Fakultas Teknik Industri Universitas Pembangunan Nasional "Veteran" Yogyakarta, yang bertanda tangan di bawah ini, saya:

Nama : Nabila Nurul Aini

NIM : 123160002

Menyatakan bahwa karya ilmiah saya yang berjudul:

IDENTIFIKASI TINGKAT KEMATANGAN PADI MENGGUNAKAN METODE LINEAR DISCRIMINANT ANALYSIS

Merupakan karya asli saya dan belum pernah dipublikasikan dimanapun. Apabila ada di kemudian hari, karya saya disinyalir bukan merupakan karya asli saya, maka saya bersedia menerima konsekuensi apa pun yang diberikan Program Studi Informatika Fakultas Teknik Industri Universitas Pembangunan Nasional "Veteran" Yogyakarta kepada saya.

Demikian surat pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Yogyakarta
Pada tanggal : 15 Desember 2022
Yang menyatakan,



Nabila Nurul Aini
123160002

PERNYATAAN BEBAS PLAGIAT

Saya yang bertanda tangan di bawah ini:

Nama : Nabila Nurul Aini

NIM : 123160002

Fakultas/Prodi: Teknik Industri/Informatika

Dengan ini saya menyatakan bahwa judul Tugas Akhir:

***IDENTIFIKASI TINGKAT KEMATANGAN PADI MENGGUNAKAN METODE
LINEAR DISCRIMINANT ANALYSIS***

Adalah hasil kerja saya sendiri dan benar bebas dari plagiat kecuali cuplikan serta ringkasan yang terdapat di dalamnya setelah saya jelaskan sumbernya (sitasi) dengan jelas. Apabila pernyataan ini terbukti tidak benar maka saya bersedia menerima sanksi sesuai peraturan Mendiknas RI No. 17 Tahun 2000 dan Pernyataan dan Perundang-undangan yang berlaku.

Demikian surat pernyataan ini saya buat dengan penuh tanggung jawab.

Yogyakarta, 15 Desember 2022

Yang membuat pernyataan



Nabila Nurul Aini
12316002

HALAMAN PERSEMBAHAN

Alhamdulillahirabbil alamin

“Dengan segala puji syukur kehadiran Allah SWT yang telah melimpahkan karunia-Nya, sehingga saya bisa menyelesaikan tugas akhir ini. Tugas akhir ini saya persembahkan kepada orang-orang tercinta bapak, ibu, adik serta kepada sahabat-sahabat, dan teman-teman angkatan 2016 Informatika. Terimakasih atas segala dukungan dan doanya.”

Last but not least, I wanna thank me I wanna thank me for believing in me,
I wanna thank me for doinng all this hard work I wanna thank me for having no days off.
I wanna thank me for never quitting I wanna thank me for always being a giver and tryna
give more than I receive.

ABSTRAK

Indonesia merupakan negara tropis yang kaya akan keanekaragaman hayati. Beraneka ragam tanaman pangan dan buah-buahan dikembangkan untuk menghasilkan varietas-varietas unggul baru. Produksi tanaman pangan terus ditingkatkan karena kebutuhan masyarakat untuk mengonsumsi tanaman pangan juga terus bertambah. Salah satu sumber pangan paling besar adalah padi. Tanaman padi saat ini menjadi sumber makanan pokok bagi masyarakat di Indonesia. Menurut Direktorat Jenderal Tanaman Pangan dan Hortikultura tujuan dari pemanenan padi adalah untuk memperoleh gabah dari lapangan dengan tingkat kematangan yang optimal, mencegah kerusakan dan kehilangan hasil seminimal mungkin. Kematangan dan kadar air yang terdapat dalam padi memberikan kontribusi yang sangat besar terhadap kualitas beras. Waktu panen yang tidak tepat menyebabkan banyak kerugian, diantaranya adalah tingginya jumlah gabah yang tidak matang dengan sempurna, kualitas gabah yang rendah, kemungkinan serangan penyakit yang lebih besar ketika disimpan serta bobot gabah yang lebih ringan yang akan mengakibatkan kerugian bagi para petani (Mustikasucy, 2020). Produksi beras saat ini lebih menantang karena adanya perubahan iklim, sehingga penting untuk menentukan waktu yang optimal untuk memanen padi. Terdapat beberapa perubahan morfologi yang terjadi pada tanaman padi untuk memperlihatkan tingkat kematangannya. Perubahan morfologi yang terjadi diantaranya ada perubahan bentuk, ukuran, berat, warna, kadar air dan karakteristik kimia (Ansari, 2022).

Pada penelitian ini, metode *Linear Discriminant Analysis* dipilih karena metode ini dikembangkan untuk mengatasi masalah pada ukuran sampel yang kecil dengan cara memaksimalkan pemisahan antara pemisahan kelas (*class scatters*) dan meminimalkan pemisahan dalam pemisahan kelas (*with-in class scatters*) untuk menghilangkan masalah *overfitting* dengan cara mengurangi kesalahan ketika memperkirakan parameter dan mengurangi waktu komputasi klasifikasi (Shambulinga & Sadashivappa, 2020).

Penggunaan metode Linear Discriminant Analysis dalam mengidentifikasi tingkat kematangan padi memberikan hasil yang baik,. Berdasarkan hasil pengujian sistem dengan menggunakan metode *Linear Discriminant Analysis* dan pembagian data 70% : 30% dengan hasil 91%, 80% : 20% dengan hasil 93%, dan 90% : 10% dengan hasil 91%. Dari pembagian data *training* dan data *testing* tersebut diperoleh nilai akurasi tertinggi pada pembagian data 80% : 20%. Dengan akurasi sebesar 93%.

Kata kunci: Padi, Kematangan, LDA

ABSTRACT

Indonesia is a tropical country rich in biodiversity. A wide variety of food and fruit plants are developed to produce new superior varieties. Production of food crops continues to be increased because people's need to consume food crops also continues to grow. One of the biggest food sources is rice. Rice plants are currently a staple food source for people in Indonesia. According to the Directorate General of Food Crops and Horticulture, the purpose of harvesting rice is to obtain grain from the field at an optimal level of maturity, preventing damage and minimizing yield loss. Maturity and moisture content contained in rice contribute greatly to the quality of rice. Improper harvest time causes many losses, including the high amount of grain that is not perfectly ripe, low grain quality, greater possibility of disease attack when stored and lighter grain weight which will result in losses for farmers (Mustikasucy, 2020). Rice production is currently more challenging due to climate change, so it is important to determine the optimal time to harvest rice. There are several morphological changes that occur in rice plants to show their level of maturity. Morphological changes that occur include changes in shape, size, weight, color, water content and chemical characteristics (Ansari, 2022).

In this study, the Linear Discriminant Analysis method was chosen because this method was developed to solve problems with small sample sizes by maximizing the separation between class scatters and minimizing within-in class scatters to eliminate overfitting problems by how to reduce errors when estimating parameters and reduce classification computing time (Shambulinga & Sadashivappa, 2020).

The use of the Linear Discriminant Analysis method in identifying the level of maturity of rice gives good results. Based on the results of system testing using the Linear Discriminant Analysis method and data sharing 70%: 30% with 91% results, 80%: 20% with 93% results, and 90%: 10% with 91% results. From the distribution of training data and testing data, the highest accuracy value was obtained for the distribution of data 80%: 20%. With an accuracy of 93%.

Keywords: Rice, Maturity, LDA

KATA PENGANTAR

Puji syukur kehadirat Allah SWT yang senantiasa memberikan karunia, rahmat serta hidayah-nya sehingga penulis dapat menyelesaikan tugas akhir yang berjudul “IDENTIFIKASI TINGKAT KEMATANGAN PADI MENGGUNAKAN METODE LINEAR DISCRIMINANT ANALYSIS”. Tugas akhir ini merupakan salah satu syarat kelulusan untuk menyelesaikan pendidikan pada jenjang strata satu (S1) Program Studi Informatika Fakultas Teknik Industri Universitas Pembangunan Nasional “Veteran” Yogyakarta. Laporan ini tidak dapat terselesaikan tanpa bantuan dari berbagai pihak, baik secara langsung maupun tidak langsung. Pada kesempatan ini, penulis ingin menyampaikan ucapan terima kasih kepada :

1. Allah SWT yang senantiasa memberikan petunjuk dan kemudahan dalam penulisan tugas akhir.
2. Bapak saya Herri Koesdiyanto, Ibu saya Ambar Wahyuni, Adik saya Anissa Sephia Wulandari beserta kerabat dan saudara yang selalu memberikan do’a dan dukungan dalam menyelesaikan tugas akhir ini.
3. Bapak Dr. Heriyanto, A.Md., S. Kom., M. Cs selaku Ketua Program Studi Informatika Universitas Pembangunan Nasional “Veteran” Yogyakarta
4. Bapak Dessyanto Boedi Prasetyo, S.T., M.T.. dan Ibu Juwairiah, S.Si.,M.T. selaku dosen pembimbing saya atas ilmu dan bimbingannya yang telah diberikan kepada penulis.
5. Bapak Dr. Heriyanto, A.Md., S.Kom., M.Cs. dan Bapak Mangaras Yanu Florestiyanto, S.T., M.T.selaku dosen penguji.
6. Bapak Heru Cahya Rustamadji, S.T., M.T. dan Bapak Dessyanto Boedi Prasetyo,S.T., M.T. selaku dosen wali.
7. Seluruh dosen dan staf tata usaha Program Studi Teknik Informatika Universitas Pembangunan Nasional “Veteran” Yogyakarta.
8. Rafly Pradana Putra yang senantiasa telah menemani, memotivasi dan menyemangati selama pengerjaan tugas akhir.
9. Teman – teman Informatika Angkatan 2016 yang telah mendukung dan memberikan bantuan selama pengerjaan tugas akhir.
10. Seluruh pihak yang telah membantu penyusunan laporan tugas akhir ini dan tidak dapat disebutkan satu persatu.

Penulisan tugas akhir ini masih banyak kekurangan dan kesalahan sehingga penulis mengharapkan adanya saran, tanggapan dan kritik yang bersifat membangun. Semoga tugas akhir ini dapat menambah pengetahuan pembaca dan semua pihak.

Yogyakarta, 15 Desember 2022

Penulis

DAFTAR ISI

HALAMAN PENGESAHAN PEMBIMBING	ii
HALAMAN PENGESAHAN PENGUJI.....	iii
SURAT PERNYATAAN KARYA ASLI TUGAS AKHIR.....	iv
PERNYATAAN BEBAS PLAGIAT	v
HALAMAN PERSEMBAHAN.....	vi
ABSTRAK	vii
ABSTRCT	viii
KATA PENGANTAR.....	ix
DAFTAR ISI	x
DAFTAR TABEL	xii
DAFTAR GAMBAR.....	xiii
DAFTAR MODUL	xiv
BAB I PENDAHULUAN	1
1.1 Latar Belakang Masalah	1
1.2 Rumusan Masalah.....	3
1.3 Batasan Masalah	3
1.4 Tujuan Penelitian	4
1.5 Manfaat Penelitian	4
1.6 Metodologi Penelitian dan Pengembangan Sistem.....	4
1.6.1 Metodologi Penelitian	4
1.6.2 Metode Pengembangan Sistem.....	4
BAB II TINJAUAN PUSTAKA	6
2.1 Padi	6
2.1.1 Masa Pertumbuhan Padi	6
2.1.2 Varietas Padi.....	7
2.1.3 Jenis Budidaya Padi.....	8
2.2 Citra	8
2.3 Citra digital	9
2.4 Pengolahan Citra.....	12
2.4.1 Akuisisi Citra.....	13
2.4.2 Image Augmentation	13
2.4.3 Resize	14
2.4.4 Pengolahan Warna.....	14
2.4.5 Perataan Histogram (Histogram Equalization).....	16
2.4.6 Segmentasi Citra.....	16
2.5 Ekstraksi Fitur.....	17
2.6 Confusion Matrix	17
2.7 Computer Vision.....	17
2.8 Linear Discriminant Analysis	18
2.9 Tinjauan Literatur	19
BAB III METODOLOGI PENELITIAN DAN PENGEMBANGAN SISTEM.....	24
3.1 Metodologi Penelitian.....	24
3.1.1 Pengumpulan Data.....	25

3.1.2	Data Preprocessing	27
3.1.3	Ekstraksi Fitur	32
3.1.4	Linear Discriminant Analysis	33
3.1.5	Model Linear Discriminant Analysis	37
3.1.6	Pengujian	37
3.2	Metode Pengembangan Sistem	37
3.2.1	Communication (Project Initiation & Requirement Gathering)	38
3.2.2	Planning (Estimating, Scheduling, Tracking)	38
3.2.3	Modeling (Analysis & Design)	39
3.2.4	Construction (Code & Test)	41
BAB IV HASIL DAN PEMBAHASAN		43
4.1	Hasil dan Pembahasan	43
4.1.1	Preprocessing	45
4.1.2	Ekstraksi Fitur	47
4.1.3	Linear Discriminant Analysis	48
4.2	Pengujian	49
4.3	Pembahasan	52
BAB V PENUTUP		56
5.1	Kesimpulan	56
5.2	Saran	56
DAFTAR PUSTAKA		57

DAFTAR TABEL

Tabel 2.1 Kelas <i>Oryza sativa</i>	6
Tabel 3.1 Tabel contoh hasil akuisisi citra.	26
Tabel 3.2 Contoh image dari data hasil akuisisi citra setelah <i>labelling</i>	27
Tabel 3.3 Contoh hasil augmentasi	28
Tabel 3.4 Tabel contoh nilai HSV, label, dan kelas.	34
Tabel 3.5 Fungsi diskriminan	37
Tabel 3.6 Hasil perbandingan nilai f1 dan f2p	37
Tabel 3.7 Kebutuhan non-fungsional	39
Tabel 3.8 Kebutuhan non-fungsional.	39
Tabel 3.9 Perancangan Pengujian.....	42
Tabel 4.1 Tabel pengujian pembagian data <i>training</i> dan data <i>testing</i> 70 % : 30%	49
Tabel 4.2 Tabel pengujian pembagian data <i>training</i> dan data <i>testing</i> 80 % : 20%	50
Tabel 4.3 Tabel pengujian pembagian data <i>training</i> dan data <i>testing</i> 90% : 10%	51
Tabel 4.4 Tabel perbandingan akurasi.....	52
Tabel 4.5 Tabel percobaan identifikasi.....	55

DAFTAR GAMBAR

Gambar 1.1 Tahapan metode <i>Waterfall</i>	5
Gambar 2.1 Representasi citra digital.....	10
Gambar 2.2 Citra berwarna	10
Gambar 2.3 Binary image.....	11
Gambar 2.4 <i>Grayscale image</i>	12
Gambar 2.5 Citra berindeks.....	12
Gambar 2.6 Contoh <i>augmentation</i>	14
Gambar 2.7 Model Warna RGB (Gonzalez & Woods, 2018).....	15
Gambar 2.8 Hubungan antara computer vision, pengolahan citra dan pengenalan pola.....	18
Gambar 2.9 Hierarki dalam computer vision	18
Gambar 3.1 Metodologi penelitian.....	24
Gambar 3.2 Hasil pembacaan dengan <i>lux meter</i>	26
Gambar 3.4 <i>Flowchart preprocessing</i>	28
Gambar 3.5 Contoh hasil <i>resize image</i>	29
Gambar 3.6 Contoh hasil konversi RGB ke HSV.	31
Gambar 3.7 Contoh hasil <i>histogram equalization</i>	31
Gambar 3.8 Contoh hasil segmentasi.	32
Gambar 3.9 <i>flowchart</i> ekstraksi fitur.....	33
Gambar 3.10 Contoh hasil ekstraksi fitur.....	33
Gambar 3.11 <i>flowchart</i> identifikasi <i>Linear Discriminant Analysis</i>	34
Gambar 3.12 <i>Flowchart</i> metode pengembangan sistem.	38
Gambar 3.13 Perancangan arsitektur sistem.....	40
Gambar 3.14 Perancangan antar muka bagian pelatihan.....	41
Gambar 3.15 Perancangan antar muka bagian identifikasi.	41
Gambar 4.1 Tampilan antarmuka halaman awal.	43
Gambar 4.2 Antarmuka awal halaman <i>training</i>	43
Gambar 4.3 Tampilan antarmuka halaman hasil <i>training</i>	44
Gambar 4.4 Tampilan antarmuka halaman <i>identify</i>	44
Gambar 4.5 Hasil identifikasi.....	45
Gambar 4.6 Hasil <i>preprocessing</i>	47
Gambar 4.7 Hasil ekstraksi fitur.....	48
Gambar 4.8 <i>Confusion matrix</i> 70% : 30%.....	50
Gambar 4.9 <i>Confusion matrix</i> 80% : 20%	51
Gambar 4.10 <i>Confusion matrix</i> 90% : 10%.....	52
Gambar 4.11 Gabungan matang dan mentah dimana bulir padi matang lebih banyak dibandingkan dengan mentah	53
Gambar 4.12 Gabungan matang dan mentah dimana bulir padi mentah lebih banyak dibandingkan dengan matang	54
Gambar 4.13 Gabungan matang dan mentah dengan jumlah yang sama	54
Gambar 4.14 Gabungan matang dan mentah dengan jumlah yang sama.....	55

DAFTAR MODUL

Modul 4.1 Augmentasi Data.....	45
Modul 4.2 <i>Preprocessing</i>	46
Modul 4.3 Ekstraksi Fitur.....	47
Modul 4.4 Linear Discriminant Analysis	48
Modul 4.5 <i>Confusion Matrix</i>	49

BAB I PENDAHULUAN

1.1 Latar Belakang Masalah

Indonesia merupakan negara tropis yang kaya akan keanekaragaman hayati. Beraneka ragam tanaman pangan dan buah-buahan dikembangkan untuk menghasilkan varietas-varietas unggul baru. Produksi tanaman pangan terus ditingkatkan karena kebutuhan masyarakat untuk mengonsumsi tanaman pangan juga terus bertambah. Salah satu sumber pangan paling besar adalah padi. Tanaman padi saat ini menjadi sumber makanan pokok bagi masyarakat di Indonesia. Menurut Direktorat Jenderal Tanaman Pangan dan Hortikultura tujuan dari pemanenan padi adalah untuk memperoleh gabah dari lapangan dengan tingkat kematangan yang optimal, mencegah kerusakan dan kehilangan hasil seminimal mungkin. Kematangan dan kadar air yang terdapat dalam padi memberikan kontribusi yang sangat besar terhadap kualitas beras. Waktu panen yang tidak tepat menyebabkan banyak kerugian, diantaranya adalah tingginya jumlah gabah yang tidak matang dengan sempurna, kualitas gabah yang rendah, kemungkinan serangan penyakit yang lebih besar ketika disimpan serta bobot gabah yang lebih ringan yang akan mengakibatkan kerugian bagi para petani (Mustikasucy, 2020). Produksi beras saat ini lebih menantang karena adanya perubahan iklim, sehingga penting untuk menentukan waktu yang optimal untuk memanen padi. Terdapat beberapa perubahan morfologi yang terjadi pada tanaman padi untuk memperlihatkan tingkat kematangannya. Perubahan morfologi yang terjadi diantaranya ada perubahan bentuk, ukuran, berat, warna, kadar air dan karakteristik kimia (Ansari, 2022).

Secara tradisional, petani menentukan umur dan waktu panen dengan menghitung jumlah hari setelah waktu tanam. Apabila padi telah berumur 120 hari, maka petani akan melakukan panen pada periode ini. Tanaman padi mengalami beberapa pertumbuhan yang signifikan, seperti bulir padi yang bertambah pada ukuran dan berat bulir padi serta warna daun yang berubah dari hijau menjadi kuning karena kandungan nitrogen yang ditransfer dari daun ke biji. Malai padi dewasa juga mengalami perubahan warna dari hijau menjadi coklat keemasan, akan tetapi kondisi iklim yang basah dapat menghambat proses degreening pada malai padi (Ansari, 2022). Namun, hal ini berdasarkan pada pengalaman dan pengamatan menggunakan mata telanjang yang tidak selalu dapat diandalkan.

Beberapa metode untuk mengatasi masalah klasifikasi antara lain adalah metode *K-Nearest Neighbor* (KNN), *Principal Component Analysis* (PCA), dan *Linear Discriminant Analysis* (LDA). Metode *K-Nearest Neighbor* memiliki kelebihan yaitu perhitungan yang sederhana dan komputasi yang cepat serta metode *K-Nearest Neighbor* ini mampu melakukan *training* dengan menggunakan data yang *noisy*. Akan tetapi metode *K-Nearest Neighbor* akan bekerja lebih efektif apabila menggunakan data latih yang jumlahnya besar (Hanafi, Fadillah, & Insan, 2019) *Principal Component Analysis* (PCA) mampu mengurangi jumlah dimensi dengan menggunakan transformasi linier dari yang ruang berdimensi tinggi menjadi ruang berdimensi rendah. Akan tetapi PCA kurang baik dalam melakukan pemisahan kelas sehingga akurasi yang didapatkan rendah (Azman, Khairunniza-Bejo, Ismail, & Wayayok, 2014) *Linear Discriminant Analysis* merupakan sebuah metode *supervised* yang bekerja yang mereduksi dimensi yang bertujuan untuk

memproyeksikan kumpulan data ke ruang berdimensi lebih rendah dengan pemisahan kelas yang lebih baik untuk menghindari *overfitting* dan juga mengurangi biaya komputasi.

Penelitian serupa juga dilakukan oleh Azman, N., et al dalam Anami, Naveen, Hanamaratti (2017) dengan judul estimasi kematangan padi menggunakan ruang warna RGB, dalam penelitian ini menggunakan 15 sampel padi dengan jumlah 50 gram. Nilai *threshold* yang digunakan sebesar 0,34 dan 0,32 untuk memisahkan gambar dan background. Nilai *threshold* tersebut didapatkan dari proses *trial* dan *error*. Penelitian serupa telah dilakukan oleh Anggi Diatma Styandi (2019) tentang Klasifikasi Umur Padi Berdasarkan Data Sensor Warna Dengan Menggunakan Metode KNN. Dalam penelitian ini menggunakan arduino uno dengan sensor TCS3200 dengan pengujian sebanyak 20 kali terhadap perubahan warna RGB. Dari pengujian tersebut dihasilkan perubahan yang tidak terlalu signifikan karena memiliki nilai yang mendekati dalam setiap pengujian padi. Dalam penelitian yang dilakukan oleh Anami, Naveen, Hanamaratti (2017) yang berjudul Klasifikasi Penuaan Bulir Padi Menggunakan Fitur Warna RGB dan HSI melakukan pengujian dengan menggunakan Multilayer *Backpropagation Neural Network* dengan rata-rata akurasi klasifikasi periode penuaan sebesar 71,55%. Dalam penelitian ini untuk meningkatkan hasil akurasi, digunakan sebuah metode pengurangan fitur. Sepuluh fitur warna yang dikurangi dari ruang warna RGB dan HSI menghasilkan akurasi klasifikasi periode penuaan dengan rata-rata 93,07%. Faktor penyimpanan dalam penelitian ini dapat mempengaruhi klasifikasi penuaan. Dalam penelitian yang dilakukan oleh Imam Adi Sulistyio (2017) yang berjudul Sistem Deteksi Panen Padi Berdasar Warna Daun Menggunakan *Fuzzy C-Means Clustering* didapatkan akurasi sebesar 94,11% dengan rata-rata nilai hue untuk padi yang belum siap panen sebesar 0,23-0,34 dan nilai *hue* untuk data padi siap panen sebesar 0,11-0,21, dalam penelitian ini peneliti memberikan saran agar tidak hanya daun padi yang dijadikan acuan dalam menentukan masa panen namun bisa ditambahkan bulir padi sebagai acuan untuk menentukan masa panen. Penelitian serupa lainnya yang telah dilakukan adalah *color vision* untuk menentukan kematangan padi oleh Cho Lip Haw et al dalam Wang (2021), pada penelitian ini mengambil nilai hue daun dan bulir padi yang dibagi menjadi tiga bagian yaitu bagian terminal, middle, dan basal. Hal ini dilakukan karena ruang warna HSV memiliki sifat yang tidak berubah yang merupakan kebalikan dari ruang warna RGB yang berubah. Dari penelitian ini data dari bagian basal memberikan korelasi yang baik dibandingkan dengan daun. Daun tidak dapat dijadikan indikator yang baik untuk menentukan kematangan padi karena daun mengering lebih cepat dan beberapa daun akan menggulung hingga berbentuk seperti jarum. Selain itu, daun cenderung menjadi inang bagi hama padi. Hama padi menyerang daun dan mengakibatkan daun menjadi keriting dan berwarna putih pucat.

Pada penelitian ini, metode *Linear Discriminant Analysis* dipilih karena metode ini dikembangkan untuk mengatasi masalah pada ukuran sampel yang kecil dengan cara memaksimalkan pemisahan antara pemisahan kelas (*class scatters*) dan meminimalkan pemisahan dalam pemisahan kelas (*within class scatters*) untuk menghilangkan masalah *overfitting* dengan cara mengurangi kesalahan ketika memperkirakan parameter dan mengurangi waktu komputasi klasifikasi (Shambulinga & Sadashivappa, 2020). Penelitian serupa juga dilakukan (Pereira et al., 2018), dalam penelitian ini pengklasifikasian

menggunakan metode *Linear Discriminant Analysis* memperoleh nilai akurasi sebesar 83% dengan menggunakan ekstraksi fitur RGB. Dalam penelitian ini menggunakan metode *Linear Discriminant Analysis* terhadap objek yang telah dipilih yaitu padi akan tetapi ruang warna RGB memiliki perbedaan warna yang tidak terlalu signifikan sehingga mempersulit dalam pengklasifikasian. Penelitian lain oleh Ghazali, Eko K. Subha, Galuh M., M. Burhannudin (2018) yang mengklasifikasikan kematangan tomat berdasarkan kelas mentah dan matang menggunakan metode *Linear Discriminant Analysis* yang memperoleh akurasi sebesar 93,10%. Metode *Linear Discriminant Analysis* ini banyak digunakan untuk pengenalan wajah (*face recognition*), sehingga dalam penelitian ini akan menerapkan *Linear Discriminant Analysis* dengan objek yang berbeda yaitu padi. Berdasarkan penelitian terdahulu *Linear Discriminant Analysis* memiliki beberapa kelebihan, diantaranya adalah waktu komputasi yang terhitung cepat serta mampu memberikan nilai akurasi yang cukup tinggi dan mampu bekerja dengan menggunakan jumlah sampel kecil (Shambulinga & Sadashivappa, 2020). Pada penelitian ini metode *Linear Discriminant Analysis* akan diuji dengan pengolahan citra untuk melakukan identifikasi dan melihat hasil akurasi dengan menggunakan metode *Linear Discriminant Analysis*.

Oleh karena itu, penelitian ini dilakukan untuk melakukan identifikasi tingkat kematangan padi menggunakan metode *Linear Discriminant Analysis* dan mengetahui nilai akurasi yang diperoleh menggunakan metode *Linear Discriminant Analysis*. Dalam penelitian ini menggunakan metode *Linear Discriminant Analysis* karena metode *Linear Discriminant Analysis* mampu memaksimalkan pemisahan antara pemisah kelas (*scatters class*) dan meminimalkan pemisahan dalam pemisahan kelas (*with-in class scatters*) sehingga hasil yang didapatkan lebih akurat. Dalam penelitian ini menggunakan data padi sejumlah 40 bulir padi setiap citra yang diperoleh secara langsung dengan menggunakan kamera dengan resolusi 12 megapixel, f/1.8 aperture, focal length 4 mm dengan pengambilan jarak 15 cm, yang kemudian dibagi menjadi data latih dan data uji.

1.2 Rumusan Masalah

Berdasarkan latar belakang tersebut maka permasalahan yang dapat dirumuskan, yaitu:

1. Bagaimana identifikasi padi menggunakan metode *Linear Discriminant Analysis*?
2. Bagaimana tingkat akurasi metode *Linear Discriminant Analysis* dalam identifikasi tingkat kematangan padi?

1.3 Batasan Masalah

Agar batasan dalam penelitian ini tidak terlalu luas, maka penelitian akan dibatasi sebagai berikut:

1. Sistem yang dibuat hanya mampu mengidentifikasi tingkat kematangan padi berdasarkan kelas matang dan mentah.
2. Data primer digunakan dalam penelitian ini dengan pengambilan bulir padi di sebuah sawah di Ambarawa
3. Jenis padi yang akan digunakan dalam penelitian ini adalah padi IR 64 dengan jumlah bulir padi sebanyak 40 bulir padi dalam setiap citra.

4. Citra yang digunakan dalam penelitian ini diambil dengan menggunakan kamera *smartphone* dengan spesifikasi resolusi 12 megapixel, *f/1.8 aperture, focal length 4mm* pada pukul 17.00 WIB.
5. Cahaya yang digunakan sebesar 1934 *lux*.
6. Jarak pengambilan gambar sepanjang 15 cm dari objek.
7. Ukuran piksel 3024 x 4032.

1.4 Tujuan Penelitian

Tujuan dilakukannya penelitian ini adalah sebagai berikut.

1. Mengetahui tingkat kematangan padi menggunakan metode *Linear Discriminant Analysis*.
2. Mengetahui dan menguji tingkat akurasi metode *Linear Discriminant Analysis* dalam proses identifikasi tingkat kematangan tanaman padi.

1.5 Manfaat Penelitian

Dengan dilakukannya penelitian ini diharapkan dapat mengetahui performa dari metode metode *Linear Discriminant Analysis* dalam mengidentifikasi tingkat kematangan padi.

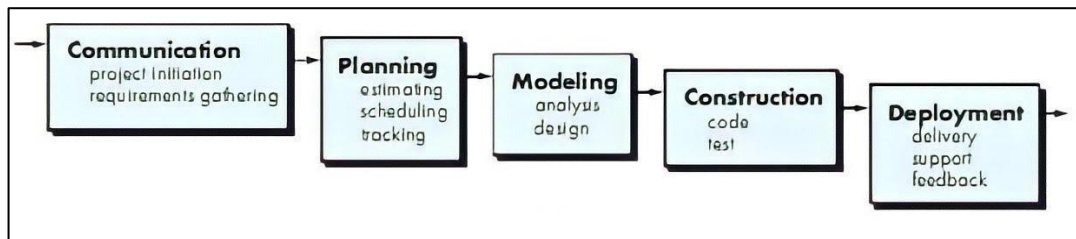
1.6 Metodologi Penelitian dan Pengembangan Sistem

1.6.1 Metodologi Penelitian

1. Pengumpulan Data
Alur metodologi penelitian yang dilakukan adalah sebagai berikut:
 - a) Studi Literatur
 - b) Observasi
 - c) Wawancara
 - d) Akuisisi Citra
 - e) *Labelling*
2. *Preprocessing*
3. Ekstraksi Fitur
4. Identifikasi *Linear Discriminant Analysis*
5. Pelatihan (*training*)
6. Pengujian (*testing*)
7. Pengujian Hasil Identifikasi

1.6.2 Metode Pengembangan Sistem

Metodologi yang diterapkan dalam pengembangan sistem ini adalah dengan pengembangan metode *waterfall* (siklus air terjun). Metode *waterfall* merupakan metode pengembangan yang bersifat sistematis dan sekuensial, yang mana dalam model pengembangan *waterfall* ini sistem dikerjakan secara berurutan atau secara linear. Berikut merupakan langkah-langkah atau tahapan dalam metode pengembangan sistem *waterfall* (Pressman, 2015):



Gambar 1.1 Tahapan metode *Waterfall*

- a. *Communication*
- b. *Planning (Estimating, Scheduling, Tracking)*
- c. *Modeling (Analysis & Design)*
- d. *Construction (Code & Test)*
- e. *Deployment (Delivery, Support, Feedback)*

1.7 Sistematika Penelitian

Sistematika penulisan yang digunakan dalam penyusunan laporan penelitian ini adalah sebagai berikut :

BAB I PENDAHULUAN

Pada bagian ini membahas tentang latar belakang masalah, perumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian, dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Pada bagian ini memuat tentang dasar teori yang digunakan untuk analisis dan perancangan sistem serta implementasi pada penelitian ini. Selain itu juga sebagai bahan referensi dan pondasi untuk memperkuat argumentasi dalam penelitian ini. Teori-teori yang sesuai dengan penelitian ini antara lain padi, pengolahan citra, dan metode *linear discriminant analysis*.

BAB III METODOLOGI PENELITIAN DAN PENGEMBANGAN SISTEM

Pada bagian ini berisi langkah-langkah yang dilakukan dalam penelitian yang dimulai dari perancangan hingga implementasi sistem.

BAB IV HASIL DAN PEMBAHASAN

Pada bagian ini akan menyajikan hasil penelitian yang berisi hasil implementasi dari analisis perancangan yang telah dibuat pada bab sebelumnya. Selain itu berisi pengujian terhadap hasil penelitian beserta pembahasannya.

BAB V PENUTUP

Pada bagian ini berisi kesimpulan dari hasil penelitian dan saran yang diajukan oleh penulis untuk pengembangan penelitian selanjutnya.

BAB II TINJAUAN PUSTAKA

2.1 Padi

Padi merupakan salah satu tanaman yang sangat penting dalam kehidupan manusia. Tanaman padi dapat ditemukan dengan mudah terutama di daerah pedesaan. Tanaman padi ini termasuk ke dalam genus *Oryza* L. yang memiliki kurang lebih 25 spesies yang tersebar di daerah subtropis maupun tropis. Padi yang saat ini dikonsumsi oleh banyak orang adalah hasil persilangan antara *Oryza Officinalis* dan *Oryza sativa* F. *Spontane* (Mustikasucy, 2020).

Padi termasuk kedalam *Famili Poaceae*. Tingkat populasi famili ini sangat karena banyak dibudidayakan dan digunakan sebagai sumber pangan. Di dalam *Famili* ini terdapat tujuh subfamili, yaitu subfamili *Arundinoideae*, *Bambusoideae*, *Centothecoideae*, *Chloridoideae*, *Panicoideae*, *Pooideae* dan *Stipoideae* (Dinas Ketahanan Pangan, 2021).

Padi merupakan tanaman semusim yang umumnya hidupnya hanya satu musim saja atau kurang lebih tiga bulan sebelum dipanen. Padi memiliki akar serabut, batang pendek, bentuk daun lanset, bunga majemuk, tipe malai bercabang, struktur menyerupai batang dan ditutupi oleh rambut pendek dan panjang.

Tabel 2.1 Kelas *Oryza sativa*

Regnum	Plantae
Divisio	Spermatophyta
Sub divisio	Angiospermae
Classis	Monocotyledoneae
Ordo	Poales
Familia	Graminae
Genus	<i>Oryza</i>
Species	<i>Oryza sativa</i> L.

2.1.1 Masa Pertumbuhan Padi

Pertumbuhan padi dibagi dua masa yaitu masa vegetatif dan generatif. Dari kedua masa tersebut, masa generatif tersebut kemudian diperluas menjadi fase reproduktif dan pematangan. Masa reproduktif yang merupakan masa yang berlangsung kurang lebih selama 35 hari di daerah tropis sedangkan untuk fase pematangan berlangsung kurang lebih selama 30 hari. Masa vegetatif dari masing-masing varietas padi menentukan perbedaan umur padi. Sebagai contoh varietas padi IR 64 dikatakan matang dan telah siap untuk dipanen dalam 120 hari memiliki yang masa vegetatif selama 55 hari (Timur, 2020).

Masa vegetatif merupakan permulaan dari pertumbuhan tanaman. Pertumbuhan padi diawali dengan perkecambahan benih sampai terbentuknya malai padi. Pada tahap perkecambahan, benih melakukan penyerapan air dari lingkungan (disebabkan oleh perbedaan kadar air di antara benih dan lingkungan). Masa dormansi (masa dimana benih mengalami masa tidur sehingga tidak akan mengalami pertumbuhan meskipun ditanam pada kondisi optimum) akan pecah dengan ditandai munculnya radicle dan plumule. Perkecambahan benih sangat dipengaruhi oleh kelembaban, cahaya, dan suhu. Petani akan merendam benih selama kurang lebih 24 jam dan pemeraman selama 24 jam. Tahap perkecambahan benih akan berakhir sampai daun pertama tumbuh yang berlangsung kurang lebih selama 3-5 hari.

Tahap pertunasan diawali pada saat benih berkecambah sampai anak daun pertama tumbuh. Pada umumnya petani seringkali melewatkan tahap pertumbuhan ini di masa persemaian. Di awal masa persebaran mulai tumbuh akar seminal sampai akar adventif yang akan membentuk sistem akar serabut yang permanen. Radikula dan akar seminal sementara akan digantikan oleh akar serabut. Seiring dengan terus bertumbuhnya tunas, dua daun turut berkembang. Fase ini ditandai dengan daun yang terus berkembang 3-4 hari/daun pada tahap awal pertumbuhan hingga terbentuknya daun sempurna. Dengan demikian setelah melewati 15-20 hari setelah penyebaran bibit telah memiliki 5 daun dan sistem perakaran yang memiliki perkembangan sangat pesat. Dalam keadaan seperti ini bibit telah siap untuk dipindah tanamkan.

Setelah tumbuhnya daun kelima, tanaman akan mulai membentuk anakan daun seiring dengan tumbuhnya tunas yang baru. Anakan daun akan tumbuh dari tunas axillary yang terdapat buku batang yang kemudian menggantikan tempat daun hingga berkembang. Bibit akan memperlihatkan posisi dua anakan pertama yang mengapit batang utama dan daunnya. Setelah tumbuh, anakan pertama akan memperlihatkan anakan sekunder, pertumbuhan ini akan berlanjut sampai anakan maksimal. Di fase ini terdapat dua tahap yang sangat penting yaitu pembentukan anakan aktif yang selanjutnya akan disusul dengan perpanjangan batang (*stem elongation*). Dua tahap ini dapat bertumbuh dengan tumpang tindih, tanaman yang telah berhenti mengalami pertumbuhan anakan akan mengalami perpanjangan batang, buku kelima dari batang bagian bawah malai, dan bertambah panjang kurang lebih 2-4 cm sebelum pembentukan malai. Disamping itu, tanaman yang muda dan masih dapat membentuk anakan duan baru, sehingga akan sehingga akan memperlihatkan perkembangan cabang yang sangat pesat. Pada umumnya fase pembentukan anakan berlangsung selama kurang lebih 30 hari. Pada tanaman yang menggunakan sistem tanam benih langsung periode ini terjadi kurang dari 30 hari karena bibit tidak mengalami fase stagnasi seperti tanaman sistem tanam pindah yang beradaptasi terlebih dahulu dengan lingkungan baru setelah dipindah tanamkan. Pembentukan anakan yang lama dan tidak produktif ini biasanya disebabkan oleh penggunaan pupuk yang berlebihan.

Menurut Surmaini & Syahbuddin (2016) curah hujan dan status ketersediaan air tanah menjadi dasar konsep penentuan awal musim tanam. Terdapat 3 status dalam ketersediaan air untuk menentukan awal musim tanam, antara lain adalah sebagai berikut:

- a. Ketersediaan air yang mencukupi tanaman apabila curah hujan lebih besar dari 50 mm/dasarian
- b. Ketersediaan air dikatakan hampir mencukupi apabila curah hujan 25–50 mm/dasarian.
- c. Ketersediaan dikatakan tidak mencukupi apabila curah hujan kurang dari 25 mm/dasarian

2.1.2 Varietas Padi

Tanaman padi diklasifikasikan berdasarkan varietasnya. Secara umum tanaman padi diklasifikasikan kedalam tiga varietas, yaitu (Dinas Ketahanan Pangan, 2021):

- a. Varietas Padi Hibrida

Varietas padi ini dikenal sebagai varietas padi sekali tanam. Hasil panen padi varietas ini akan maksimal apabila hanya dengan sekali tanam, namun apabila

benih tersebut ditanam kembali maka panen yang dihasilkan akan mengurangi pengurangan baik itu dari segi kuantitas maupun kualitas. Beberapa contoh dari varietas padi hibrida yaitu: Intani 1 dan 2, PP1, H1, Bernas Prima, dan lain-lain.

b. **Varietas Padi Unggul**

Varietas padi ini merupakan kebalikan dari varietas padi hibrida. Apabila varietas padi hibrida hanya untuk sekali tanam, varietas padi unggul ini mampu berkali-kali tanam apabila ditanam menggunakan teknik yang baik. Hasil panen dari padi varietas ini dapat digunakan sebagai benih untuk ditanam kembali. Varietas padi unggul ini merupakan varietas yang telah dilepas oleh pemerintah dengan SK Menteri Pertanian dan telah melewati berbagai uji coba. Harga benih padi unggul ini juga tergolong murah sekitar Rp 5000 – Rp 10.000. Contoh dari padi varietas unggul, yaitu: Padi Ciherang, IR-64, Cimelati, Cibogo, Padi Way Apo Buru, dan lain-lain.

c. **Varietas Padi Lokal**

Varietas padi ini merupakan varietas padi yang telah beradaptasi pada suatu daerah, sehingga varietas padi jenis ini memiliki karakteristik yang spesifik dari lokasi di tempat padi tersebut beradaptasi. Contoh padi varietas lokal, yaitu: varietas kebo, Dharma Ayu, Pemuda Idaman, Merong (Pasuruan), Simenep, Srimulih, Andel Jaran, Ketan Lusi, Ekor Kuda, Gropak (Kulon Progo-Jogja) dan lain-lain.

2.1.3 Jenis Budidaya Padi

Secara umum padi dibagi beberapa jenis menurut budidayanya, diantaranya adalah sebagai berikut (Dinas Ketahanan Pangan, 2021).

a. **Padi gogo**

Padi gogo merupakan salah satu jenis padi yang ditanam di area kering dan tidak membutuhkan air yang banyak. Padi gogo ini juga sering disebut dengan padi tegalan. Dalam jenis budidaya padi gogo ini biasanya juga diterapkan sistem tumpang sari yaitu tanaman yang ditanam bukan hanya satu jenis namun ditanam dengan jenis tanaman produksi yang lainnya.

b. **Padi rawa**

Padi rawa atau yang sering disebut dengan padi pasang surut. Padi varietas ini biasanya dibudidayakan di daerah rawa-rawa karena padi rawa ini mampu membentuk batang yang panjang sehingga dapat mengikuti perubahan keadaan air.

2.2 Citra

Citra merupakan sebuah gambaran, kemiripan, atau imitasi dari suatu objek. Suatu citra dapat berbentuk foto, sinyal video atau bersifat digital yang dapat langsung disimpan pada suatu media penyimpanan (Sutojo, 2017). Menurut Jumadi, Yupianti, & Sartika (2021), citra merupakan representasi dua dimensi dari visual dan menyangkut seni, teknik, dan sebagainya. Suatu citra merupakan kumpulan dari piksel-piksel atau titik-titik yang berwarna dan memiliki berbentuk dua dimensi. Ditinjau dari presisi yang digunakan untuk menentukan koordinat dalam dua larik dimensi dari suatu citra yang digunakan untuk menentukan tingkat keabuan dari elemen atau warna citra, maka citra dapat dibagi menjadi empat kelas citra diantaranya adalah kontinu-kontinu, diskrit-kontinu, kontinu-diskrit, dan diskrit-diskrit yang mana dalam label pertama menunjukkan presisi dari titik-titik

koordinat pada bidang citra sedangkan label kedua menunjukkan posisi skalabilitas dari warna citra. Diskrit dinyatakan dengan presisi angka yang memiliki batasan, sedangkan kontinu dinyatakan dengan presisi angka yang tidak terhingga. Citra juga mempunyai karakteristik yang tidak dimiliki oleh data teks, yaitu kekayaan akan informasi. Sebagai hasil keluaran, sebuah citra dapat berupa:

- a. Citra diam (*still image*)
Pengertian dari citra diam adalah citra tunggal yang tidak bergerak, atau biasa disebut dengan citra.
- b. Citra bergerak (*moving image*)
Pengertian dari citra bergerak adalah rentetan citra diam yang ditampilkan secara beruntun (*sequential*) sehingga memberi kesan pada mata sebagai gambar yang bergerak. Setiap citra di dalam rangkaian tersebut disebut *frame*.

2.3 Citra digital

Citra digital adalah suatu gambar dua dimensi yang terbentuk dari representasi dari suatu fungsi intensitas cahaya dalam bentuk diskrit di bidang dua dimensi melalui berbagai jenis proses digitalisasi menggunakan alat optik yang dilengkapi dengan fungsi digitalisasi citra yang kemudian mampu menghasilkan suatu citra diskrit. Citra digital tersusun oleh sekumpulan piksel terdiri dari koordinat (x, y) dan amplitudo (x, y) . Koordinat ini menunjukkan letak piksel dalam suatu citra, sedangkan amplitudo menunjukkan nilai intensitas warna dari suatu citra. Citra merupakan fungsi kontinu dari intensitas cahaya yang terdapat dalam bidang dua dimensi secara matematis (Gonzalez & Woods, 2018).

Secara harfiah citra digital merupakan gambar dua dimensi yang dapat diolah komputer sebagai suatu notasi diskrit nilai digital yang berupa piksel (*picture elements*). Untuk mendapatkan notasi tersebut diperlukan suatu proses digitalisasi. Di dalam proses ini akan menghasilkan citra digital. Secara matematis, citra adalah sebuah fungsi kontinu dari intensitas cahaya pada bidang dua dimensi. Dimensitas citra inilah yang kemudian dinyatakan dengan panjang x lebar.

Menurut (Munantri, Sofyan, & Florestiyanto, 2020), berdasarkan dimensitasnya suatu citra digital dapat diwakilkan oleh sebuah matriks dua dimensi $f(x, y)$ yang mencakup M kolom dan N baris, dimana pada perpotongan kolom dan baris tersebut yang selanjutnya akan menghasilkan suatu bagian terkecil dari sebuah citra yang disebut piksel. Persamaan matriks dalam citra digital dinyatakan sebagai persamaan sebagai berikut:

$$f(x, y) \approx \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0, M - 1) \\ f(1,0) & f(1,1) & \dots & f(1, M - 1) \\ \vdots & \vdots & \vdots & \vdots \\ f(N - 1,0) & f(N - 1,1) & \dots & f(N - 1, M - 1) \end{bmatrix} \dots\dots\dots(2.1)$$

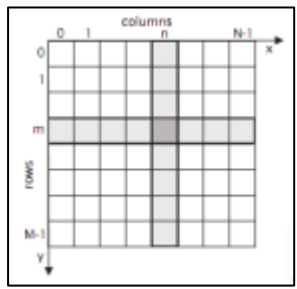
Suatu citra $f(x, y)$ secara matematis dapat dituliskan sebagai berikut:
 $0 \leq x \leq M - 1; 0 \leq y \leq N - 1; 0 \leq f(x, y) \leq G - 1 \dots\dots\dots(2.2)$

- Keterangan :
- M = jumlah piksel baris yang terdapat pada array citra
 - N = jumlah piksel kolom yang terdapat pada array citra
 - G = nilai skala keabuan

Besarnya nilai dari M , N dan G pada umumnya merupakan nilai perpangkatan dari dua.

$$M = 2^m; N = 2^n; G = 2^k \dots\dots\dots(2.3)$$

Nilai M , N dan G tersebut bernilai positif. Dimana M merupakan jumlah piksel baris (*row*) yang terdapat pada array citra, sedangkan N merupakan jumlah piksel kolom (*coloumn*) yang terdapat pada array citra, dan G merupakan nilai dari skala keabuan (*gray level*) atau tingkat keterangan dalam citra. Nilai x dan y dalam fungsi $f(x, y)$ merupakan suatu koordinat spasial dan amplitudo f yang terdapat pada titik koordinat (x, y) yang dikenal dengan intensitas atau tingkat keabuan dari citra pada suatu titik. Bila koordinat spasial dan amplitude memiliki nilai yang terbatas (*finite*) dan memiliki nilai diskrit maka dapat dikatakan bahwa citra tersebut merupakan suatu citra digital.

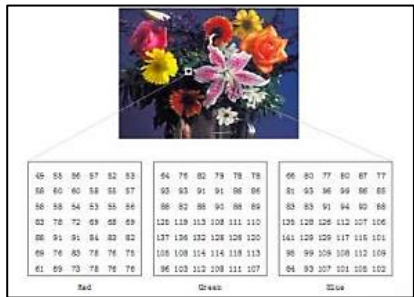


Gambar 2.1 Representasi citra digital

Menurut Pulung dan Sutejo (2017) berdasarkan kombinasi warna piksel, citra dapat dibagi menjadi empat jenis, yaitu:

a. Citra berwarna (*True Color Image*)

Citra berwarna (*True Color Image*) adalah suatu citra yang terbentuk dari tiga kanal warna yaitu kanal merah, kanal hijau, dan kanal biru. Citra berwarna disebut juga sebagai 24-bit *color image*, hal ini dikarenakan citra berwarna membutuhkan penyimpanan sebesar 24-bit dalam tiap pikselnya. Tiap-tiap kanal tersebut mempunyai intensitas piksel dengan kedalaman bit sebesar 8-bit atau mempunyai variasi warna sebanyak $255^3 = 16.581.375$ (16K) derajat warna (0-255). Tiap-tiap piksel yang terdapat pada citra berwarna mempunyai intensitas warna yang merupakan kombinasi dari tiga nilai intensitas yang terdapat dalam kanal R, G, B.



Gambar 2.2 Citra berwarna

b. Citra biner (Monokrom)

Citra biner (Monokrom) adalah suatu citra yang mempunyai kedalaman sebesar 1 bit pada pikselnya. Citra biner ini hanya mempunyai dua warna saja yaitu hitam dan putih dan hanya membutuhkan 1 bit memori untuk setiap piksel agar dapat

melakukan penyimpanan kedua warna tersebut. Berikut merupakan fungsi dari binary image.

$$I_{Bin}(x, y) = \begin{cases} 0 & I_{BW}(x, y) < T \\ 255 & I_{BW}(x, y) \geq T \end{cases} \dots\dots\dots(2.4)$$

Berikut merupakan fungsi binary image dalam bentuk floating point.

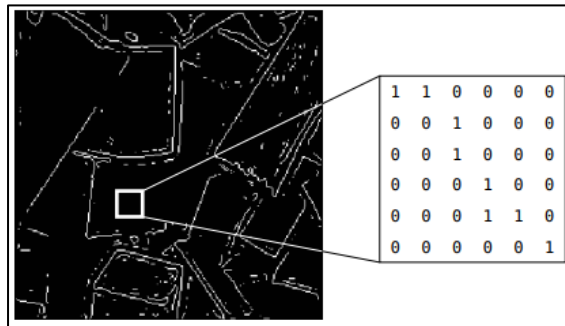
$$I_{Bin}(x, y) = \begin{cases} 0 & I_{BW}(x, y) < T \\ 1 & I_{BW}(x, y) \geq T \end{cases} \dots\dots\dots(2.5)$$

Keterangan :

$I_{BW}(x, y)$ = nilai piksel *gray* di titik (x, y)

$I_{Bin}(x, y)$ = nilai piksel biner pada titik (x, y)

T = nilai *threshold*



Gambar 2.3 Binary image

c. Citra *grayscale* (skala keabuan)

Citra grayscale (skala keabuan) adalah suatu citra yang nilai intensitas pikselnya diukur berdasarkan derajat nilai keabuan. Pada citra grayscale 8-bit, derajat warna hitam sampai dengan putih dibagi menjadi 256 derajat keabuan warna, dimana warna putih sempurna dilambangkan dengan nilai 255 dan citra hitam sempurna dilambangkan dengan nilai 0.

$$I_{BW}(x, y) = \frac{I_R(x,y)+I_G(x,y)+I_B(x,y)}{3} \dots\dots\dots(2.6)$$

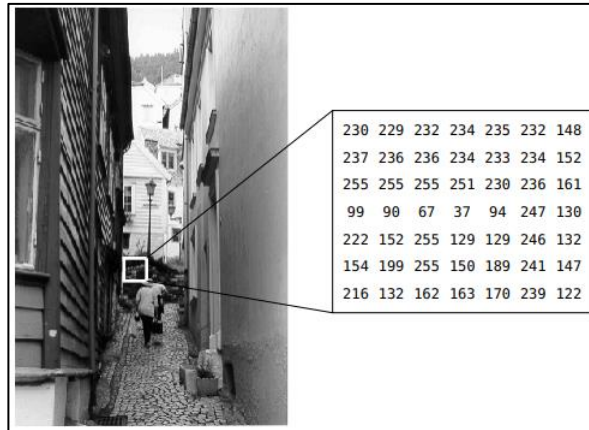
Keterangan :

$I_{BW}(x, y)$ = nilai piksel *black and white* pada titik (x, y)

$I_R(x, y)$ = nilai piksel *red* pada titik (x, y)

$I_G(x, y)$ = nilai piksel *green* pada titik (x, y)

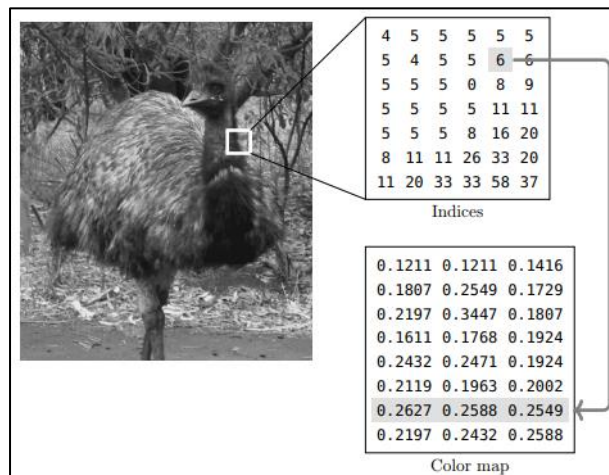
$I_B(x, y)$ = nilai piksel *blue* pada titik (x, y)



Gambar 2.4 Grayscale image

d. Citra Berindeks

Citra berindeks merupakan citra yang berbeda cara penyimpanannya dari citra yang sebelum-sebelumnya. Citra berindeks terdiri dari sebuah matrik citra dan marik *color map*. Matrik citra dapat menunjukkan alamat tempat warna citra tersebut disimpan,.



Gambar 2.5 Citra berindeks.

2.4 Pengolahan Citra

Meskipun sebuah citra kaya akan informasi, akan tetapi seringkali citra yang dimiliki mengalami penurunan kualitas atau yang sering disebut dengan degradasi. Contoh penurunan kualitas tersebut seperti cacat atau adanya *noise*, warna yang terlalu kontras, atau brightness yang berlebihan. Citra dengan kondisi seperti ini akan lebih sulit untuk diinterpretasikan karena informasi yang disampaikan oleh citra tersebut akan berkurang. Oleh karena itu, diperlukan suatu kegiatan yang dilakukan untuk memperbaiki kualitas dari sebuah citra agar mudah untuk diinterpretasikan baik oleh manusia atau komputer (Munir, 2019).

Pengolahan citra merupakan kegiatan yang berfungsi untuk memperbaiki kualitas dari sebuah citra agar mudah diinterpretasikan oleh manusia maupun komputer. Beberapa teknik dalam pengolahan citra mentransformasikan citra menjadi citra yang lain, namun dengan citra keluaran yang lebih baik dari citra masukan (Munir, 2019)

2.4.1 Akuisisi Citra

Akuisisi citra merupakan tahapan awal yang bertujuan untuk mendapatkan citra sebagai data yang akan digunakan dalam penelitian. Akuisisi citra merupakan langkah awal untuk memperoleh citra yang bertujuan untuk menentukan data yang dibutuhkan. Citra inputan pada proses akuisisi diperoleh dari gambar, video, scanner maupun hardware lainnya sebagai penunjang dalam pengambilan citra. Dalam proses akuisisi citra seringkali dibutuhkan proses digitalisasi karena belum tentu data yang diperoleh merupakan data digital (Sutojo, 2017).

2.4.2 Image Augmentation

Image augmentation secara artifisial membuat training images melalui beberapa cara pemrosesan atau kombinasi pemrosesan, seperti rotasi acak, pergeseran, geser, flip, transformasi geometris, transformasi ruang warna, kombinasi warna, penghapusan acak, neural style transfer. Dibawah ini merupakan pertimbangan penting yang berhubungan dengan geometric augmentations (Mikołajczyk & Grochowski, 2019).

a. Membalik (*Flipping*)

Membalik dengan sumbu horizontal dibandingkan membalik dengan sumbu vertikal. Augmentasi ini adalah salah satu yang paling mudah untuk diterapkan dan terbukti berguna pada kumpulan data seperti CIFAR-10 dan ImageNet.

b. Memotong (*Cropping*)

Memotong gambar dapat dilakukan sebagai salah satu langkah pemrosesan untuk data gambar dengan menggabungkan dimensi panjang dan lebar dengan cara memotong *central patch* dari setiap gambar. Memotong secara acak juga dapat digunakan memberikan efek yang sangat mirip dengan translasi. Terdapat perbedaan antara memotong dan translasi yaitu apabila memotong akan mengurangi ukuran input seperti $(256, 256) \rightarrow (224, 224)$, sedangkan translasi mempertahankan dimensi spasial dari gambar.

c. Rotasi

Augmentasi rotasi dilakukan dengan cara memutar gambar ke kanan atau kiri pada sumbu antara 1° dan 359° . Keamanan dari augmentasi rotasi sangat ditentukan oleh parameter derajat rotasi. Rotasi kecil seperti antara 1 dan 20 atau 1 hingga 20 akan sangat mempengaruhi hasil. Seiring dengan meningkatnya derajat rotasi, label data tidak lagi dipertahankan setelah transformasi.

d. Translasi

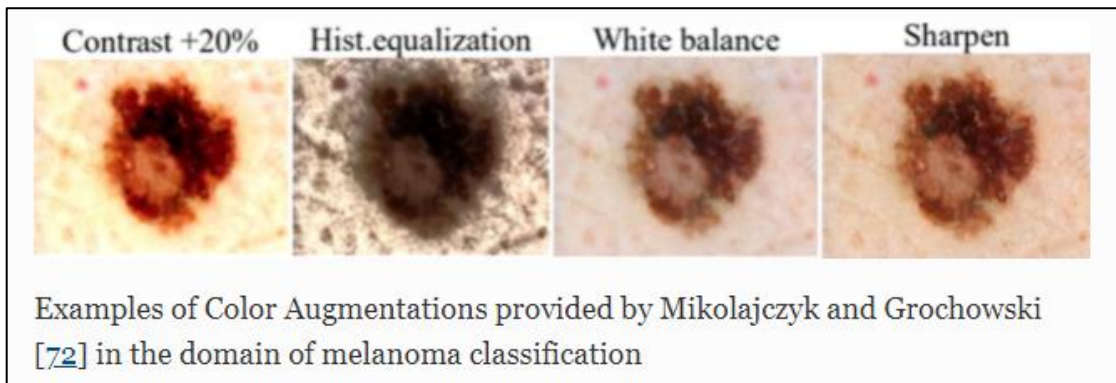
Augmentasi translasi dilakukan dengan cara menggeser gambar ke kiri, kanan, atas, atau bawah sangat berguna pada transformasi untuk menghindari posisi bias dalam data.

e. Transformasi Ruang Warna

Data gambar dikodekan kedalam 3 *stacked matrices*, masing-masing berukuran tinggi x lebar. Matriks ini mewakili nilai piksel untuk masing-masing nilai warna RGB. Bias pencahayaan merupakan salah satu masalah yang paling sering ditemui dalam pengenalan gambar, Oleh karena itu, efektivitas transformasi ruang warna juga dikenal sebagai transformasi fotometrik.

Transformasi ruang warna juga dapat dilakukan pada aplikasi pengolahan gambar. Nilai piksel dari gambar di setiap saluran warna RGB digabungkan untuk membentuk histogram warna. Histogram ini dapat dimanipulasi untuk menerapkan filter yang mampu mengubah karakteristik ruang warna dari sebuah gambar.

Terdapat banyak kebebasan dalam berkreasi menggunakan augmentasi ruang warna. Mengubah distribusi warna dari gambar juga dapat menjadi solusi yang baik untuk masalah pencahayaan yang dihadapi pada saat melakukan *training* data.



Gambar 2.6 Contoh *augmentation*.

2.4.3 Resize

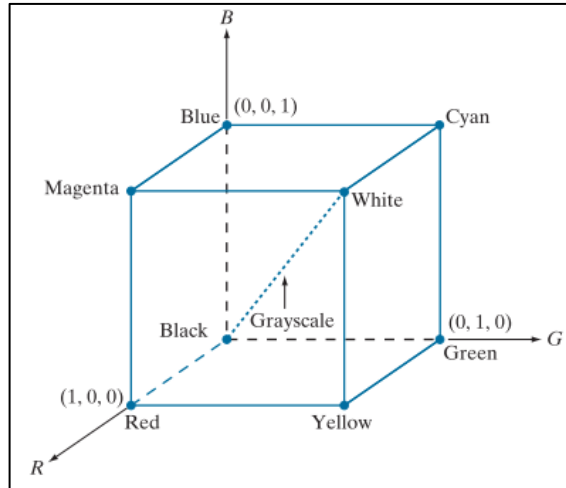
Mengubah ukuran gambar artinya mengubah dimensinya, baik itu lebar saja, tinggi saja atau mengubah keduanya. Selain itu, rasio aspek gambar asli dapat dipertahankan dalam gambar yang diubah ukurannya. (LearnOpenCV, t.thn.)

2.4.4 Pengolahan Warna

Suatu hasil kombinasi cahaya yang memiliki panjang gelombang yang berbeda dan memberikan rentang warna paling tinggi yaitu *Red* (Merah), *Green* (Hijau), dan *Blue* (Biru). Warna-warna tersebut disebut dengan warna primer atau RGB. Warna-warna lain yang terbentuk didapatkan dari campuran-campuran warna primer. Macam-macam warna dapat dihasilkan dari campuran warna primer C1, C2, dan C3 dengan rumus sebagai berikut (Ratnasari, 2018).

$$C = C1 + C2 + C3 \dots\dots\dots(2.7)$$

Dalam pengolahan warna citra dapat berupa perbaikan citra melalui peningkatan kualitas citra, restorasi citra dengan cara memperbaiki model citra maupun atribut yang lain. Pengolahan warna merupakan suatu teknik dalam pengolahan citra digital dengan model warna yang biasa digunakan adalah model warna RGB (red, green, blue). Contoh citra yang menggunakan model warna RGB adalah citra yang dihasilkan dari kamera digital maupun display monitor. Pada model warna RGB, masing-masing warna dinyatakan sebagai suatu komponen dari spektrum dasar *red*, *green*, *blue*. Model ini menggunakan sistem koordinat Cartesian. Ilustrasi dari model warna RGB ditentukan pada Gambar 2.7 dibawah ini.



Gambar 2.7 Model Warna RGB (Gonzalez & Woods, 2018)

Nilai warna yang terdapat pada Gambar 2.7 diasumsikan apabila telah dinormalkan, sehingga nilai maksimal yang terdapat pada sumbu R, G, B memiliki nilai satu. Posisi dari warna primer dari model RGB ditunjukkan pada tiga sudut yang lain. Posisi dari warna hitam berada pada titik asal koordinat yaitu (0,0,0) dan warna putih berada pada sudut berseberangan yaitu (1,1,1) (Gonzalez & Woods, 2018).

Sebelum dilaksanakan proses segmentasi, terlebih dahulu dilakukan pengolahan ruang warna RGB menjadi HSV. Hal ini dilakukan sebab ruang warna HSV lebih murni dan belum bercampur seperti halnya ruang warna RGB. Hue merupakan representasi dari warna cahaya yang tampak (merah, jingga, kuning, hijau, biru, dan ungu). Sedangkan *saturation* merupakan tingkat kemurnian warna, semakin tinggi nilai saturasi warna maka akan semakin murni citra. Berikut adalah rumus perhitungan konversi ruang warna RGB ke HSV.

$$R' = \frac{R}{(R+G+B)} \dots\dots\dots(2.8)$$

$$G' = \frac{G}{(R+G+B)} \dots\dots\dots(2.9)$$

$$B' = \frac{B}{(R+G+B)} \dots\dots\dots(2.10)$$

$$Cmax = \max(R', G', B') \dots\dots\dots(2.11)$$

$$Cmin = \min(R', G', B') \dots\dots\dots(2.12)$$

$$\Delta = Cmax - Cmin \dots\dots\dots(2.13)$$

Perhitungan nilai *Hue*:

$$H = \begin{cases} 0 & \Delta = 0 \\ 60^\circ \times \left(\frac{G'-B'}{\Delta} \text{ mod } 6\right), C_{max} = R' \\ 60^\circ \times \left(\frac{B'-R'}{\Delta} + 2\right), C_{max} = G' \\ 60^\circ \times \left(\frac{R'-G'}{\Delta} + 4\right), C_{max} = B' \end{cases} \dots\dots\dots(2.14)$$

Perhitungan nilai *Saturation*:

$$S = \begin{cases} 0, C_{max} \neq 0 \\ \frac{\Delta}{C_{max}}, C_{max} \neq 0 \end{cases} \dots\dots\dots(2.15)$$

Perhitungan nilai *Value*:

$$V = Cmax \dots\dots\dots(2.16)$$

2.4.5 Perataan Histogram (Histogram Equalization)

Perataan histogram atau *histogram equalization* adalah proses merubah nilai-nilai intensitas citra sehingga penyebarannya seragam (*uniform*). Perataan histogram didapatkan dengan mengubah derajat keabuan suatu piksel (r) dengan derajat keabuan yang baru (s) dengan fungsi transformasi T , yang dalam hal ini $s = T(r)$. Yang artinya r dapat didapatkan kembali dari s dengan transformasi invers $r = T^{-1}(s)$ dimana $0 \leq s \leq 1$. Untuk $0 \leq r_k \leq 1$, maka $0 \leq T(r) \leq 1$. Ini dilakukan untuk menjamin pemetaan konsisten pada rentang nilai yang diperbolehkan.

Tujuan dari perataan histogram adalah untuk memperoleh penyebaran histogram yang merata, sehingga setiap derajat keabuan memiliki jumlah piksel yang relatif sama. Sebab, histogram menyatakan peluang piksel dengan derajat keabuan tertentu maka rumus untuk menghitung perataan histogram adalah sebagai berikut (Sutojo, 2017).

$$P_r(r_k) = \frac{n_k}{n} \text{ dalam hal ini } r_k = \frac{k}{L-1}, 0 \leq k \leq L-1 \dots\dots\dots(2.17)$$

Yang artinya, derajat (k) dinormalkan terhadap derajat keabuan terbesar ($L-1$). Nilai $r_k = 1$ menyatakan putih dalam skala keabuan yang didefinisikan.

2.4.6 Segmentasi Citra

Segmentasi citra merupakan suatu teknik yang bertujuan untuk membedakan antara objek dengan latar belakang (*background*). Segmentasi citra bergantung dengan permasalahan yang akan diselesaikan. Proses segmentasi harus dihentikan apabila obyek telah terisolasi atau terlihat dengan jelas. Tingkat akurasi dalam proses segmentasi citra yang dilakukan sangat bergantung kepada tingkat keberhasilan prosedur analisis yang dilakukan (Sutama, Magdalena, & Wijayanto, 2018).

Menurut Castleman (1996), segmentasi citra dibagi menjadi tiga pendekatan, diantaranya adalah sebagai berikut:

1. Pendekatan Wilayah (*Region Approach*) memiliki tujuan untuk membagi citra menjadi beberapa wilayah sehingga mendapatkan suatu daerah sesuai dengan kriteria yang telah ditentukan
2. Pendekatan tepi (*Edge Approach*) memiliki tujuan untuk melakukan identifikasi tepi piksel dan menggabungkannya menjadi sebuah batas yang telah ditentukan.
3. Pendekatan batas (*boundary approach*) memiliki tujuan untuk menentukan batas antar wilayah.

Segmentasi citra dapat dilakukan dengan menggunakan bermacam-macam metode, namun mempunyai tujuan yang sama yaitu memperoleh representasi dari suatu citra. Segmentasi dapat dilakukan dengan beberapa macam metode, akan tetapi cukup sulit untuk memilih metode yang komprehensif. Oleh sebab itu pemilihan metode dalam segmentasi citra bergantung pada pendekatan yang akan diterapkan serta fitur yang didapatkan dari suatu citra.

Segmentasi citra dengan menggunakan ruang warna HSV (*Hue, Saturation, Value*) merupakan proses segmentasi dengan pendekatan wilayah (*region approach*) yang bekerja dengan melakukan analisis nilai warna pada setiap piksel citra dan membagi citra sesuai dengan fitur yang diinginkan (Pereira et al., 2018).

Dalam penelitian ini segmentasi yang digunakan merupakan segmentasi warna dengan menggunakan ruang warna HSV. Ruang warna HSV banyak digunakan pada

penelitian sains karena karakteristiknya yang tidak berubah, tidak seperti ruang warna RGB (S., N., & G., 2017). *Hue* merupakan representasi dari warna, *Saturation* merupakan representasi dari warna yang mendominasi, dan *Value* merupakan representasi dari kecerahan. Oleh karena itu, HSV dapat mencari berdasarkan posisi warna dan kemurnian warna dibandingkan mencari nilai RGB tertentu (Pereira et al., 2018).

Ruang warna RGB digunakan sebagai standar acuan warna pada citra digital. Oleh karena itu, perlu dilakukan konversi dari model warna RGB ke model. Namun untuk mendeteksi warna yang diinginkan perlu untuk menentukan nilai toleransi pada setiap dimensi HSV. Nilai toleransi tersebut yang selanjutnya digunakan sebagai ambang batas dalam proses threshold dengan warna yang diinginkan. Berikut merupakan langkah-langkah dalam melakukan proses segmentasi menggunakan ruang warna HSV (Giannakopoulos, 2008):

1. Menentukan nilai HSV yang akan dijadikan sebagai acuan (T) serta nilai HSV yang dijadikan sebagai nilai toleransi (Tol).
2. Melakukan transpose citra RGB menjadi citra HSV
3. Menghitung nilai tengah (x) dari nilai HSV yang dijadikan sebagai acuan (T) serta nilai toleransi (tol).
4. Apabila nilai tengah (x) tidak berada pada $T - Tol < x < T + Tol$ maka akan berwarna hitam
5. Melakukan transpose kembali citra menjadi RGB, dan tampilkan hasil segmentasi.

2.5 Ekstraksi Fitur

Ekstraksi fitur adalah sebuah proses untuk mengambil atau mengekstrak karakteristik atau informasi dari suatu citra untuk digunakan dalam klasifikasi yang berbentuk susunan-susunan struktur dari suatu gambar (Sutojo, 2017).

2.6 Confusion Matrix

Confusion matrix adalah ukuran yang sangat populer digunakan saat memecahkan masalah klasifikasi. Hal ini dapat diterapkan untuk klasifikasi biner serta untuk masalah klasifikasi multiclass. Berikut merupakan rumus yang digunakan untuk menghitung confusion matrix

Berikut adalah rumus yang digunakan untuk menghitung nilai *accuracy*.

$$accuracy = \frac{TP+TN}{TP+FP+FN+TN} * 100 \dots\dots\dots(2.18)$$

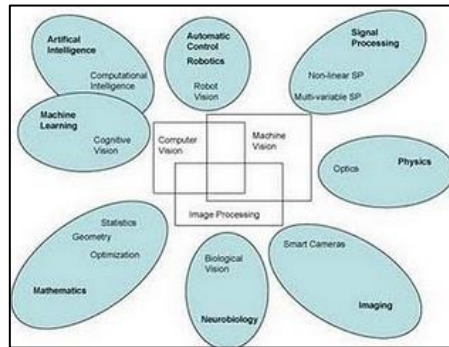
$$precision = \frac{TP}{TP+FP} * 100 \dots\dots\dots(2.19)$$

$$recall = \frac{TP}{TP+FN} * 100 \dots\dots\dots(2.20)$$

2.7 Computer Vision

Menurut Erick Wengrowski (2012), *computer vision* merupakan suatu cara dimana komputer dapat melihat layaknya manusia, mampu melakukan pengolahan informasi dan menjalankan tugas-tugas tertentu. Tujuan umum pembuatan *computer vision* ini merupakan pembuatan suatu model nyata yang bersumber dari sebuah citra. *Computer vision* pada umumnya menggunakan kamera sebagai media penyedia data. Teknik *computer vision* berfokus pada bagaimana komputer dapat mendeteksi objek yang diamati, berbeda dengan komputer grafik yang lebih berfokus pada bagaimana komputer mampu

melakukan manipulasi terhadap gambar. Teknik yang digunakan dalam *computer vision* merupakan kombinasi antara pengolahan citra dan pengenalan pola



Gambar 2.8 Hubungan antara computer vision, pengolahan citra dan pengenalan pola.

Computer vision memiliki beberapa macam pemrosesan yang dimulai dengan tahap *preprocessing* seperti *noise removal*, *image sharpening* dan segmentasi ruang warna citra. Tahapan selanjutnya adalah tahapan ekstraksi fitur seperti deteksi tepi dan deteksi kontur, selanjutnya adalah tahap identifikasi fitur seperti *pattern matching*, dilanjutkan dengan tahapan kompleks untuk melakukan rekonstruksi berbasis model citra. Berikut merupakan proses yang ada di dalam *computer vision* (Mutiara, 2005):

1. Memperoleh citra digital
2. Melakukan pengolahan citra digital
3. Melakukan analisis dan interpretasi hasil citra dengan tujuan tertentu.

Sehingga proses yang ada di dalam *computer vision* dapat diklasifikasikan seperti pada Gambar 2.9 (Mutiara, 2005).

Hirarki Pemrosesan	Contoh Algoritma
<i>preprocessing</i>	<i>noise removal</i> <i>contrast enhancement</i>
<i>lowest-level feature extraction</i>	<i>edge detection</i> <i>texture detection</i>
<i>intermediate-level feature identification</i>	<i>connectivity</i> <i>pattern matching</i> <i>boundary coding</i>
<i>high-level scene interpretation via images</i>	<i>model-base recognition</i>

Gambar 2.9 Hierarki dalam computer vision

2.8 Linear Discriminant Analysis

Salah satu metode yang terkenal dalam ekstraksi fitur adalah *Principal Component Analysis* atau yang sering disebut dengan PCA. Dalam metode ekstraksi fitur PCA memiliki kelemahan yaitu pemisahan kelas yang kurang optimal, sehingga *Linear Discriminant Analysis* merupakan metode yang mengatasi kelemahan dari metode PCA (Cahyani, Wiryasaputra, & Gustriansyah, 2018).

Linear Discriminant Analysis (LDA) adalah salah satu metode yang menggunakan teori statistik yang sudah banyak digunakan dalam berbagai hal seperti *machine learning*, pengolahan data, maupun pengolahan citra yang dipakai untuk mereduksi dimensi. LDA berfokus pada pengklasifikasian terhadap data dan mencari kombinasi atribut yang terbaik

agar dapat digunakan untuk memisahkan kelas-kelas yang terdapat didalam dataset dengan baik. Dalam penggunaan LDA, data set lokasi tetap, namun kelas yang terbentuk menjadi lebih terpisah sehingga, menyebabkan jarak antar kelas menjadi lebih besar, sedangkan jarak antar data latih dalam satu kelas menjadi lebih kecil. Berikut merupakan tahapan metode Linear Discriminant Analysis:

1. Mengubah data latih dan data uji menjadi vektor $m_i = \begin{bmatrix} x_i \\ x_i \end{bmatrix}$. Dimana $i = 0, 1$ merupakan kelas matang dan mentah.

2. Membuat kelas matang dan mentah pada data latih dan data uji.

3. Menghitung rata-rata dalam kelas (m_i) dan rata-rata keseluruhan kelas (m).

4. Menghitung matriks sebaran antar kelas (S_b) menggunakan persamaan $\sum_{i=1}^k n_i (m_i - m) (m_i - m)^T \dots\dots\dots(2.21)$

5. Menghitung matriks sebaran dalam kelas (S_w) menggunakan persamaan $\sum_{j=1}^k (x_j - m_i) (x_j - m_i)^T \dots\dots\dots(2.22)$

6. Mencari matriks W dengan menggunakan persamaan $W = S_w^{-1} S_b \dots\dots\dots(2.23)$

7. Mencari vektor eigen dari W dan mengurutkan nilai eigen (λ)sesuai dengan urutan nilai yang ada pada nilai eigen dari besar ke kecil.

8. Melakukan reduksi dimensi dengan melakukan proyeksi vektor eigen terpilih ke ruang dimensi yang lebih rendah (*fisher basis vector*) menggunakan persamaan $\bar{u}_i = V^T x_i \dots\dots\dots(2.24)$

9. Dimana \bar{u}_i adalah vektor hasil proyeksi yang berdimensi lebih rendah, v adalah matriks yang berisi vektor-vektor eigen dari matriks W, dan x_i adalah vektor yang mewakili citra padi.

2.9 Tinjauan Literatur

Adapun beberapa penelitian yang telah dilakukan sebelumnya dan dijadikan referensi dalam penelitian ini dapat dilihat pada tabel.

No	Penulis	Judul	Keterangan
1.	Ghazali, Eko K. Subha, Galuh M., M. Burhannudin, M. Izzun Niam 2018	Aplikasi Kematangan Tomat Berdasarkan Warna dengan Metode <i>Linear Discriminant Analysis</i> (LDA)	Di dalam penelitian ini menggunakan ruang warna RGB dan metode Linear Discriminant Analysis untuk melakukan klasifikasi kematangan tomat dengan dua kelas yaitu, mentah dan matang. Diperoleh nilai akurasi sebesar 93,10% untuk kelas matang dan 100% untuk kelas mentah.
2.	Styandi, Anggi Diatma Syauqy, Dahnia Kurniawan, Wijaya, 2019	Klasifikasi Umur Padi berdasarkan Data Sensor Warna dengan menggunakan Metode K-NN	Di dalam penelitian ini perolehan data menggunakan sensor warna dengan ruang warna RGB yang dibagi menjadi dua kelas, yaitu mentah dan matang. Dari hasil penelitian ini diperoleh nilai akurasi sebesar 80% dengan k=5 dan k=9 sebesar 10%. Perbedaan nilai

No	Penulis	Judul	Keterangan
			RGB yang tidak terlalu signifikan mengakibatkan kesulitan dalam klasifikasi,
3	Basavaraj S. Anami, Naveen N. M., N. G. Hanamaratti, 2017	Bulk Paddy Grain Ageing Period Classification using RGB and HSI Color Features	Dalam penelitian ini terdiri dari dua tahap, yaitu tahap identifikasi varietas dan identifikasi kematangan padi yang mengacu pada warna bulir padi. Untuk mengekstraksi fitur dilakukan dengan mengkombinasikan ruang warna RGB dan HSI. Selanjutnya, tahap pengklasifikasian dilakukan dengan Multilayer Backpropagation Neural Network (BPNN) dengan akurasi 71,55%. Untuk optimasi dilakukan reduksi fitur yang kemudian menghasilkan akurasi 93,07%
4.	Imam Adi Sulisty, 2017	Sistem Deteksi Panen Padi Berdasar Warna Daun Menggunakan Fuzzy C-Means Clustering	Penelitian ini menggunakan data latih berjumlah 66 foto yang diperoleh bahwa rata-rata nilai hue data belum panen (hijau) yaitu 0.23 – 0.34, dan rata-rata nilai hue data siap panen (kuning) yaitu 0.10 – 0.21. Nilai rata-rata hue yang didapat dari data latih digunakan sebagai nilai acuan atau standar untuk pengujian. Dari penelitian tersebut didapatkan nilai akurasi sebesar 94,11%. Di dalam penelitian ini juga disarankan untuk menambahkan butir padi sebagai acuan bukan hanya daun padi saja.
5.	Septa Cahyani, Rita Wiryasaputra, Rendra Gustriansyah, 2018	Identifikasi Huruf Kapital Tulisan Tangan Menggunakan Linear Discriminant Analysis dan Euclidean Distance	Pada penelitian ini menggunakan metode <i>Linear Discriminant Analysis</i> untuk ekstraksi fitur serta <i>euclidean distance</i> sebagai <i>classifier</i> . Dari pengujian dengan menggunakan <i>k-fold cross validation</i> ini menghasilkan rata-rata tingkat akurasi sebesar 75,39% dan waktu komputasi sebesar 0.41999 detik dengan menggunakan citra sebesar 50x50 piksel.
6.	Rizka Purmaya Sari, Ulla Delfana Rosiani, Arie Rachmat Syulistyo, 2020	Implementasi Metode <i>Linear Discriminant Analysis</i> Untuk Deteksi Kematangan Pada Buah Stroberi	Dalam penelitian ini menggunakan ruang warna HSV untuk segmentasi citra serta metode Linear Discriminant Analysis sebagai <i>classifier</i> . Hasil akurasi yang diperoleh dari keseluruhan dataset dari 210 dataset hasilnya adalah 177 benar dan 43 salah dengan tingkat akurasi 84,28%. Dalam penelitian ini diharapkan untuk ukuran <i>image</i> yang diolah bisa sama ukurannya.
7.	Edwar Budiman,	Pendeteksi Jenis Autis pada	Masa yang sangat penting bagi

No	Penulis	Judul	Keterangan
	Edy Santoso, Tri Afirianto, 2017	Anak Usia Dini Menggunakan Metode <i>Linear Discriminant Analysis</i> (LDA)	pertumbuhan anak, namun orang tua tidak mengetahui hambatan yang terjadi pada perkembangan anak. Di dalam penelitian ini dibagi menjadi 3 kelas yaitu autisme ringan, sedang dan berat. Data latih yang digunakan sebanyak 75 dan data uji sebanyak 25. Dari hasil pengujian tersebut didapatkan 22 data yang sesuai dan terdapat 3 data yang tidak sesuai, dengan akurasi 88%. Dari penelitian ini dapat disimpulkan bahwa data latih sangat mempengaruhi terhadap hasil akurasi dari sistem pendeteksi autisme. Jumlah data latih tersebut mempengaruhi pola yang telah dibuat oleh sistem sehingga data mengklasifikasikan dengan lebih akurat. Hal ini terjadi karena metode LDA mengolah data latih untuk dijadikan sebagai patokan dalam menentukan data uji tersebut ke dalam suatu kelas.
7.	Diding Suhandy dan Meinilwita Yulia 2018	<i>Luwak Coffee Classification Using UV-Vis Spectroscopy Data: Comparison of Linear Discriminant Analysis and Support Vector Machine Methods</i>	Penelitian ini melakukan perbandingan antara metode <i>Linear Discriminant Analysis</i> dan <i>Support Vector machines Classification</i> (SVMC). Data yang digunakan sebanyak 50 sampel kopi luwak murni dan 50 sampel kopi luwak non murni yang diambil dengan metode UV spectrometer pada mode transmisi. Hasilnya menunjukkan bahwa UV-Vis spectroscopy dikombinasikan dengan LDA dan SVMC sangat efektif untuk mengklasifikasikan sampel kopi luwak dan non luwak. Pada pengklasifikasian dengan menggunakan metode SVM didapatkan akurasi terbesar 98% dan LDA terbesar 96%. Namun karena kesederhanaan dan volume perhitungan yang dibutuhkan, pada penelitian kali ini metode LDA lebih unggul daripada metode SVMC.
9.	Retno Nugroho Whidhiasih, Nursinta Adi Wahanani, Supriyanto 2013	Klasifikasi Buah Belimbing Berdasarkan Citra Red-Green-Blue Menggunakan KNN Dan LDA	Penelitian ini membandingkan algoritma KNN dan Linear Discriminant Analysis dengan menggunakan citra RGB. Tahap preprocessing yang digunakan dalam penelitian ini adalah analisis

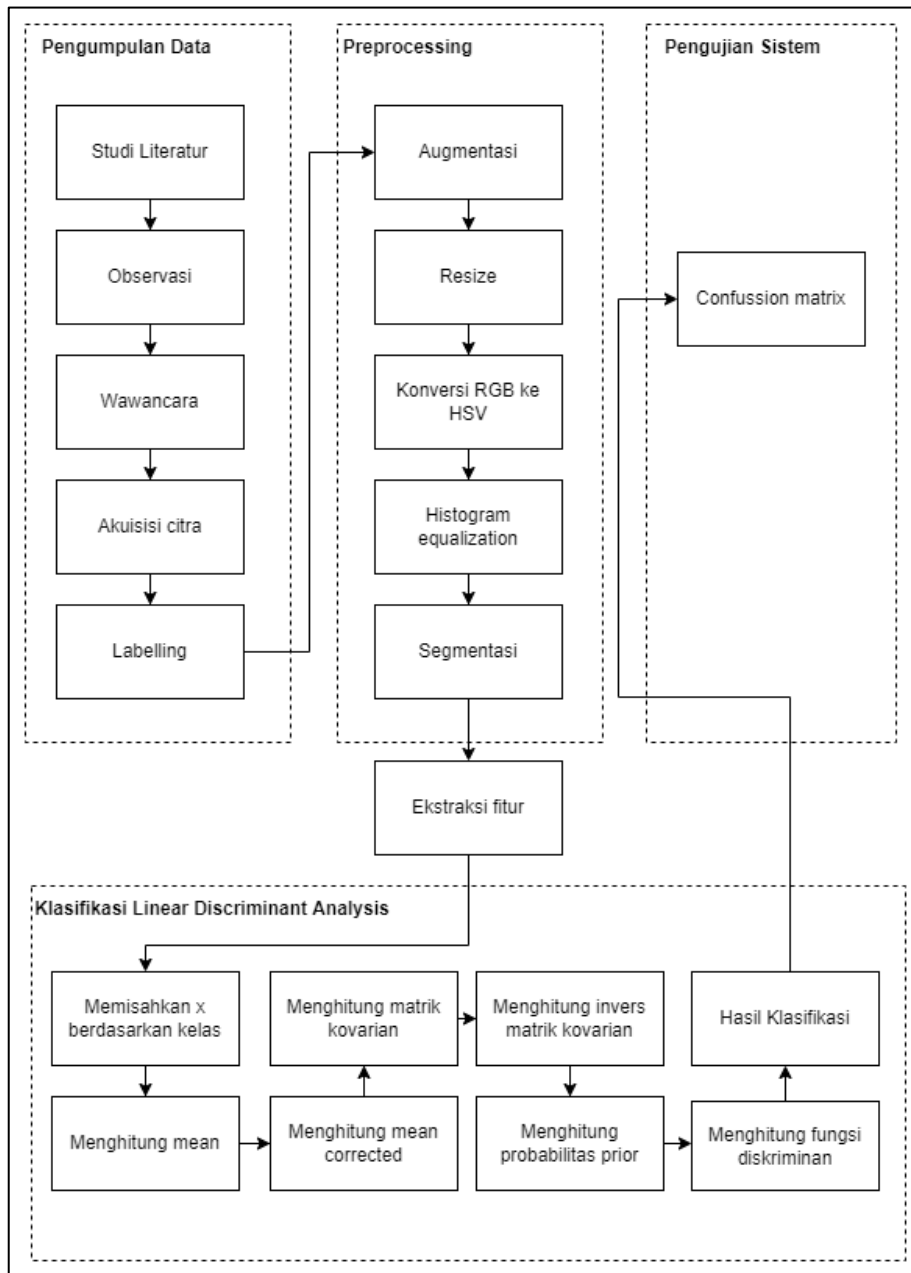
No	Penulis	Judul	Keterangan
			<p>komponen utama (PCA). Data yang digunakan pada penelitian ini adalah data berformat .bmp yang terdiri dari 4 tahap pemanenan (usia 40, 50, 60 dan 70 hari) dengan jumlah masing-masing 75 buah. Data ini merupakan hasil reduksi PCA. Pada penggunaan metode KNN data yang diujikan dibagi menjadi dua jenis data yaitu citra RGB dan data citra RGB. Pada penggunaan metode LDA digunakan data RGB yang diklasifikasikan dengan jarak mahalanobis. Tingkat akurasi yang didapatkan dari hasil klasifikasi (manis, sedang, asam) dengan menggunakan metode KNN 2 variabel RGB menghasilkan akurasi sebesar 80%. Hasil akurasi dengan menggunakan 3 variabel RGB menghasilkan akurasi sebesar 90%, sedangkan menggunakan teknik LDA maupun mahalanobis menghasilkan akurasi sebesar 91%.</p>
10.	Choe Lips haw, Wan Ishak Wan Ismail, Siti Kairunniza-Bejo, Adam Putih, Ramin Shamshiri, 2014	Colour vision to determine paddy maturity	<p>Dari hasil penelitian tersebut menunjukkan bahwa warna bulir padi berubah dari hijau ke kuning bersamaan dengan kematangannya dengan nilai hue akhir di range $32.3^{\circ} \pm 2^{\circ}$ untuk MR219. Daun padi juga berubah dari hijau ke kuning dengan nilai akhir hue sebesar $48^{\circ} \pm 18^{\circ}$. Warna daun mungkin tidak bisa menjadi indikator yang baik untuk menentukan kematangan, ini dikarenakan daun mengering dengan cepat saat akan matang dimana beberapa daun juga akan melengkung menjadi bentuk jarum. Selain itu daun padi cenderung juga menjadi inang bagi beberapa hama. Hama menyerang daun dan menyebabkan daun menjadi keriting dan berubah warna menjadi putih pucat. Ketika padi tumbuh menuju ke fase matang, proses de-greening (penguningan) dihasilkan dari hilangnya nitrogen yang diserap oleh bulir matang yang menyebabkan daun berubah warna dari hijau menjadi kuning. Serta dari bulir bertambah seiring dengan kematangannya, tapi</p>

No	Penulis	Judul	Keterangan
			nilai hue dari bulir dan daun menurun dari hijau ke kuning. Meskipun demikian, berat dari bulir mungkin tidak bertambah tetapi akan bertahan ketika mencapai kematangan optimal.

BAB III METODOLOGI PENELITIAN DAN PENGEMBANGAN SISTEM

3.1 Metodologi Penelitian

Pada bab ini akan membahas metodologi penelitian dan analisis kebutuhan yang akan digunakan dalam penelitian ini. Metode penelitian yang digunakan dalam penelitian ini adalah metodologi penelitian kuantitatif. Metode penelitian kuantitatif merupakan metode yang digunakan untuk meneliti pada populasi atau sampel tertentu dengan menggunakan teknik pengambilan sampel secara acak, pengumpulan data memakai instrumen penelitian, analisis data yang bersifat kuantitatif atau statistik dengan tujuan untuk menguji hipotesis yang telah ditetapkan (Sugiyono, 2015).



Gambar 3.1 Metodologi penelitian.

Pada kerangka penelitian pada Gambar 3.1 diatas, tahapan metodologi penelitian yang meliputi pengumpulan data, *preprocessing*, ekstraksi fitur, identifikasi *Linear Discriminant Analysis*, dan pengujian.

3.1.1 Pengumpulan Data

Data adalah salah satu hal yang sangat penting dalam sebuah penelitian. Tanpa adanya suatu data dalam penelitian, maka penelitian tersebut tidak dapat dilakukan (DQLab, 2021). Oleh karena itu, dalam penelitian ini memerlukan data yang harus dikumpulkan terlebih dahulu. Selain itu, data yang digunakan juga harus dapat dipertanggungjawabkan kebenarannya.

Dalam penelitian ini menggunakan data citra padi yang diambil melalui di salah satu sawah yang terdapat di Ambarawa. Proses pengumpulan data ini memiliki tujuan agar mendapatkan citra padi yang akan digunakan dalam proses identifikasi kematangan padi. Berikut adalah tahap pengumpulan data yang digunakan dalam penelitian ini.

a. Studi Pustaka

Studi pustaka dilakukan untuk mencari referensi. Studi pustaka bertujuan untuk menguatkan landasan literatur dan landasan teori. Studi pustaka juga dilakukan untuk membandingkan beberapa penelitian yang telah dilakukan untuk menguatkan landasan teori yang digunakan pada penelitian ini. Studi pustaka yang terdapat pada penelitian-penelitian terdahulu digunakan sebagai bahan pertimbangan dalam menentukan langkah, metode yang digunakan pada penelitian ini.

b. Observasi

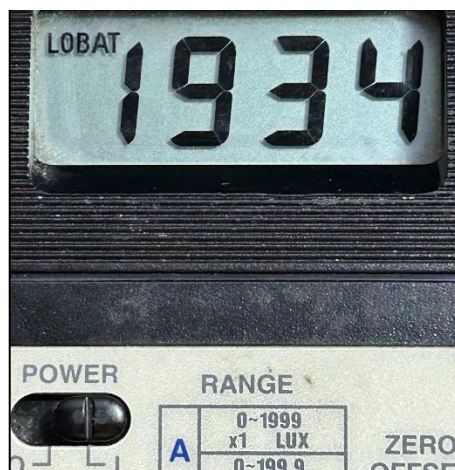
Observasi yang dilakukan pada penelitian ini dilakukan di salah satu sawah yang terdapat di Ambarawa. Observasi yang dilakukan bertujuan untuk mendapatkan data dalam penelitian berupa citra padi. Data citra padi tersebut selanjutnya akan diolah untuk mendapatkan data nilai warna yang yang kemudian akan diolah untuk mengidentifikasi kematangan padi.

c. Wawancara

Tahap wawancara merupakan untuk memperoleh informasi secara langsung dari ahli pertanian dengan mengajukan beberapa pertanyaan seputar padi, kematangan padi dan waktu panen padi. Dalam tahapan ini wawancara dilakukan dengan ahli padi dengan mengajukan beberapa pertanyaan seputar padi yang digunakan untuk menunjang penelitian ini.

d. Akuisisi Citra

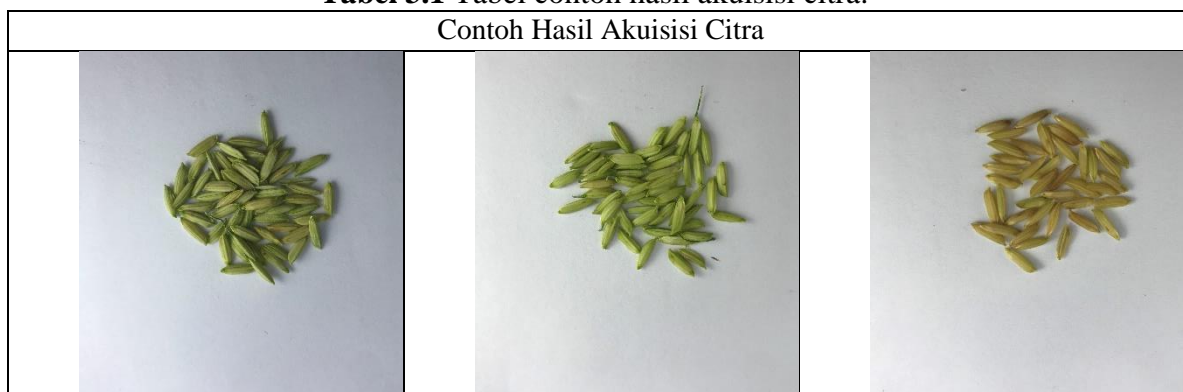
Tahap akuisisi citra atau tahapan pengambilan gambar dilakukan pada ruangan tertutup menggunakan *studio box* buatan. *Studio box* buatan dibuat dengan menggunakan kardus yang dilapisi dengan kertas karton putih untuk mengurangi pantulan cahaya yang dihasilkan dua buah lampu LED Philips 20 watt sebanyak 2 buah. Selanjutnya, dalam penelitian ini dilakukan penghitungan jumlah cahaya dengan menggunakan *lux meter* dengan *range* 0 – 1999 yang diperoleh nilai 1934 lux. Berikut adalah hasil pembacaan cahaya dengan menggunakan *lux meter*.



Gambar 3.2 Hasil pembacaan dengan *lux meter*.

Pengambilan gambar dilakukan dengan jarak 15 cm dari objek foto dengan resolusi foto sebesar 4032 x 3024. Pengambilan gambar dilakukan dengan menggunakan kamera *smartphone* 12MP pada sore hari atau jam 5 sore. Gambar tersebut kemudian akan diolah yang akan digunakan sebagai data latih untuk pengetahuan awal dari sistem serta digunakan sebagai data uji yang digunakan untuk menguji performa dari sistem. Diperoleh sebanyak 40 data dengan masing-masing citra sebanyak 40 bulir padi yang selanjutnya akan di augmentasi menjadi sebanyak 400 untuk memaksimalkan performa. Berikut adalah citra hasil akuisi pada penelitian ini.



Tabel 3.1 Tabel contoh hasil akuisisi citra.



e. *Labelling*

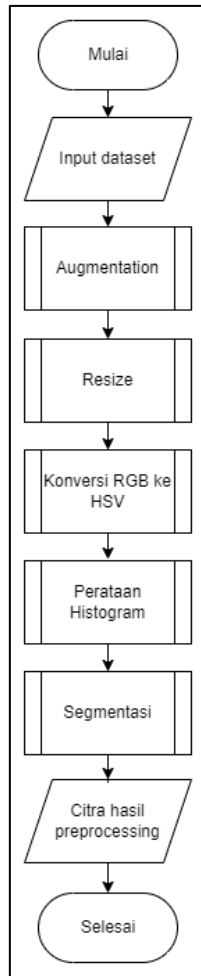
Tahap *labelling* merupakan tahapan memberi label pada citra. Label yang digunakan pada penelitian ini ada matang dan mentah. Tahapan *labelling* pada penelitian ini dilakukan bersama ahli dalam bidang penelitian agar mendapatkan citra dengan label yang tepat. Berikut merupakan contoh data serta pembagiannya.

Tabel 3.2 Contoh image dari data hasil akuisisi citra setelah *labelling*

No.	Data Hasil Akuisisi Citra	Kategori	Jumlah Data	
			sebelum augmentasi	setelah augmentasi
1.		matang	20	400
2.		mentah	20	400

3.1.2 Data Preprocessing

Tahap data *preprocessing* merupakan tahapan dimana data disiapkan untuk melewati proses pembelajaran sebelum dibentuk menjadi sebuah model. Data citra padi yang telah didapatkan dari proses observasi dan *labelling* bersama ahli selanjutnya akan melewati tahapan *preprocessing*. Tahapan *preprocessing* yang akan dilakukan pada penelitian ini meliputi *augmentation*, *resize*, konversi RGB ke HSV, perataan histogram (*histogram equalization*) dan segmentasi. Berikut merupakan *flowchart* dari tahapan *preprocessing*. Berikut adalah *flowchart* dari *preprocessing*.








Gambar 3.3 Flowchart preprocessing

1. *Augmentation*

Proses augmentasi data yang bertujuan untuk memperbanyak data dengan bentuk berbeda. Proses augmentasi yang digunakan dalam penelitian ini adalah *rotation* yang memutar citra dengan derajat tertentu, *shear* adalah menggeser citra menurut sumbunya, *zoom* adalah memperbesar maupun menambahkan pixel untuk memperbesar gambar, dan *flip* adalah membalikkan masing-masing baris *pixel* atau kolom *pixel* (Mikołajczyk & Grochowski, 2019). Dari Tabel 3.2 berikut dapat dilihat contoh hasil *augmentation* beserta hasil dari setiap *augmentation*.

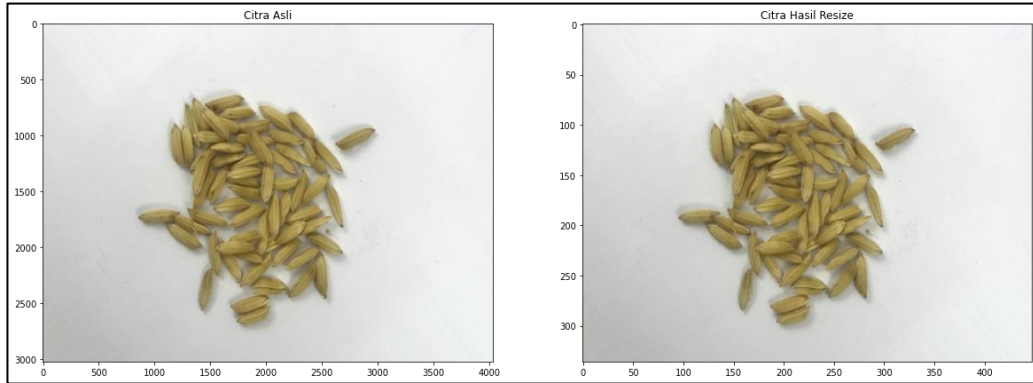
Tabel 3.3 Contoh hasil augmentasi

Asli	<i>Rotation</i>	<i>Shear</i>	<i>Zoom</i>	<i>Flip</i>
				

2. *Resize*

Ukuran citra padi yang dihasilkan cukup besar, sehingga mempengaruhi kecepatan sistem dalam melakukan pemrosesan. Untuk itu diperlukan untuk merubah ukuran menjadi

lebih. Dalam penelitian ini citra padi *resize* menjadi dari citra awal yaitu 4032 x 3024 menjadi 432 x 324. Dari Gambar 3.3 berikut dapat dilihat contoh hasil *resize*.



Gambar 3.4 Contoh hasil *resize image*.

3. Konversi RGB ke HSV

Citra padi awal yang merupakan hasil dari akuisisi citra memiliki ruang warna RGB. Dalam penelitian ini ruang warna yang akan digunakan pada tahapan preprocessing merupakan HSV. Ruang warna HSV digunakan karena memiliki warna yang murni belum bercampur seperti RGB. Dimana Hue warna dari cahaya yang tampak (merah, jingga, kuning, hijau, biru, dan ungu). Saturation merepresentasikan merupakan tingkat kemurnian warna, semakin tinggi nilai saturasinya hasil warna akan menjadi lebih murni. (Sulistyo, 2017). Tahap *preprocessing* yang dilakukan dalam penelitian ini dengan melakukan konversi nilai nilai RGB pada setiap piksel kedalam HSV dimana *hue* (H) diperoleh dari persamaan 2.17, *saturation* (S) diperoleh dari persamaan 2.18, *value* (V) diperoleh dari persamaan 2.19.

Sebelum melakukan konversi nilai RGB ke HSV, dilakukan terlebih dahulu proses normalisasi nilai RGB

- a. Mencari nilai R menggunakan persamaan 2.11

$$R'_{1,1} = \frac{R'_{1,1}}{R'_{1,1} + G'_{1,1} + B'_{1,1}} = \frac{199}{199 + 205 + 219} = 0,319$$

$$R'_{1,2} = \frac{R_{1,2}}{R'_{1,2} + G'_{1,2} + B'_{1,2}} = \frac{197}{197 + 203 + 218} = 0,319$$

$$R'_{1,3} = \frac{R_{1,3}}{R'_{1,3} + G'_{1,3} + B'_{1,3}} = \frac{198}{198 + 204 + 220} = 0,318$$

$$\vdots$$

$$R'_{432,1} = \frac{R_{335,1}}{R'_{335,1} + R'_{335,2} + R'_{335,3}} = \frac{243}{243 + 246 + 255} = 0,327$$

- b. Mencari nilai G dengan menggunakan persamaan 2.12

$$G'_{1,1} = \frac{G'_{1,1}}{R'_{1,1} + G'_{1,1} + B'_{1,1}} = \frac{205}{199 + 205 + 219} = 0,329$$

$$G'_{1,2} = \frac{G'_{1,2}}{R'_{1,2} + G'_{1,2} + B'_{1,2}} = \frac{52}{197 + 203 + 218} = 0,328$$

$$G'_{1,3} = \frac{G'_{1,3}}{R'_{1,3} + G'_{1,3} + B'_{1,3}} = \frac{49}{198 + 204 + 220} = 0,328$$

$$\vdots$$

$$G'_{432,1} = \frac{G'_{335,1}}{G'_{335,1} + G'_{335,2} + G'_{335,3}} = \frac{246}{243 + 246 + 255} = 0,330$$

c. Mencari nilai B dengan menggunakan persamaan 2.13

$$B_{1,1} = \frac{B'_{1,1}}{R'_{1,1} + G'_{1,1} + B'_{1,1}} = \frac{219}{199 + 205 + 219} = 0,352$$

$$B_{1,2} = \frac{B'_{1,2}}{R'_{1,2} + G'_{1,2} + B'_{1,2}} = \frac{218}{197 + 203 + 218} = 0,353$$

$$B_{1,3} = \frac{B'_{1,3}}{R'_{1,3} + G'_{1,3} + B'_{1,3}} = \frac{220}{198 + 204 + 220} = 0,354$$

⋮

$$B_{432,1} = \frac{B'_{335,1}}{R'_{335,1} + G'_{335,2} + B'_{335,3}} = \frac{255}{243 + 246 + 255} = 0,343$$

Sesudah memperoleh nilai R, G, B dilakukan konversi nilai HSV. Berikut merupakan tahapan-tahapan dalam konversi RGB menjadi HSV

a. Mendefinisikan *value* (V) dengan persamaan

$$V'_{1,1} = \max \{R'_{1,1}, G'_{1,1}, B'_{1,1}\} = 0,352$$

$$V'_{1,2} = \max \{R'_{1,2}, G'_{1,2}, B'_{1,2}\} = 0,353$$

$$V'_{1,3} = \max \{R'_{1,3}, G'_{1,3}, B'_{1,3}\} = 0,354$$

⋮

$$V'_{335,1} = \max \{R'_{335,1}, G'_{335,1}, B'_{335,1}\} = 0,343$$

b. Mendefinisikan *saturation* (S) dengan persamaan

$$S'_{1,1} = 1 - \frac{\min \{R'_{1,1}, G'_{1,1}, B'_{1,1}\}}{v'} = 1 - \frac{0,319}{0,352} = 0,094$$

$$S'_{1,2} = 1 - \frac{\min \{R'_{1,2}, G'_{1,2}, B'_{1,2}\}}{v'} = 1 - \frac{0,329}{0,353} = 0,068$$

$$S'_{1,3} = 1 - \frac{\min \{R'_{1,3}, G'_{1,3}, B'_{1,3}\}}{v'} = 1 - \frac{0,318}{0,354} = 0,102$$

⋮

$$S'_{432,1} = 1 - \frac{\min \{R'_{336,1}, G'_{336,1}, B'_{336,1}\}}{v} = 1 - \frac{0,327}{0,343} = 0,047$$

c. Mendefinisikan hue (H) dengan persamaan

$$H = \begin{cases} 0 & S = 0 \\ \left(\frac{60 \times (G' - B')}{\Delta} \right), V = R' & \\ 60 \times \left(\frac{B' - R'}{\Delta} + 2 \right) & , V = G' \\ 60 \times \left(\frac{R' - G'}{\Delta} + 4 \right) & , V = B' \end{cases}$$

$$H = H + 360 \text{ jika } H < 0$$

$$\left(\frac{60 \times (G' - B')}{\Delta} \right), V = R'$$

$$H'_{1,1} = \left(\frac{60 \times (G'_{1,1} - B'_{1,1})}{G'_{1,1} \times V'_{1,1}} \right) = \left(\frac{60 \times (0,329 - 0,352)}{0,329 \times 0,352} \right) = 11,896$$

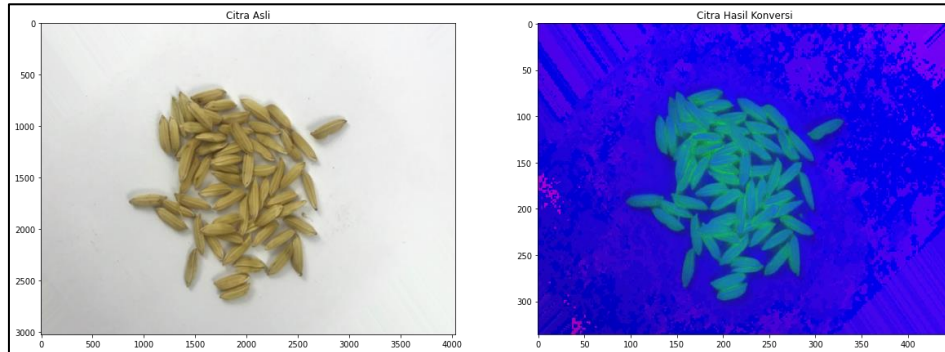
$$H'_{1,2} = \left(\frac{60 \times (G'_{1,2} - B'_{1,2})}{G_{1,2} \times V_{1,2}} \right) = \left(\frac{60 \times (0,328 - 0,353)}{0,328 \times 0,353} \right) = 12.931$$

$$H'_{1,3} = \left(\frac{60 \times (G'_{1,3} - B'_{1,3})}{G_{1,3} \times V_{1,3}} \right) = \left(\frac{60 \times (0,328 - 0,354)}{0,328 \times 0,354} \right) = 13.448$$

⋮

$$H'_{432,1} = \left(\frac{60 \times (G'_{335,1} - B_{335,1})}{G_{336,1} \times V_{336,1}} \right) = \left(\frac{60 \times (0,330 - 0,343)}{0,330 \times 0,343} \right) = 6.903$$

Dari Gambar 3.4 berikut dapat dilihat contoh hasil konversi RGB ke HSV.



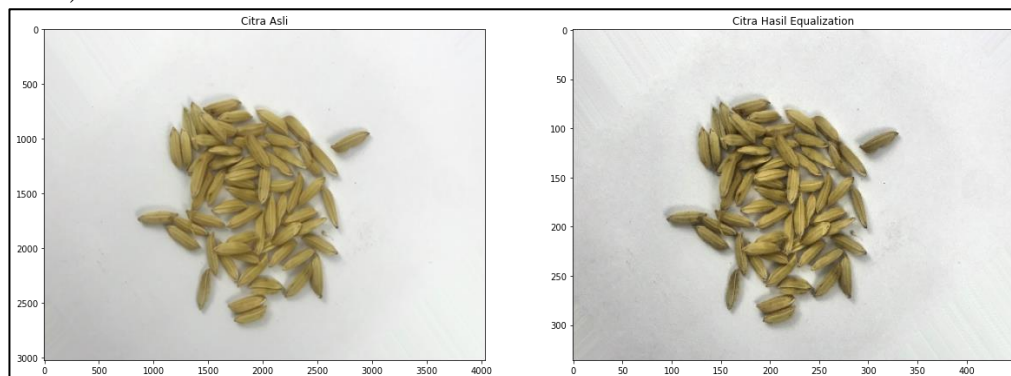
Gambar 3.5 Contoh hasil konversi RGB ke HSV.

4. Histogram *Equalization*

Citra yang telah melalui proses segmentasi kemudian akan dilanjutkan dengan proses *histogram equalization*. Proses *histogram equalization* merupakan proses perataan histogram. Dalam penelitian ini dilakukan proses perataan histogram (*histogram equalization*) dengan menggunakan ruang warna HSV agar diperoleh penyebaran citra yang lebih merata sehingga akan menghasilkan citra dengan penyebaran *brightness* maupun *contrast* yang merata sehingga tidak terlalu gelap dan pucat. Berikut merupakan proses *histogram equalization* (Sutojo, 2017).

- a. Citra HSV yang telah melewati proses segmentasi selanjutnya dipisahkan menjadi nilai *Hue*, *Saturation*, dan *Value*.
- b. Melakukan perataan histogram terhadap *value*
- c. Menggabungkan nilai *Hue*, *Saturation*, dan *value* perataan histogram.

Dari Gambar 3.5 berikut dapat dilihat contoh hasil perataan histogram (*histogram equalization*).



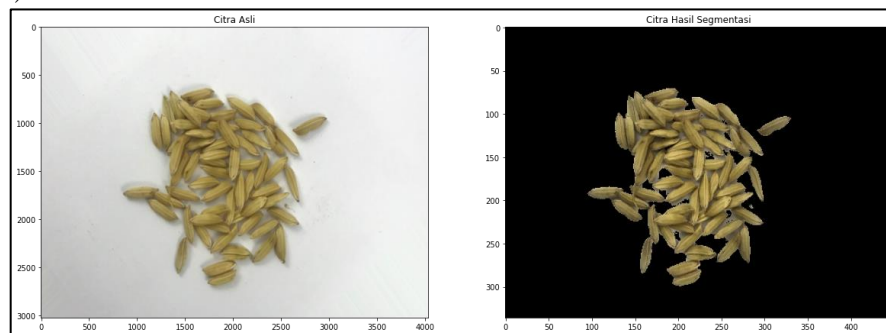
Gambar 3.6 Contoh hasil *histogram equalization*.

5. Segmentasi

Setelah melakukan konversi ke dalam ruang warna HSV selanjutnya dilakukan segmentasi yang bertujuan untuk memisahkan antara background dan objek. Dalam penelitian ini segmentasi dilakukan menggunakan nilai *upper* dan *lower* yang bertujuan untuk memisahkan antara objek dan latar belakang. Berikut merupakan tahapan metode segmentasi dengan deteksi warna HSV (Giannakopoulos, 2008)

- a. Pemilihan sampel piksel sebagai acuan warna untuk membentuk segmen yang diinginkan. Citra digital menggunakan model warna RGB sebagai standar acuan warna, oleh karena itu proses awal pada metode ini memerlukan konversi model warna RGB ke HSV. Untuk membentuk segmen sesuai dengan warna yang diinginkan maka ditentukan nilai toleransi pada setiap dimensi warna HSV, kemudian nilai toleransi tersebut digunakan dalam perhitungan proses adaptive threshold. Hasil dari proses *threshold* tersebut akan membentuk segmen area dengan warna sesuai toleransi yang diinginkan.
- b. Menentukan sampel citra yang akan digunakan sebagai acuan warna, pada penelitian ini sampel piksel yang digunakan adalah citra padi dengan kategori mentah paling pertama, kategori mentah paling terakhir, kategori matang paling pertama, dan kategori matang paling terakhir.
- c. Sampel citra yang telah dikonversi dari ruang warna RGB ke dalam ruang warna HSV kemudian dicari nilai H_{min} , S_{min} , V_{min} dan H_{max} , S_{max} , V_{max} dengan menggunakan citra padi dengan kategori mentah paling awal sebagai *lower green*, kategori mentah paling akhir sebagai *upper green*, kategori matang pertama sebagai *lower brown*, dan kategori matang paling akhir sebagai *upper brown*. Nilai yang digunakan untuk *lower_green* = [16, 43, 40], *upper_green* = [179, 255, 255], *lower_brown* = [4, 88, 43], dan *upper_brown* = [9, 243, 255].
- d. Setelah mendapatkan nilai yang digunakan sebagai acuan, dilanjutkan dengan masking citra berdasarkan nilai acuan warna tersebut. Apabila nilai warna yang ada pada citra tidak berada pada range acuan maka akan berwarna hitam (HSV (0, 0, 0) atau RGB (0, 0, 0)).

Dari Gambar 3.7 berikut dapat dilihat contoh hasil perataan histogram (*histogram equalization*).

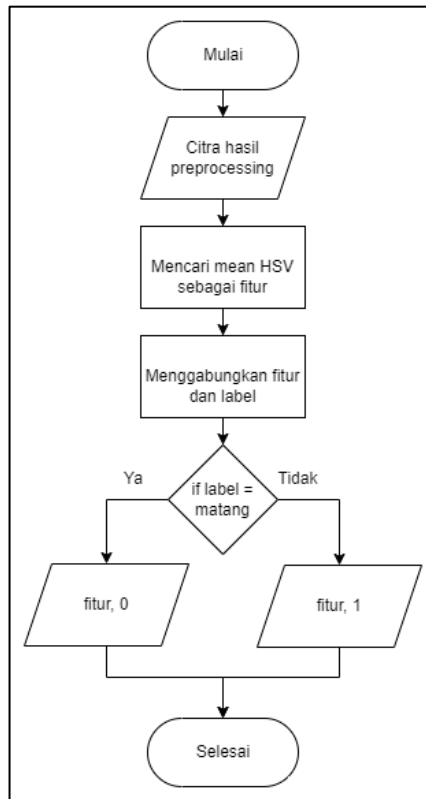


Gambar 3.7 Contoh hasil segmentasi.

3.1.3 Ekstraksi Fitur

Setelah melewati proses segmentasi selanjutnya dilakukan ekstraksi fitur dengan menggunakan rata-rata dari nilai HSV. Nilai rata-rata dari masing-masing citra tersebut selanjutnya digabungkan dengan label. Label 0 merupakan citra dengan kelas matang dan

label 1 merupakan kelas mentah. Sehingga fitur yang dihasilkan adalah nilai rata-rata HSV, label. Berikut merupakan *flowchart* dalam tahapan ekstraksi fitur.



Gambar 3.8 *flowchart* ekstraksi fitur.

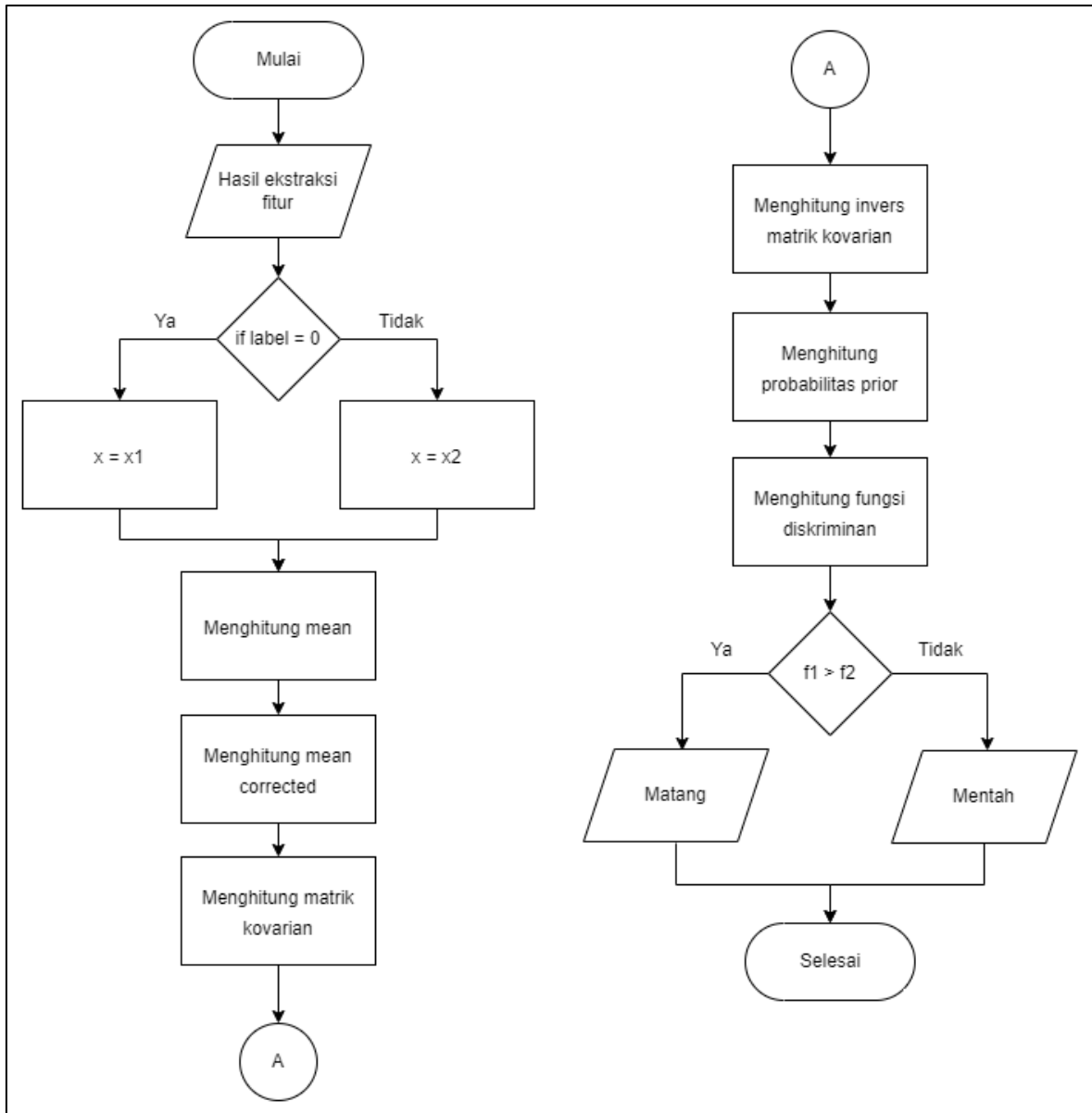
Berikut adalah contoh dari hasil ekstraksi fitur pada penelitian ini.

`[[[2.135828550170068, 11.591830091411564, 11.416281356292517), 0]]`

Gambar 3.9 Contoh hasil ekstraksi fitur.

3.1.4 Linear Discriminant Analysis

Setelah melewati ekstraksi fitur, data selanjutnya akan masuk kedalam proses identifikasi. Selain bertujuan untuk mengidentifikasi data berdasarkan kelasnya, juga bertujuan untuk mengukur seberapa akurat metode *Linear Discriminant Analysis* dalam melakukan identifikasi terhadap tingkat kematangan padi. *Linear Discriminant Analysis* adalah klasifier dengan beberapa data yang telah diketahui labelnya. Data yang telah diketahui labelnya tersebut digunakan dalam menentukan fungsi diskriminan (Ghazali, Eko K. Subha, Galuh M., M. Burhannudin, 2018). Fungsi diskriminan merupakan *output* dari hasil analisis yang dipakai dalam mengelompokkan ke dalam salah satu kelas. Fungsi diskriminan yang terbentuk akan digunakan untuk memisahkan kelompok. Berikut merupakan *flowchart* dalam tahapan *Linear Discriminant Analysis*.



Gambar 3.10 flowchart identifikasi *Linear Discriminant Analysis*.

Berikut merupakan contoh nilai HSV yang dihasilkan dari tahapan sebelumnya (ekstraksi fitur).

Tabel 3.4 Tabel contoh nilai HSV, label, dan kelas.

No	Nilai			Label	Kelas
	H	S	V		
1	2.95	6.63	5.45	0	matang
2	2.53	7.79	8.45	0	matang
3	3.57	5.65	2.34	0	matang
4	3.16	5.47	3.65	0	matang
5	2.58	4.46	2.45	1	mentah
6	2.16	6.22	5.45	1	mentah
7	3.27	3.52	8.23	1	mentah

Berikut merupakan perhitungan dengan menggunakan metode *Linear Discriminant Analysis*

- a. Mengubah dataset yang telah diketahui labelnya menjadi matrix
 $X = \text{features (hue, saturation, value)}$
 $Y = \text{class (matang dan mentah)}$

$$x = \begin{bmatrix} 2.95 & 6.63 & 5.45 \\ 2.53 & 7.79 & 8.45 \\ 3.57 & 5.65 & 2.34 \\ 3.16 & 5.47 & 3.65 \\ 2.58 & 4.46 & 2.45 \\ 2.16 & 6.22 & 5.45 \\ 3.27 & 3.52 & 8.23 \end{bmatrix} \quad y = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

- b. Memisahkan x menjadi x_1 dan x_2 sesuai dengan kelasnya

$$x_1 = \begin{bmatrix} 2.95 & 6.63 & 5.45 \\ 2.53 & 7.79 & 8.45 \\ 3.57 & 5.65 & 2.34 \\ 3.16 & 5.47 & 3.65 \end{bmatrix} \quad x_2 = \begin{bmatrix} 2.58 & 4.46 & 2.45 \\ 2.16 & 6.22 & 5.45 \\ 3.27 & 3.52 & 8.23 \end{bmatrix}$$

- c. Menghitung *mean* (rata-rata) dari setiap kelas

$$\mu_1 = [3.05 \quad 6.38 \quad 4.97], \quad \mu_2 = [2.67 \quad 4.73 \quad 5.38]$$

$$\mu = [2.89 \quad 5.68 \quad 5.15]$$

- d. Menghitung x_i^0 (*mean corrected*) dengan cara $x_i - \mu$

$$x_1^0 = \begin{bmatrix} 0.06 & 0.95 & 0.30 \\ -0.36 & 2.11 & 3.30 \\ 0.68 & -0.02 & -2.81 \\ 0.27 & -0.21 & -1.50 \end{bmatrix}, \quad x_2^0 = \begin{bmatrix} -0.31 & -1.22 & -2.70 \\ -0.73 & 0.55 & 0.30 \\ 0.39 & -2.16 & 3.08 \end{bmatrix}$$

- e. Menghitung matrik kovarian dari setiap data di masing-masing kelas dan matriks kovarian global

$$c_i = \frac{(x_i^0)^T x_i^0}{n_i}$$

$$c_1 = \frac{\begin{bmatrix} 0.06 & 0.95 & 0.30 \\ -0.36 & 2.11 & 3.30 \\ 0.68 & -0.02 & -2.81 \\ 0.27 & -0.21 & -1.50 \end{bmatrix}^T \begin{bmatrix} 0.06 & 0.95 & 0.30 \\ -0.36 & 2.11 & 3.30 \\ 0.68 & -0.02 & -2.81 \\ 0.27 & -0.21 & -1.50 \end{bmatrix}}{4}$$

$$c_1 = \frac{\begin{bmatrix} 0.06 & -0.36 & 0.68 & 0.27 \\ 0.95 & 2.11 & -0.02 & -0.21 \\ 0.3 & 3.3 & -2.81 & -1.5 \end{bmatrix} \times \begin{bmatrix} 0.06 & 0.95 & 0.30 \\ -0.36 & 2.11 & 3.30 \\ 0.68 & -0.02 & -2.81 \\ 0.27 & -0.21 & -1.50 \end{bmatrix}}{4}$$

$$c_1 = \begin{bmatrix} 0.17 & -0.19 & -0.87 \\ -0.19 & 1.35 & 1.91 \\ -0.87 & 1.91 & 5.28 \end{bmatrix}$$

$$c_2 = \frac{\begin{bmatrix} -0.31 & -1.22 & -2.70 \\ -0.73 & 0.55 & 0.30 \\ 0.39 & -2.16 & 3.08 \end{bmatrix}^T \begin{bmatrix} -0.31 & -0.73 & -0.39 \\ -1.22 & 0.55 & -2.16 \\ -2.70 & 0.30 & 3.08 \end{bmatrix}}{3}$$

$$c_2 = \frac{\begin{bmatrix} -0.31 & -0.73 & 0.39 \\ -1.22 & 0.55 & -2.16 \\ 2.70 & 0.30 & 3.08 \end{bmatrix} \times \begin{bmatrix} -0.31 & -1.22 & -2.70 \\ -0.73 & 0.55 & 0.30 \\ 0.39 & -2.16 & 3.08 \end{bmatrix}}{3}$$

$$c_2 = \begin{bmatrix} 0.26 & -0.29 & 0.60 \\ -0.29 & 2.14 & -1.07 \\ 0.60 & -1.07 & 5.62 \end{bmatrix}$$

$$C(1,1) = \frac{4}{7} \times (0.17) + \frac{3}{7} \times (0.26) = 0.21$$

$$C(1,2) = \frac{4}{7} \times (-0.19) + \frac{3}{7} \times (-0.29) = -0.23$$

$$C(1,3) = \frac{4}{7} \times (-0.87) + \frac{3}{7} \times (0.60) = -0.24$$

$$C(2,1) = \frac{4}{7} \times (-0.19) + \frac{3}{7} \times (-0.29) = -0.23$$

$$C(2,2) = \frac{4}{7} \times (1.35) + \frac{3}{7} \times (2.14) = 1.69$$

$$C(2,3) = \frac{4}{7} \times (1.91) + \frac{3}{7} \times (-1.07) = 0.63$$

$$C(3,1) = \frac{4}{7} \times (-0.87) + \frac{3}{7} \times (0.60) = -0.24$$

$$C(3,2) = \frac{4}{7} \times (1.91) + \frac{3}{7} \times (-1.07) = -0.63$$

$$C(3,3) = \frac{4}{7} \times (5.28) + \frac{3}{7} \times (5.62) = 5.43$$

$$C = \begin{bmatrix} 0.21 & -0.23 & -0.24 \\ -0.23 & 1.69 & 0.63 \\ -0.24 & 0.63 & 5.43 \end{bmatrix}$$

f. Menghitung invers dari matrik kovarian

$$C^{-1} = \begin{bmatrix} 5.90 & 0.75 & 0.17 \\ 0.75 & 0.71 & -0.05 \\ 0.17 & -0.05 & 0.20 \end{bmatrix}$$

g. Menghitung probabilitas prior setiap kelas dengan menghitung banyak data di setiap kelas dibagi dengan banyak data.

$$p = \begin{bmatrix} \frac{4}{7} \\ \frac{3}{7} \\ \frac{1}{7} \end{bmatrix} = \begin{bmatrix} 0.57 \\ 0.43 \end{bmatrix}$$

h. Menghitung fungsi diskriminan, dimana f_i yang dipilih yang paling maksimal sebagai keputusan kelasnya

$$f_i = \mu_i C^{-1} x_k^T - \frac{1}{2} \mu_i C^{-1} \mu_i^T + \ln(p_i) \dots \dots \dots (3.1)$$

Tabel 3.5 Fungsi diskriminan

f_1	f_2
59.19	57.05
60.53	58.39
63.69	60.58
54.35	53.04
32.70	34.73
37.82	38.98
49.74	51.36

- i. Membandingkan nilai f_1 dan f_2 . Jika $f_1 > f_2$ maka masuk kelas matang dan $f_1 < f_2$ maka masuk kelas mentah.

Tabel 3.6 Hasil perbandingan nilai f_1 dan f_2

f_1	f_2	Hasil	Prediksi	Aktual	Hasil Identifikasi
59.19	57.05	0.00	Matang	Matang	Benar
60.53	58.39	0.00	Matang	Matang	Benar
63.69	60.58	0.00	Matang	Matang	Benar
54.35	53.04	0.00	Matang	Matang	Benar
32.70	34.73	1.00	Mentah	Mentah	Benar
37.82	38.98	1.00	Mentah	Mentah	Benar
49.74	51.36	1.00	Mentah	Mentah	Benar

3.1.5 Model Linear Discriminant Analysis

Hasil dari proses pelatihan dataset adalah model *Linear Discriminant Analysis* dengan bentuk *best_model.sav*. File tersebut berisi nilai training, label masing-masing data yang telah disimpan sebelumnya untuk melakukan klasifikasi kematangan padi.

3.1.6 Pengujian

Pengujian dilakukan terhadap dataset yang telah dikumpulkan sebanyak 800 dengan beberapa perbandingan data *training* dan data *testing*, yakni 70% : 30%, 80% : 20%, dan 90% : 10%. Pengujian dalam penelitian ini menggunakan *confusion matrix*. *Confusion matrix* merupakan sebuah metode untuk mengukur kemampuan sebuah sistem ketika melakukan identifikasi data. Pengukuran ini dilakukan dengan menghitung nilai aktual yang dibandingkan dengan jumlah keseluruhan data dan diperoleh nilai akurasinya.

Confusion matrix memiliki beberapa istilah dalam merepresentasikan identifikasi yang didapatkan. Diantaranya adalah *true positive* (TP) atau jumlah data positif yang teridentifikasi dengan benar oleh sistem yang telah dibuat, *true negative* (TN) atau jumlah data negatif yang teridentifikasi dengan benar oleh sistem yang telah dibuat, *false positive* (FP) atau jumlah data positif yang teridentifikasi salah oleh sistem yang telah dibuat, dan *false negative* (FN) atau jumlah data negatif yang teridentifikasi salah oleh sistem yang telah dibuat.

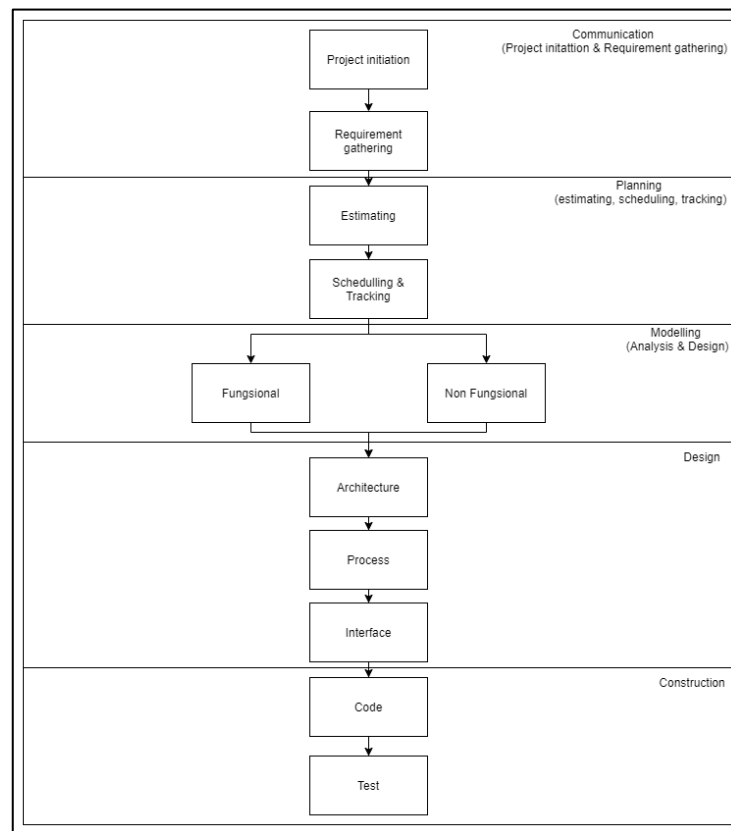
3.2 Metode Pengembangan Sistem

Metode pengembangan yang dilakukan pada penelitian ini merupakan metode *waterfall* yang dikembangkan oleh Pressman. Model *waterfall* merupakan sebuah model klasik yang memiliki sifat sistematis, berurutan dalam pembangunan *software*. Nama model ini sesungguhnya adalah *Linear Sequential Model*. Model ini juga sering disebut

dengan *Classic Life Cycle* atau metode *waterfall*. Yang terdiri dari beberapa tahapan diantaranya, yakni (Pressman, 2015):

- a. *Communication (Project initiation & requirement gathering)*
- b. *Planning (Estimating Scheduling, Tracking)*
- c. *Modeling (Analysis & Design)*
- d. *Construction (Code & Test)*
- e. *Deployment (Delivery, Support, Feedback)*

Akan tetapi, dalam penelitian ini hanya mencapai pada tahap *construction (code & test)*. Berikut adalah alur dari pengembangan sistem yang digunakan pada penelitian ini. Berikut adalah *flowchart* metode pengembangan sistem yang digunakan dalam penelitian ini.



Gambar 3.11 *Flowchart* metode pengembangan sistem.

3.2.1 Communication (Project Initiation & Requirement Gathering)

a. *Project Initiation*

Tahapan ini merupakan tahap analisis permasalahan yang ada yaitu mengetahui tingkat kematangan padi menggunakan sistem yang dibangun.

b. *Requirement Gathering*

Tahapan ini merupakan tahap pengumpulan data yang akan digunakan untuk membangun sistem, tahapan pengumpulan data dalam penelitian ini terdiri dari studi literatur, observasi, wawancara, akuisisi citra, dan *labeling*.

3.2.2 Planning (Estimating, Scheduling, Tracking)

Tahapan ini merupakan tahapan perancangan sistem yang digunakan untuk mengetahui kebutuhan apa saja yang digunakan dalam pembangunan sistem baik

kebutuhan perangkat lunak maupun perangkat keras. Dalam penelitian ini kebutuhan perangkat lunak yang dibutuhkan diantaranya adalah bahasa pemrograman yang akan digunakan, *code editor*, *framework*, *server*, dan *browser* yang digunakan untuk menunjang dalam pembangunan sistem. Sementara itu, untuk kebutuhan perangkat keras yang digunakan adalah laptop dan perangkat pendukung yang digunakan untuk akuisisi citra.

3.2.3 Modeling (Analysis & Design)

Tahapan ini adalah tahap desain model arsitektur sistem, di mana *modeling* berkonsentrasi pada pengembangan struktur data sistem, arsitektur perangkat lunak, tampilan antarmuka, dan algoritma yang digunakan dalam membangun sistem. Analisis kebutuhan bertujuan untuk menggambarkan kebutuhan sistem agar sistem dapat berfungsi sebagaimana mestinya. Pada tahapan ini, proses pemodelan dibagi menjadi dua tahapan yaitu: *analysis* dan *design*.

1. Analysis

Tahapan *analysis* dalam *modelling* bertujuan untuk merinci kebutuhan apa saja yang dibutuhkan dari sistem yang dibuat sehingga bisa berjalan sesuai perencanaan. Di dalam analisis kebutuhan sistem terdapat dua kebutuhan yakni kebutuhan fungsional dan non-fungsional.

a. Kebutuhan fungsional

Untuk memberi gambaran dan informasi yang terjadi dalam sistem identifikasi kematangan padi dan *output* yang dihasilkan oleh sistem dilakukan analisa kebutuhan fungsional. Berikut merupakan kebutuhan fungsional yang dibutuhkan dalam pengembangan sistem ini, diantaranya adalah sebagai berikut:

1. Sistem dapat melakukan *preprocessing* pada dataset.
2. Sistem dapat melakukan pembuatan model *Linear Discriminant Analysis*.
3. Sistem dapat mendeteksi kematangan padi menggunakan metode *Linear Discriminant Analysis*
4. Sistem dapat menampilkan hasil deteksi kematangan padi.

b. Kebutuhan non-fungsional

Untuk mengetahui spesifikasi kebutuhan dan properti yang digunakan ketika membangun sistem dilakukan analisis kebutuhan non-fungsional. Spesifikasi kebutuhan ini melibatkan analisis perangkat keras (*hardware*) dan analisis perangkat lunak (*software*). Adapun kebutuhan non-fungsional yang dibutuhkan dalam pengembangan sistem ini adalah sebagai berikut.

Tabel 3.7 Kebutuhan non-fungsional

No	Perangkat Keras	Keterangan
1	<i>Laptop</i>	Asus A442UR
2	<i>Processor</i>	Intel Core i7 @2,90 GHz
3	RAM	8GB DDR4
4	<i>Storage</i>	SSD 237GB
5	<i>Graphic</i>	NVIDIA GEFORCE 930 MX
6	Perangkat <i>input</i> dan <i>output</i>	Keyboard, mouse, <i>mini diy studio box</i> , lampu LED 10 watt Philips, <i>smartphone</i> , dan <i>lux meter</i>

Dalam penelitian ini menggunakan beberapa perangkat lunak (*software*) pendukung yang terdiri dari beberapa komponen yaitu:

Tabel 3.8 Kebutuhan non-fungsional.

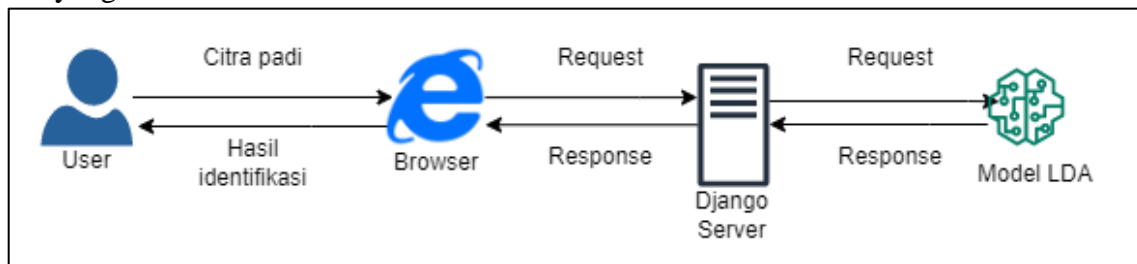
No	Perangkat Lunak	Keterangan
1	Operating System	Windows 10 Pro 64 bit
2	Python, PHP	Bahasa pemrograman
3	Visual Studio Code	<i>Code editor</i>
4	Django	<i>Framework</i>
5	XAMPP	Perangkat Lunak sebagai server lokal
6	Microsoft Edge	<i>Web browser</i>
7	Draw.io	Website untuk desain diagram dan desain <i>interface</i>

5. Design

Tahap desain terdiri dari gambaran atau alur dari sistem yang akan dibuat. Dalam tahapan design berisi perancangan arsitektur sistem, *flowchart*, dan desain antarmuka.

a. Perancangan Arsitektur sistem

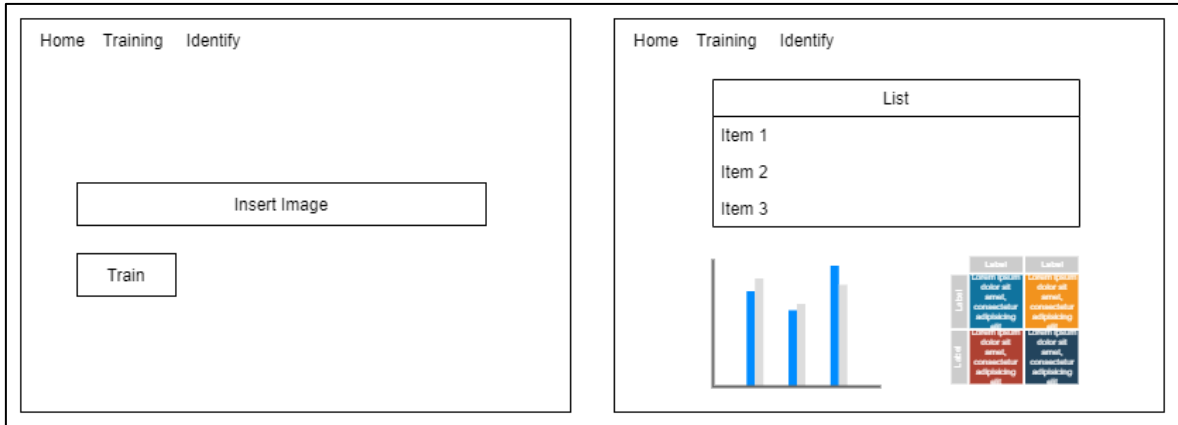
Berikut ini merupakan gambaran keseluruhan proses yang berjalan dari sistem identifikasi kematangan padi. Di dalam gambar terlihat bahwa dalam perancangan arsitektur sistem ini memiliki sifat *single user*, dimana *user* akan menginputkan data citra padi kedalam sistem deteksi kematangan padi. Setelah data citra padi tersebut masuk ke dalam sistem, sistem deteksi padi akan memproses citra padi tersebut dengan tahapan *preprocessing*, segmentasi dan ekstraksi ciri secara berkala sehingga dapat menghasilkan output deteksi kematangan padi yang akurat.



Gambar 3.12 Perancangan arsitektur sistem.

b. Perancangan Antar Muka

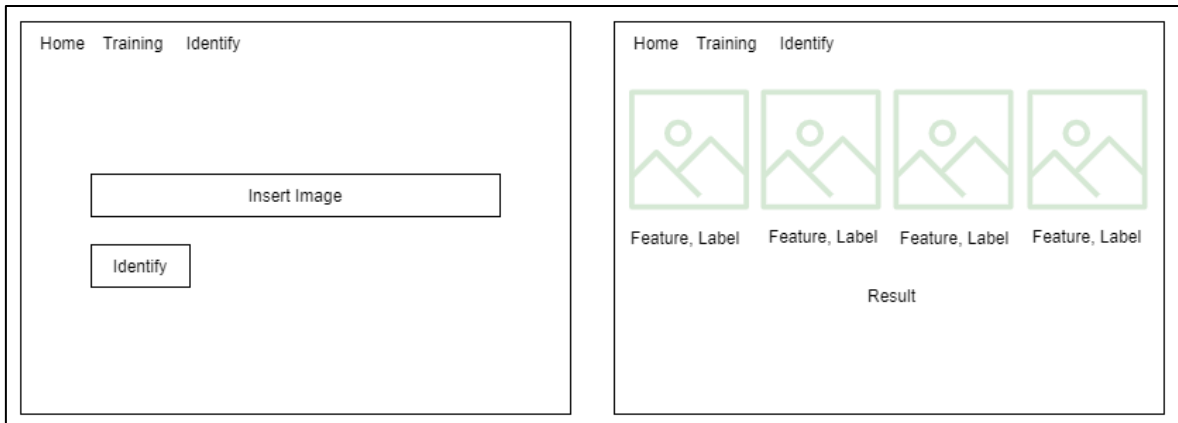
Perancangan antarmuka terkait erat dengan tampilan untuk menghubungkan pengguna ke sistem, dengan tujuan mengkomunikasikan fitur-fitur sistem sehingga dapat digunakan dengan maksimal. Dalam antar muka pelatihan dibutuhkan input berupa path, dimana path tersebut merupakan lokasi folder data yang akan dilatih. Selanjutnya untuk *output* yang akan dihasilkan terdiri dari tabel yang berisi nama file dari masing-masing citra, path dimana citra tersebut berada, dan kelas, *chart* untuk menampilkan hasil akurasi, presisi, dan recall dari setiap pelatihan data, serta *confusion matrix* dari pelatihan data tersebut. Berikut adalah perancangan antar muka dalam penelitian ini



Gambar 3.13 Perancangan antar muka bagian pelatihan.

Gambar 3.13 diatas merupakan halaman *training* (pelatihan). Untuk memulai proses training diawali dengan melakukan *input path* folder yang akan dilatih. Selanjutnya sistem akan melatih isi path folder tersebut. Apabila telah selesai, akan ditampilkan halaman yang berisi data citra apa saja yang dilatih dan masuk kedalam kelas apa serta akurasi, presisi dan *recall* yang ditampilkan ke dalam bentuk *chart*.

Selanjutnya, untuk halaman antar muka identifikasi dibutuhkan input berupa citra padi yang akan diidentifikasi, dan untuk *output* yang dihasilkan adalah citra hasil dari masing-masing tahap *preprocessing*, fitur, label dan hasil identifikasi.



Gambar 3.14 Perancangan antar muka bagian identifikasi.

Dari Gambar 3.14 Diatas merupakan rancangan untuk antar muka pada halaman *identify* yang bertujuan untuk mengidentifikasi citra padi dengan menggunakan model yang telah dilatih sebelumnya. Proses identifikasi dimulai dengan melakukan *insert* citra padi yang akan diuji. Setelah itu tekan klik tombol *identify* untuk melakukan identifikasi terhadap citra padi yang telah diinput. Selanjutnya, hasil yang akan ditampilkan berupa gambar dan hasil identifikasi citra padi tersebut masuk kedalam kelas matang ataupun mentah.

3.2.4 Construction (Code & Test)

Tahapan ini melakukan tahapan realisasi model menggunakan bahasa pemrograman tertentu. Dalam proses pengujian sistem dilakukan pembagian terhadap dataset yang berjumlah 800 data menggunakan *train test split* dengan beberapa pembagian data, yakni 70% : 30%, 80% : 20%, dan 90% : 10%. Proses pengujian yang digunakan dalam

penelitian ini adalah *confusion matrix*. Berikut merupakan rancangan pengujian yang dilakukan pada penelitian ini.

Tabel 3.9 Perancangan Pengujian

	Prediksi	
Aktual	Mentah	Matang
Mentah		
Matang		
Total		

BAB IV HASIL DAN PEMBAHASAN

4.1 Hasil dan Pembahasan

Pada bab ini akan membahas terkait hasil rancangan yang telah dibuat pada bab sebelumnya serta hasil sistem berdasarkan analisis dan rancangan yang telah dibangun. Dalam penelitian ini metode pengujian *confusion matrix* digunakan. Dari pengujian tersebut diharapkan mampu mengetahui akurasi serta kekurangan dari penelitian yang sudah dilaksanakan sehingga bisa digunakan untuk masukan ataupun pengembangan bagi penelitian selanjutnya. Berikut adalah hasil tampilan halaman awal yang terdapat dalam penelitian ini.



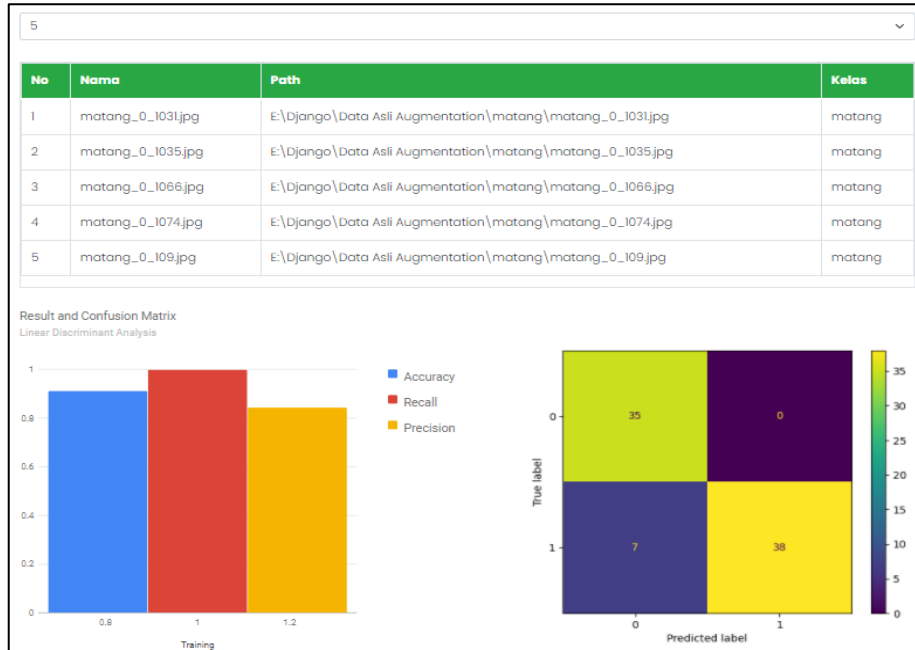
Gambar 4.1 Tampilan antarmuka halaman awal.

Dalam gambar 4.1 dapat dilihat apabila terdapat beberapa menu pada *navbar*, yaitu *home*, *training*, dan *identify*. Menu *home* pada *navbar* memiliki fungsi untuk kembali ke halaman awal. Sementara menu *training* merupakan menu dimana proses *training* serta *testing* berlangsung. Dan yang terakhir merupakan menu *identify*, dimana dalam menu ini *user* dapat melakukan identifikasi terhadap citra padi. Berikut adalah antarmuka halaman awal *training* dan halaman hasil *training*.



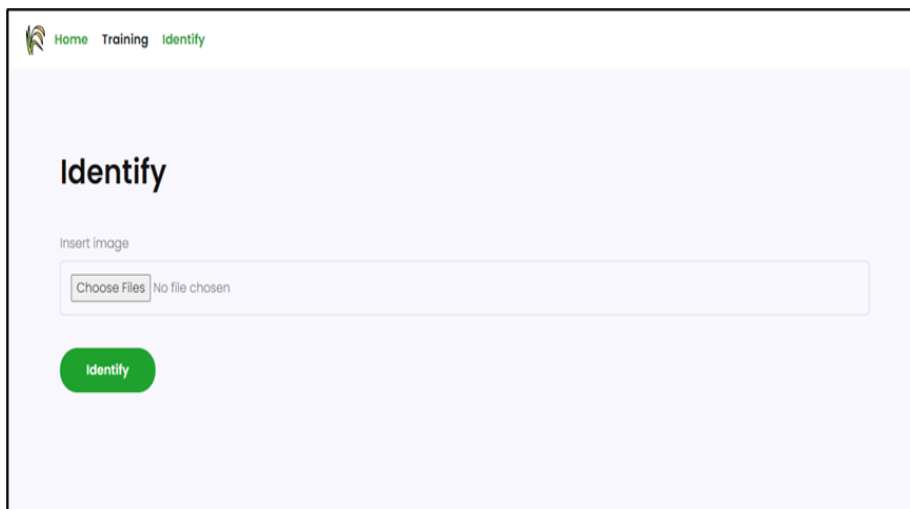
Gambar 4.2 Antarmuka awal halaman *training*

Dari halaman antarmuka di atas dapat dilihat jika dalam sistem ini membutuhkan path dari dataset yang akan *training*. Setelah melakukan *input path* folder dimana dataset yang akan *training* dan klik *button train* maka sistem akan meneruskan ke proses selanjutnya, yaitu *training*. Dalam penelitian ini sistem membutuhkan waktu kurang lebih 5 menit untuk menyelesaikan proses *training*. Apabila proses *training* telah selesai maka akan ditampilkan halaman seperti berikut



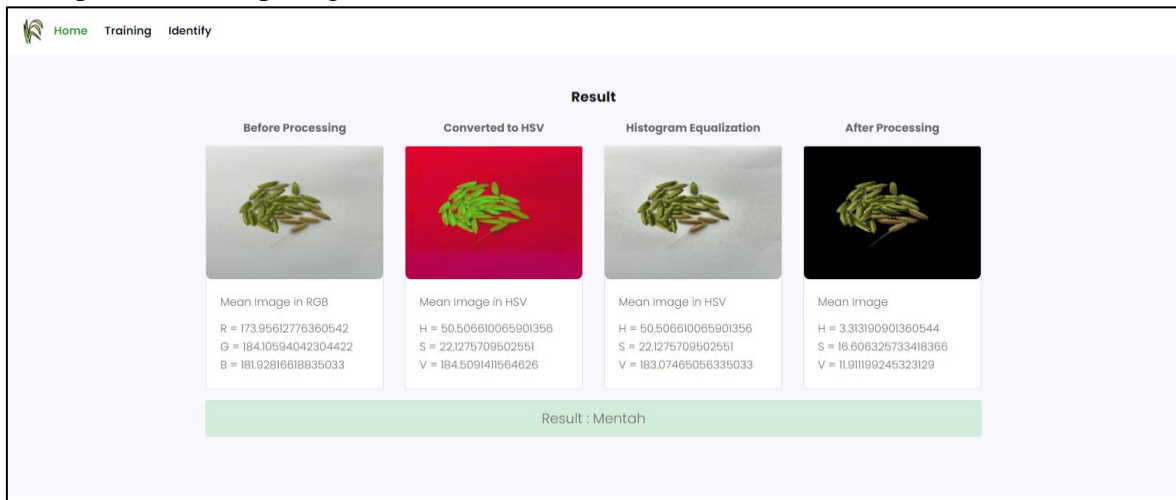
Gambar 4.3 Tampilan antarmuka halaman hasil *training*.

Dari gambar di atas dapat dilihat jika halaman *training* menampilkan tabel yang berisi nama *file*, *path* dan kelas dari masing-masing citra yang *training* serta *chart* yang berisi akurasi dari model yang *training* yang dilengkapi dengan *confusion matrix* dari data *test*. Hasil *training* tersebut nantinya akan di *export* menjadi sebuah model yang disimpan menjadi *best_model.sav* yang akan digunakan untuk melakukan identifikasi citra. Model dalam penelitian ini isinya akan tergantikan jika *user* melakukan *training* ulang. Namun, jika tidak melakukan *training* ulang maka hasil *training* paling terakhir yang digunakan oleh sistem. Berikut merupakan halaman untuk melakukan identifikasi padi.



Gambar 4.4 Tampilan antarmuka halaman *identify*.

Setelah melakukan *input* citra padi yang akan diidentifikasi selanjutnya akan ditampilkan hasil seperti gambar berikut.



Gambar 4.5 Hasil identifikasi.

Dalam halaman identifikasi dalam penelitian ini berisi tahapan *preprocessing*, ekstraksi fitur dan hasil identifikasi menggunakan model yang telah di *training* sebelumnya.

4.1.1 Preprocessing

Tahapan awal sebelum sistem melakukan identifikasi citra padi adalah *preprocessing*. Selain sebagai tahap awal, *preprocessing* dalam penelitian ini juga bertujuan untuk menyiapkan data supaya dapat memberikan hasil yang lebih baik saat diidentifikasi. Tahapan *preprocessing* yang dilakukan penelitian ini, yaitu: augmentasi yang bertujuan untuk memperbanyak variasi data, *resize* yang bertujuan untuk memperkecil resolusi citra, konversi RGB ke HSV yang bertujuan untuk mempersiapkan tahapan segmentasi, *histogram equalization* yang bertujuan untuk meratakan histogram pada citra agar tidak terlalu gelap, serta yang terakhir adalah segmentasi yang bertujuan untuk memisahkan antara objek dan latar belakang supaya lebih mudah mengidentifikasi objek. Akan tetapi untuk tahapan augmentasi hanya dilakukan satu kali pada seluruh data. Dimana tahapan augmentasi yang dilakukan dalam penelitian ini diantaranya adalah rotasi, shearing, dan zoom. Berikut adalah *pseudocode* untuk tahap augmentasi.

Modul 4.1 Augmentasi Data

```

INPUT :
    path dataset
OUTPUT :
    augmented dataset
path ← 'E:\Django\Data Asli Augmentation'
dir ← r"{}".format(path)
categories ← [matang, mentah]
valid_images ← [".jpg", ".jpeg", ".HEIC"]

datagen = CALL ImageDataGenerator with
    rotation_range=40,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True
for category in categories:
    path ← os.path.join(dir, category)
    
```

```

        label ← categories.index(category)
for img in os.listdir(path):
    imgpath ← os.path.join(path, img)
    img ← load_img(imgpath)
    x ← img_to_array(img)
    x ← x.reshape((1,) + x.shape)
    i ← 0
    for batch in datagen.flow(x, batch_size=1,
        save_to_dir="E:\\Django\\Data Asli
Augmentation\\{category}.format(category), save_prefix=category,
save_format='jpg'):
        i += 1
        if i > 20:
            break

```

Dari Modul 4.1 dapat dilihat merupakan implementasi dari augmentasi data. Augmentasi data yang dilakukan pada penelitian ini bertujuan untuk menghasilkan data yang lebih beragam. Augmentasi dilakukan menggunakan ImageDataGenerator yang disediakan oleh *tensorflow* dengan melakukan *rotation*, *zoom*, *shear*, *flip* pada masing-masing citra dalam setiap kategori. Selanjutnya, setelah melalui tahap augmentasi akan dilanjutkan dengan tahap *preprocessing* yang ditampilkan dalam *pseudocode* dibawah ini.

Modul 4.2 Preprocessing

```

INPUT :
    path ← request.POST['path']
OUTPUT :
    image
FUNCTION imageprocessing(frame):
    frame ← cv2.resize(frame, (336, 448))
    rgb ← cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
    hsv ← cv2.cvtColor(rgb, cv2.COLOR_RGB2HSV)
    H, S, V ← cv2.split(hsv)

    clahe ← cv2.createCLAHE(clipLimit=1.0)
    equalized_V ← clahe.apply(V)
    equalized ← cv2.merge([H, S, equalized_V])

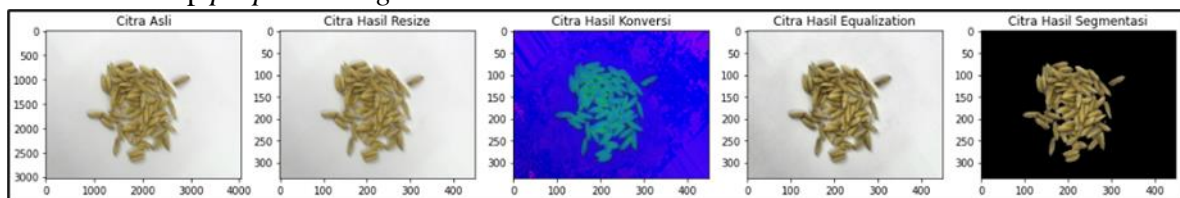
    lower_green ← np.array([16, 43, 40])
    upper_green ← np.array([179, 255, 255])
    lower_brown ← np.array([4, 88, 43])
    upper_brown ← np.array([9, 243, 255])

    mask ← cv2.inRange(equalized, lower_green, upper_green)
    mask2 ← cv2.inRange(equalized, lower_brown, upper_brown)
    final_mask ← mask + mask2
    image ← cv2.bitwise_and(equalized, equalized, mask=final_mask)
    return image
ENDFUNCTION

```

Dari modul 4.2 diatas dapat dilihat implementasi dari *flowchart* 3.2. Tahapan *preprocessing* bertujuan untuk mengurangi kekurangan pada citra, *preprocessing* yang digunakan adalah *resize* yang bertujuan untuk memperkecil resolusi citra sehingga akan menghasilkan ukuran yang lebih kecil agar mempercepat pemrosesan, konversi rgb ke hsv yang bertujuan untuk mempersiapkan pada tahapan selanjutnya dan ruang warna hsv dinilai tidak berubah dan akan memudahkan proses identifikasi (Pereira et al., 2018).

Setelah proses konversi rgb ke hsv selanjutnya akan dilanjutkan dengan *histogram equalization* yang bertujuan untuk meratakan *histogram* dari setiap citra agar menghasilkan citra yang tidak terlalu gelap ataupun terlalu terang. Dalam penelitian ini *histogram equalization* dilakukan pada value dari HSV. Citra yang telah dikonversi menjadi HSV selanjutnya dipisahkan menjadi *Hue*, *Saturation* and *Value*. *Value* atau nilai tersebut selanjutnya akan digunakan dalam perataan histogram. *Contrast Limited Adaptive Histogram Equalization* (CLAHE) digunakan untuk meratakan histogram pada penelitian ini. Setelah value citra diratakan, digabungkan kembali antara *Hue*, *Saturation*, dan *Value* sebelumnya dipisahkan. Citra yang sudah diratakan histogramnya selanjutnya akan memasuki tahapan segmentasi. Tahap segmentasi bertujuan untuk memisahkan antara objek dan latar belakang (*background*) sehingga proses identifikasi akan lebih mudah (Sutama et al., 2018). Segmentasi dalam penelitian ini dilakukan dengan menggunakan batas-batas nilai. Batas-batas nilai tersebut didapatkan dari proses *trial* dan *error*. Dengan nilai *lower green* = ([16, 43, 40]), *upper green* = ([179, 255, 255]), *lower brown* = ([4, 88, 43]), dan *upper brown* = ([9, 243, 255]). Apabila citra tersebut berada diluar *range* batas-batas nilai yang telah ditentukan maka akan menjadi hitam. Berikut merupakan contoh hasil dari tahap *preprocessing*.



Gambar 4.6 Hasil *preprocessing*.

4.1.2 Ekstraksi Fitur

Setelah melalui tahapan *preprocessing* selanjutnya citra tersebut akan diekstraksi untuk mendapatkan fitur-fitur yang akan digunakan untuk mengidentifikasi tingkat kematangan padi. Fitur yang digunakan adalah rata-rata nilai dari citra tersebut. Berikut adalah *pseudo-code* dari tahapan ekstraksi fitur.

Modul 4.3 Ekstraksi Fitur

```

INPUT :
  image
OUTPUT :
  fitur
feature ← cv2.mean(image)[:3]
data.append([feature, label])
end

```

Dari modul 4.3 diatas merupakan implementasi dari *flowchart* gambar 3.8. Dimana fitur yang digunakan adalah hasil rata-rata dari masing masing citra dan label yang kemudian dijadikan menjadi satu variabel, yaitu data. Berikut adalah hasil ekstraksi fitur yang telah diekspor ke dalam bentuk csv.

No	Nilai rata-rata	Label
0	(1.4185666454081631, 7.770368303571428, 7.484089338860544)	0
1	(1.4265518707482994, 6.569455516581632, 7.724024766156463)	0
2	(2.4304049744897958, 12.934085352891156, 14.667962106717686)	0
3	(1.760104432397959, 10.295592846513605, 9.827214870323129)	0
4	(1.6101323341836735, 8.534432132227892, 9.117493090986395)	0
5	(2.1116337159863945, 12.44127338435374, 11.930544483418366)	0
6	(2.240201158588435, 12.07849038052721, 13.611301551870747)	0
7	(1.7555072810374148, 9.45948262117347, 10.511326796343537)	0
8	(2.10726243622449, 10.326915922619047, 11.265711362670068)	0
9	(1.3103010735544218, 7.207801870748299, 7.861740008503401)	0
10	(1.4375996492346939, 7.7999375531462585, 8.390186543367347)	0
11	(1.6457403273809523, 8.056574192176871, 9.194209715136054)	0
12	(1.5901626275510203, 8.727240114795919, 8.471865699404761)	0
13	(1.2375903486394557, 7.245721726190476, 7.015379198554421)	0
14	(1.2611540710034013, 6.9991363732993195, 7.640292835884353)	0
15	(2.0704121492346936, 11.266269398384352, 12.479339392006802)	0
16	(2.9269836840986394, 15.538364955357142, 17.529290231717688)	0
17	(1.7541985544217686, 10.241270727040815, 9.59504544005102)	0
18	(2.7291865965136055, 14.314413265306122, 14.208585778061224)	0
19	(1.752132493622449, 9.542902317176871, 10.42153619260204)	0
20	(1.7437553146258502, 9.519963063350339, 10.518880208333332)	0
21	(1.800150138180272, 8.906754889455781, 9.610916241496598)	0
22	(1.3870509141156462, 7.6153273809523805, 8.278625903486395)	0

Gambar 4.7 Hasil ekstraksi fitur.

4.1.3 Linear Discriminant Analysis

Tahapan selanjutnya setelah citra melalui ekstraksi fitur adalah tahap identifikasi menggunakan metode *Linear Discriminant Analysis*. Berikut adalah tahapan identifikasi dengan metode *Linear Discriminant Analysis*.

Modul 4.4 Linear Discriminant Analysis
<pre> INPUT : best_model : model hasil dari proses training OUTPUT : Write mentah/ matang clf ← LinearDiscriminantAnalysis() clf.fit(trainX, trainY) prediction ← clf.predict(testX) prediction ← CALL predict with best_model IF prediction = 0 THEN Write matang ELSE Write mentah ENDIF </pre>

Modul 4.4 diatas merupakan penerapan dari *flowchart* 3.9 yang digunakan dalam proses identifikasi dalam sistem. Selanjutnya adalah mengambil fitur dari masing-masing citra sebagai X_k yang selanjutnya akan dimasukkan ke dalam rumus f_i . Setelah diperoleh nilai f_i untuk dibandingkan dengan nilai f_1 dan f_2 . Jika nilai $f_1 > f_2$ maka termasuk kelas matang dan jika $f_1 < f_2$ maka termasuk kedalam kelas mentah. Selanjutnya dilakukan *preprocessing* terhadap objek tersebut. Setelah melalui *preprocessing* selanjutnya mencari nilai rata-rata dari objek tersebut yang digunakan sebagai fitur citra tersebut. Selanjutnya fitur tersebut akan dipredict dengan menggunakan model yang dihasilkan dari proses

training sebelumnya. Apabila prediksi menghasilkan nilai 0 maka matang, dan nilai 1 maka mentah.

4.2 Pengujian

Tahapan pengujian yang dilakukan untuk melihat akurasi model yang dihasilkan. Pengujian dilakukan dengan melihat nilai akurasi yang dihasilkan. Nilai akurasi tersebut dihitung dari nilai yang terdapat pada *confusion matrix* meliputi nilai *True Positive (TP)*, *True Negative (TN)*, *False Positive (FP)*, dan *False Negative (FN)*.

Confusion matrix digunakan untuk menguji dataset yang telah dikumpulkan dengan beberapa perbandingan data *training* dan data *testing*, yakni 70% : 30%, 80% : 20%, dan 90% : 10%. Untuk menampilkan hasil pengujian yang dilakukan pada penelitian ini menggunakan *confusion matrix* yang disediakan sklearn. Berikut adalah modul yang digunakan untuk menampilkan *confusion matrix* dari sklearn.

```

Modul 4.5 Confusion Matrix
INPUT :
    trainX, trainY, testX, testY
OUTPUT :
    TP, TN, FP, FN

acc ← CALL accuracy_score with testY and prediction
pre ← CALL precision_score with testY and prediction
rec ← CALL recall_score with testY and prediction
cm ← CALL confusion_matrix with testY and prediction
write ('\n', classification_report, testY, prediction)

ConfusionMatrixDisplay.from_estimator(clf, testX, testY)
plt.savefig('static\plot\confusion.png')
tn, fp, fn, tp = confusion_matrix(
    list(testY), list(prediction), labels=clf.classes_.ravel())
write('True Positive', tp)
write('True Negative', tn)
write('False Positive', fp)
write('False Negative', fn)

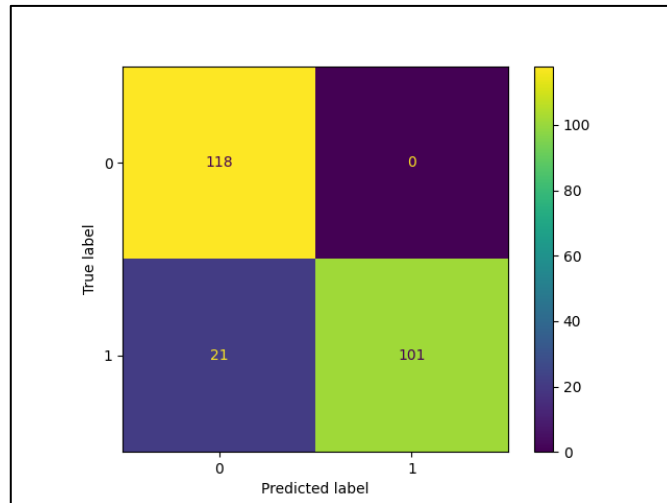
```

Dari modul 4.5 diatas dapat dilihat untuk menampilkan *confusion matrix* dari *sklearn* dalam penelitian ini. Berikut adalah tabel hasil confusion matrix dengan pembagian data *training* dan data *testing* sebesar 70% : 30% pada penelitian ini dapat dilihat pada tabel berikut.

Tabel 4.1 Tabel pengujian pembagian data *training* dan data *testing* 70 % : 30%

Aktual	Prediksi		Total
	Matang	Mentah	
Matang	101	21	122
Mentah	0	118	118
Total	101	139	240

Berikut adalah *confusion matrix* yang dihasilkan dari *matplotlib* untuk pengujian dengan data *training* dan data *testing* 70% : 30%.



Gambar 4.8 Confusion matrix 70% : 30%

Keterangan :

True Positive (TP) = 101

True Negative (TN) = 118

False Positive (FP) = 0

False Negative (FN) = 21

Dari nilai TP, TN, FP, dan FN di atas selanjutnya nilai *accuracy*, *precision*, dan *recall* melakukan perhitungan seperti pada persamaan 3..

$$\begin{aligned}
 accuracy &= \frac{101+118}{101+0+21+118} \times 100\% \\
 &= \frac{219}{240} \times 100\% \\
 &= 91\%
 \end{aligned}$$

$$\begin{aligned}
 Precision &= \frac{101}{101+0} \times 100\% \\
 &= \frac{101}{101} \times 100\% \\
 &= 100\%
 \end{aligned}$$

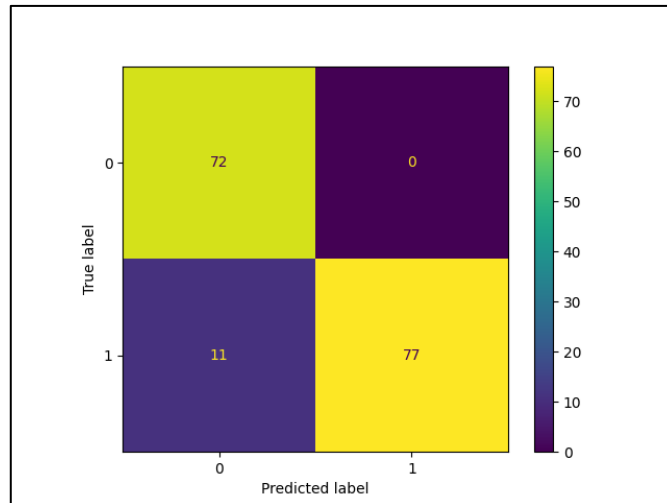
$$\begin{aligned}
 Recall &= \frac{101}{101+21} \times 100\% \\
 &= \frac{101}{122} \times 100\% \\
 &= 83\%
 \end{aligned}$$

Selanjutnya tabel *confusion matrix* dengan pembagian data training dan data testing sebesar 80% : 20% pada penelitian ini dapat dilihat pada tabel berikut.

Tabel 4.2 Tabel pengujian pembagian data *training* dan data *testing* 80 % : 20%

Aktual	Prediksi		Total
	Matang	Mentah	
Matang	77	11	88
Mentah	0	72	72
Total	77	83	160

Berikut adalah *confusion matrix* yang dihasilkan dari *matplotlib* untuk pengujian dengan data *training* dan data *testing* 80% : 20%.



Gambar 4.9 Confusion matrix 80% : 20%

Keterangan :

True Positive (TP) = 77

True Negative (TN) = 72

False Positive (FP) = 0

False Negative (FN) = 11

Dari nilai TP, TN, FP, dan FN di atas selanjutnya nilai *accuracy*, *precision*, dan *recall* melakukan perhitungan seperti pada persamaan 3.

$$\begin{aligned}
 \text{accuracy} &= \frac{77+72}{77+0+11+72} \times 100\% \\
 &= \frac{149}{160} \times 100\% \\
 &= 93\%
 \end{aligned}$$

$$\begin{aligned}
 \text{Precision} &= \frac{77}{77+0} \times 100\% \\
 &= \frac{77}{77} \times 100\% \\
 &= 100\%
 \end{aligned}$$

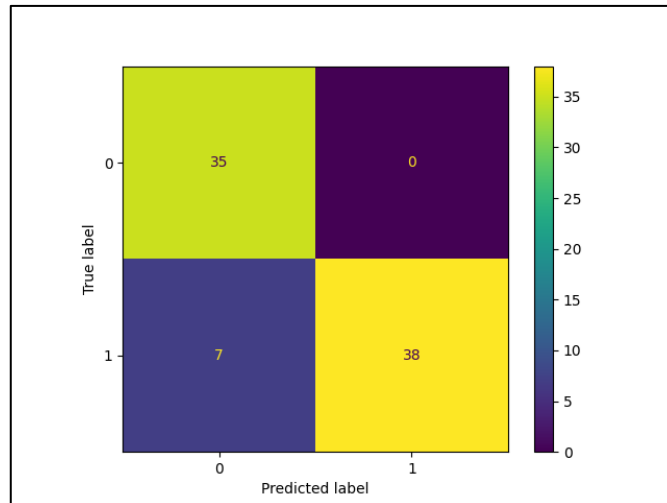
$$\begin{aligned}
 \text{Recall} &= \frac{77}{77+11} \times 100\% \\
 &= \frac{77}{88} \times 100\% \\
 &= 88\%
 \end{aligned}$$

Selanjutnya tabel hasil *confusion matrix* dengan pembagian data training dan data testing sebesar 90% : 10% pada penelitian ini dapat dilihat pada tabel berikut.

Tabel 4.3 Tabel pengujian pembagian data *training* dan data *testing* 90% : 10%

Aktual	Prediksi		Total
	Matang	Mentah	
Matang	38	7	45
Mentah	0	35	35
Total	38	42	80

Berikut adalah *confusion matrix* yang dihasilkan dari *matplotlib* untuk pengujian dengan data *training* dan data *testing* 90% : 10%.



Gambar 4.10 Confusion matrix 90% : 10%

Keterangan :

True Positive (TP) = 38

True Negative (TN) = 35

False Positive (FP) = 0

False Negative (FN) = 7

Dari nilai TP, TN, FP, dan FN di atas selanjutnya nilai *accuracy*, *precision*, dan *recall* melakukan perhitungan seperti pada persamaan 3.

$$\begin{aligned}
 \text{accuracy} &= \frac{38+35}{38+0+7+35} \times 100\% \\
 &= \frac{73}{80} \times 100\% \\
 &= 91\%
 \end{aligned}$$

$$\begin{aligned}
 \text{Precision} &= \frac{38}{38+0} \times 100\% \\
 &= \frac{38}{38} \times 100\% \\
 &= 100\%
 \end{aligned}$$

$$\begin{aligned}
 \text{Recall} &= \frac{38}{38+7} \times 100\% \\
 &= \frac{38}{45} \times 100\% \\
 &= 84\%
 \end{aligned}$$

Berikut adalah tabel perbandingan hasil akurasi dari masing-masing pembagian data training dan testing yang telah dihitung di atas.

Tabel 4.4 Tabel perbandingan akurasi.

No	Presentase	Akurasi
1.	70 : 30	91%
2.	80 : 20	93%
3.	90 : 10	91%

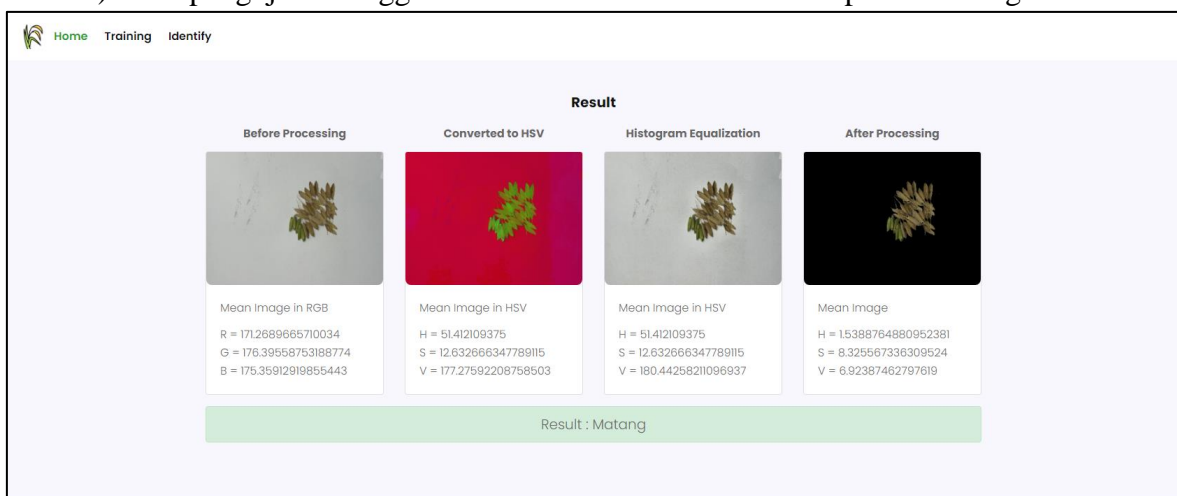
Dari tabel 4.5 di atas, dapat dilihat apabila nilai akurasi yang dihasilkan paling tinggi diperoleh pada pembagian data *training* dan data *testing* 80% : 20% sebesar 93%.

4.3 Pembahasan

Dari penelitian yang telah dibangun berdasarkan rancangan-rancangan pada bab sebelumnya untuk mengidentifikasi citra menggunakan metode *Linear Discriminant Analysis*. Data pada penelitian ini diperoleh secara langsung di Ambarawa dengan menggunakan kamera smartphone yang beresolusi 12 *megapixel* dengan bantuan studio

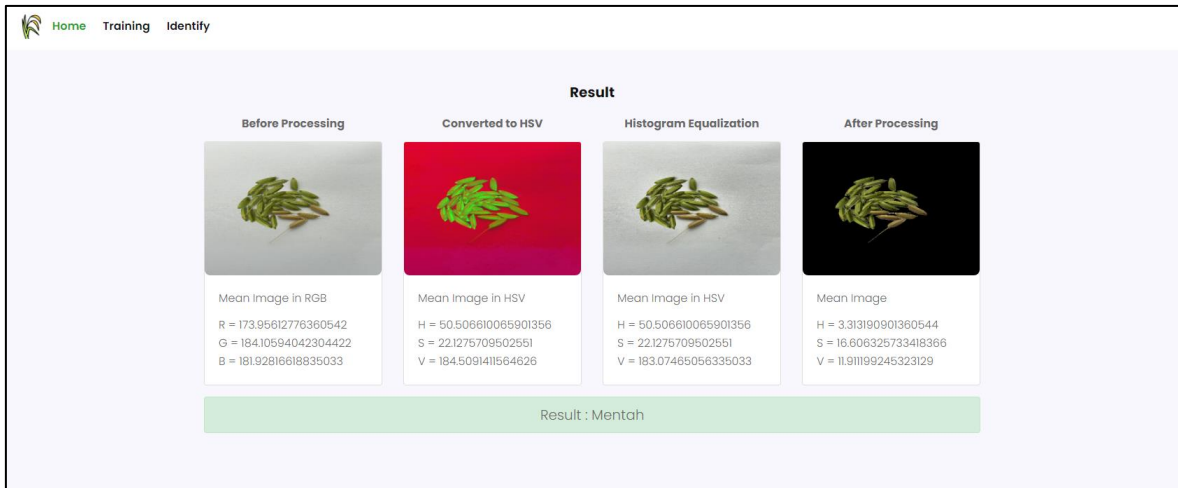
box dengan lampu phillips 20 watt dengan cahaya 1934 lux serta pengambilan jarak foto sebesar 15 cm. tersebut dipisahkan kelasnya (matang dan mentah) dengan bantuan ahli dibidangnya. Tahapan *preprocessing* yang dilakukan penelitian ini, yaitu: augmentasi yang bertujuan untuk memperbanyak variasi data, dimana dalam penelitian ini augmentasi yang digunakan adalah *zoom*, *shear*, *flip*, dan *rotation*. *Resize* yang bertujuan untuk memperkecil resolusi citra yang semula berukuran 4032 x 3024 piksel menjadi ukuran 432 x 324 piksel, konversi RGB ke HSV yang bertujuan untuk mempersiapkan tahapan segmentasi, *histogram equalization* yang bertujuan untuk meratakan histogram pada citra agar tidak terlalu gelap, serta yang terakhir adalah segmentasi berdasarkan nilai *lower* dan *upper* yang diperoleh dari proses *trian and error* menggunakan *slider* HSV yang bertujuan untuk memisahkan antara objek dan latar belakang supaya lebih mudah mengidentifikasi objek. Akan tetapi untuk tahapan augmentasi hanya dilakukan satu kali pada seluruh citra. Total keseluruhan data setelah di augmentasi sebanyak 800 citra dengan pembagian 400 citra kelas matang dan 400 citra kelas mentah.

Dari penelitian ini peneliti mencoba melakukan identifikasi untuk mengukur seberapa akurat sistem ini dalam mengidentifikasi padi. Dimana data yang digunakan adalah data berisi citra padi gabungan antara mentah dan matang dengan jumlah bulir yang sama (20 bulir padi matang dan 20 bulir padi mentah), citra padi gabungan matang dan mentah dimana bulir padi matang lebih banyak dibandingkan dengan mentah (35 bulir padi matang dan 5 bulir padi mentah), dan citra gabungan matang dan mentah dimana bulir padi mentah lebih banyak dibandingkan dengan matang (5 bulir padi matang dan 35 bulir padi mentah). Dari pengujian menggunakan data *validation* tersebut diperoleh sebagai berikut.



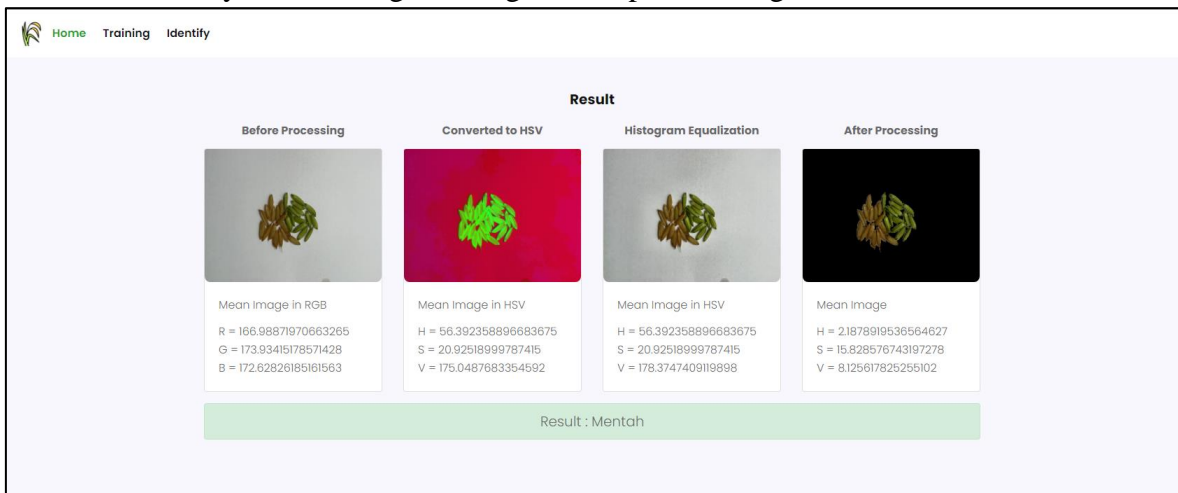
Gambar 4.11 Gabungan matang dan mentah dimana bulir padi matang lebih banyak dibandingkan dengan mentah

Dari gambar di atas yang merupakan hasil identifikasi menggunakan bulir padi gabungan matang dan mentah namun bulir padi matang lebih banyak, yakni 35 bulir padi matang dan 5 bulir padi mentah dari sawah yang sama dengan data *training* dan data *test* yang sebelumnya. Dari hasil identifikasi menggunakan sistem yang telah dibangun diperoleh hasil jika citra tersebut masuk kedalam kelas matang. Hal ini terjadi karena sistem mendeteksi berdasarkan warna, dimana dalam dalam citra tersebut warna kuning keemasan atau matang lebih dominan dibandingkan dengan warna hijau. Sehingga, hasil yang didapatkan untuk identifikasi citra gabungan antara bulir padi matang dan mentah namun bulir padi matang lebih banyak dibandingkan bulir padi mentah adalah matang.

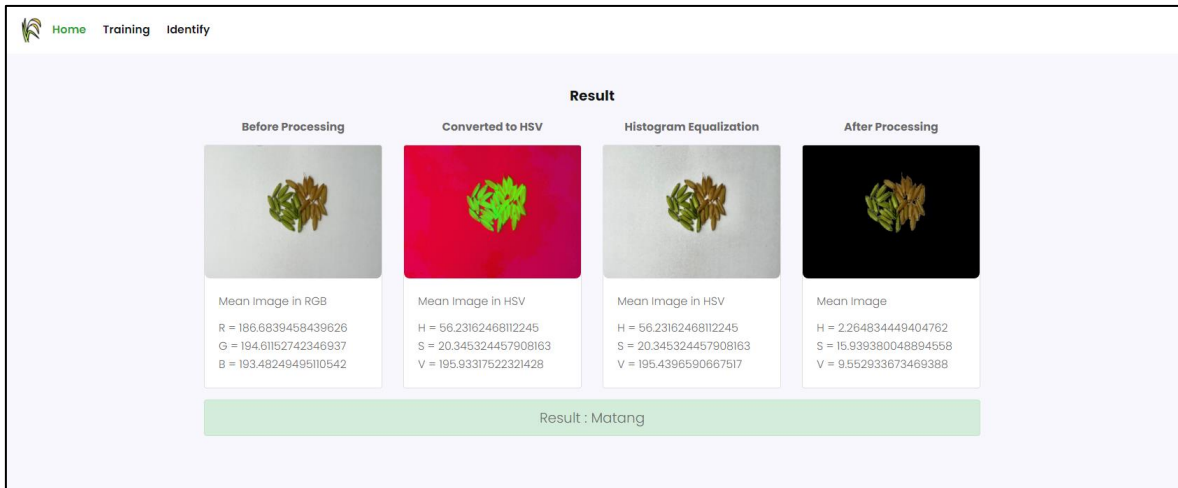


Gambar 4.12 Gabungan matang dan mentah dimana bulir padi mentah lebih banyak dibandingkan dengan matang

Dari gambar diatas yang merupakan hasil identifikasi menggunakan bulir padi gabungan matang dan mentah namun bulir padi mentah lebih banyak dibandingkan dengan matang, yakni 35 bulir padi mentah dan 5 bulir padi matang. Hasil identifikasi citra tersebut menggunakan sistem mendapatkan hasil kelas mentah dan diidentifikasi oleh sistem sebagai kelas mentah. Hal tersebut terjadi karena sistem mengidentifikasi kematangan padi berdasarkan warna, dimana dalam citra tersebut warna hijau atau padi yang mentah lebih dominan dibandingkan dengan warna kuning keemasan atau matang sehingga hasil identifikasi dari citra gabungan matang dan mentah dimana bulir padi mentah lebih banyak dibandingkan dengan bulir padi matang adalah mentah.



Gambar 4.13 Gabungan matang dan mentah dengan jumlah yang sama



Gambar 4.14 Gabungan matang dan mentah dengan jumlah yang sama.

Dari gambar 4.13 dan 4.14 merupakan hasil identifikasi dengan menggunakan citra dengan jumlah bulir padi yang sama, yakni 20 bulir padi matang dan 20 bulir padi mentah. Dalam percobaan ini dilakukan 10 kali identifikasi menggunakan sistem dan dari percobaan tersebut sistem mengidentifikasi citra tersebut ke dalam kelas matang dan kelas mentah. Berikut adalah tabel hasil percobaan yang telah dilakukan.

Tabel 4.5 Tabel percobaan identifikasi.

Percobaan ke-	Hasil
1	Matang
2	Matang
3	Mentah
4	Matang
5	Mentah
6	Mentah
7	Matang
8	Mentah
9	Matang
10	Mentah

Dari tabel di atas dapat dilihat jika sistem citra gabungan bulir padi matang dan mentah dengan jumlah yang sama dengan kelas matang dan mentah. Hal tersebut terjadi karena sistem mendeteksi berdasarkan warna, dimana dalam citra tersebut warna kuning keemasan (matang) dan warna hijau (mentah) sama-sama dominan. Sehingga hasil identifikasi yang diperoleh akan menunjukkan kelas matang dan kelas mentah

Dari sistem ini diperoleh nilai akurasi masing-masing dari pembagian data training dan data testing 70% : 30% dengan akurasi sebesar 91%, pembagian data training dan data testing 80% : 20% dengan akurasi sebesar 93%, dan pembagian data *training* dan data *testing* dengan akurasi sebesar 91%. Sehingga, pembagian data *training* dan data *testing* yang menghasilkan akurasi tertinggi adalah pembagian data *training* dan data *testing* 80% : 20% sebesar 93%.

BAB V PENUTUP

5.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan yang bertujuan mengidentifikasi kematangan padi menggunakan metode *Linear Discriminant Analysis* dapat ditarik kesimpulan dari hasil analisis, perancangan sistem dan pembahasan yang telah dilakukan, yakni agar dapat membangun sistem yang mampu mengidentifikasi kematangan padi dibutuhkan dataset bulir padi yang akan digunakan dalam proses pelatihan. Untuk menguji akurasi dari model *Linear Discriminant Analysis* tersebut dilakukan beberapa pembagian data *training* dan data *testing*, yakni 70% : 30%, 80% : 20% dan 90% : 10%. Dari masing-masing pengujian dengan pembagian data tersebut diperoleh nilai akurasi paling tinggi pada pembagian data *training* dan data *testing* sebesar 80% : 20% dengan nilai akurasi sebesar 93% dengan pengujian yang dilakukan menggunakan data uji yang tersebar secara acak. Sehingga, dapat disimpulkan apabila metode *Linear Discriminant Analysis* mampu mengidentifikasi kematangan padi dengan baik.

5.2 Saran

Berdasarkan kesimpulan serta hasil analisis pada penelitian identifikasi tingkat kematangan padi menggunakan metode *Linear Discriminant Analysis* masih terdapat kekurangan serta keterbatasan yang kedepannya diharapkan bisa digunakan sebagai pertimbangan maupun acuan dalam penelitian selanjutnya. Adapun beberapa hal yang dapat digunakan sebagai bahan pertimbangan maupun acuan dalam pengembangan penelitian ini adalah sebagai berikut.

1. Pengembangan untuk penelitian selanjutnya dapat dilakukan dengan menggunakan menambahkan berbagai variasi padi.
2. Pengembangan dapat dilakukan pada tahap *preprocessing* citra yang bertujuan untuk mengurangi derau pada gambar.
3. Pengembangan dapat dilakukan pada tahap akuisisi citra agar dapat menghasilkan gambar dengan lebih sedikit derau.
4. Pengembangan dapat dilakukan dengan menambahkan deteksi tepi untuk menghasilkan citra yang tersegmentasi dengan baik.

DAFTAR PUSTAKA

- Ansari, A. (2022). Produksi beras juga bisa beradaptasi dengan perubahan iklim, syaratnya riset iklim harus diperbanyak. Retrieved from <https://theconversation.com/produksi-beras-juga-bisa-beradaptasi-dengan-perubahan-iklim-syaratnya-riset-iklim-harus-diperbanyak-175737>
- Azman, N., Khairunniza-Bejo, S., Ismail, W. I. W., & Wayayok, A. (2014). Estimating Maturity of Paddy Using RGB Colour Space. *Journal of Advanced Agricultural Technologies*, 1(2), 119–124. <https://doi.org/10.12720/joaat.1.2.119-124>
- Cahyani, S., Wiryasaputra, R., & Gustriansyah, R. (2018). Identifikasi Huruf Kapital Tulisan Tangan Menggunakan Linear Discriminant Analysis dan Euclidean Distance. *Jurnal Sistem Informasi Bisnis*, 8(1), 57. <https://doi.org/10.21456/vol8iss1pp57-67>
- Ghazali, Eko K. Subha, Galuh M., M. Burhannudin, M. I. N. (2018). Aplikasi Kematangan Tomat Berdasarkan Warna dengan Metode Linear Discriminant Analysis (LDA). <https://Docplayer.Info/31373581-Aplikasi-Kematangan-Tomat-Berdasarkan-Warna-Dengan-Metode-Linear-Discriminant-Analysis-Lda.Html>.
- Gonzalez, R. C., & Woods, R. E. (2018). *4TH EDITION Digital image processing*.
- Hanafi, M. H., Fadillah, N., & Insan, A. (2019). Optimasi Algoritma K-Nearest Neighbor untuk Klasifikasi Tingkat Kematangan Buah Alpukat Berdasarkan Warna. *It Journal Research and Development*, 4(1), 10–18. [https://doi.org/10.25299/itjrd.2019.vol4\(1\).2477](https://doi.org/10.25299/itjrd.2019.vol4(1).2477)
- Jumadi, J., Yupianti, Y., & Sartika, D. (2021). Pengolahan Citra Digital Untuk Identifikasi Objek Menggunakan Metode Hierarchical Agglomerative Clustering. *JST (Jurnal Sains Dan Teknologi)*, 10(2), 148–156. <https://doi.org/10.23887/jstundiksha.v10i2.33636>
- Mikołajczyk, A., & Grochowski, M. (2019). 2019 International Interdisciplinary PhD Workshop, IPhDW 2019. *2019 International Interdisciplinary PhD Workshop, IPhDW 2019*, 117–122.
- Munantri, N. Z., Sofyan, H., & Florestiyanto, M. Y. (2020). Aplikasi Pengolahan Citra Digital Untuk Identifikasi Umur Pohon. *Telematika*, 16(2), 97. <https://doi.org/10.31315/telematika.v16i2.3183>
- Munir, R. (2019). *Digitalisasi Citra*. Retrieved from <https://informatika.stei.itb.ac.id/~rinaldi.munir/Citra/2019-2020/04-Digitalisasi-citra.pdf>
- Mustikasucy, Y. (2020). PANEN DAN PENANGANAN PASCA PANEN TANAMAN PADI. Retrieved from <http://cybex.pertanian.go.id/artikel/92444/panen-dan-penanganan-pasca-panen-tanaman-padi/>
- Pereira, S., Luiz, F., Barbon, S., Valous, N. A., & Barbin, D. F. (2018). Predicting the ripening of papaya fruit with digital imaging and random forests. *Computers and*

Electronics in Agriculture, 145(August 2017), 76–82.
<https://doi.org/10.1016/j.compag.2017.12.029>

- Ratnasari, M. C. D. (2018). Deep Learning Convolutional Neural Network Untuk Klasifikasi Pengenalan Objek Menggunakan MxNet (Studi Kasus : Data Citra Motif Batik Keraton dan Pesisir). Universitas Islam Indonesia.
- Anami., B., M., N., & Hanamarati., N. (2017). Bulk Paddy Grain Ageing Period Classification using RGB and HSI Color Features. *International Journal of Computer Applications*, 176(5), 33–43. <https://doi.org/10.5120/ijca2017915577>
- Shambulinga, M., & Sadashivappa, G. (2020). *Supervised Hyperspectral Image Classification using SVM and Linear Discriminant Analysis*. 11(10), 403–409.
- Sulistyo, I. A. (2017). *SISTEM DETEKSI PANEN PADI BERDASAR WARNA DAUN MENGGUNAKAN FUZZY C-MEANS CLUSTERING*.
- Surmaini, E., & Syahbuddin, H. (2016). Kriteria Awal Musim Tanam: Tinjauan Prediksi Waktu Tanam Padi Di Indonesia. *Jurnal Penelitian Dan Pengembangan Pertanian*, 35(2), 47. <https://doi.org/10.21082/jp3.v35n2.2016.p47-56>
- Sutama, V. M., Magdalena, I. R., & Wijayanto, I. (2018). *Identifikasi Objek Dominan Citra Menggunakan Metode Markov Random Field*. 5(3), 4859–4865.
- Wang, R., Han, F., & Wu, W. (2021). Estimation of paddy rice maturity using digital imaging. *International Journal of Food Properties*, 24(1), 1403–1415. <https://doi.org/10.1080/10942912.2021.1970581>