**PROPOSAL**

**IMPLEMENTASI METODE *MULTI SUPPORT VECTOR MACHINE***

**UNTUK MENENTUKAN TINGKAT KEMATANGAN BUAH**

**PISANG RAJA DAN PISANG KEPOK**

Diajukan untuk memenuhi salah satu syarat memperoleh derajat Sarjana Teknik



**NINA SULARIDA E1E1 15 036**

**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS HALU OLEO**

**KENDARI**

**2020**

# DAFTAR ISI

**HALAMAN SAMPUL** ................................................................................... i

**HALAMAN PENGESAHAN** ........................................................................ ii

**HALAMAN PERNYATAAN** ........................................................................ iv

[**INTISARI** v](#_Toc202155)

[**ABSTRACT** vi](#_Toc202156)

[**KATA PENGANTAR** vii](#_Toc202157)

[**HALAMAN PERSEMBAHAN** x](#_Toc202158)

[**DAFTAR ISI** xiii](#_Toc202159)

[**DAFTAR TABEL** xviii](#_Toc202160)

[**DAFTAR GAMBAR** xx](#_Toc202161)

[**BAB I PENDAHULUAN** 1](#_Toc202162)

[1.1. Latar Belakang 1](#_Toc202163)

[1.2. Rumusan Masalah 3](#_Toc202164)

[1.3. Batasan Masalah 3](#_Toc202165)

[1.4. Tujuan Penelitian 3](#_Toc202166)

[1.5. Manfaat Penelitian 4](#_Toc202167)

[1.6. Sistematika Penulisan 4](#_Toc202168)

[1.7. Tinjauan Pustaka 5](#_Toc202169)

[**BAB II LANDASAN TEORI** 7](#_Toc202170)

[2.1. Pisang 7](#_Toc202171)

[2.2. Kriteria Kematangan Buah Pisang 7](#_Toc202172)

[2.2.1. Pisang Raja 8](#_Toc202173)

[2.2.1.1. Pisang Raja Mentah 8](#_Toc202174)

[2.2.1.2. Pisang Raja Cukup Matang 9](#_Toc202175)

[2.2.1.3. Pisang Raja Matang 9](#_Toc202176)

[2.2.1.4. Pisang Raja Matang Sekali 9](#_Toc202177)

[2.2.2. Pisang Kepok 10](#_Toc202178)

[2.2.2.1. Pisang Kepok Mentah 10](#_Toc202179)

[2.2.2.2. Pisang Kepok Cukup Matang 11](#_Toc202180)

[2.2.2.3. Pisang Kepok Matang 11](#_Toc202181)

[2.2.2.4. Pisang Kepok Matang Sekali 12](#_Toc202182)

[2.3. Citra Digital 12](#_Toc202183)

[2.4. Segmentasi 13](#_Toc202184)

[2.4.1. Konversi Citra RGB ke HSV 13](#_Toc202185)

[2.4.2. *Thresholding* 15](#_Toc202186)

[2.4.3. *Masking image* 16](#_Toc202187)

[2.4.4. *Blob Detection* 16](#_Toc202188)

[2.5. M-SVM (*Multi* *Support Vector Mechine*) 18](#_Toc202189)

[2.6. Python 21](#_Toc202190)

[2.7. *Flowchart* 22](#_Toc202191)

[2.8. *Unified Modelling Languange* (UML) 23](#_Toc202192)

[2.8.1. *Usecase* *Diagram* 24](#_Toc202193)

[2.8.2. *Sequence Diagram* 25](#_Toc202194)

[2.8.3. *Activity* *Diagram* 25](#_Toc202195)

[2.9. Metodologi Pengembangan Sistem (*Rational Unified Process*) 26](#_Toc202196)

[**BAB III METODOLOGI PENELITIAN** 28](#_Toc202197)

[3.1. Metode Pengumpulan Data 28](#_Toc202198)

[3.2. Metode Pengembangan Sistem 28](#_Toc202199)

[3.2.1. *Inception* 29](#_Toc202200)

[3.2.2. *Elaboration* 29](#_Toc202201)

[3.2.3. *Construction* 29](#_Toc202202)

[3.2.4. *Transition* 29](#_Toc202203)

[3.3. Waktu Penelitian 30](#_Toc202204)

[3.4. Analisis Permasalahan 31](#_Toc202206)

[3.5. Rancangan Sistem 31](#_Toc202207)

[3.6. Perancangan Proses Citra Latih 34](#_Toc202208)

[3.7. Perancangan Proses Citra Uji 35](#_Toc202209)

[3.7.1. *Resize* 35](#_Toc202210)

[3.7.2. Segmentasi Objek 35](#_Toc202211)

[3.7.3. Konversi citra RGB ke HSV 35](#_Toc202212)

[3.7.4. *Masking Image* 37](#_Toc202213)

[3.7.5. *Blob detection* 39](#_Toc202214)

[3.7.6. *Multi Support Vector Mechine* (M-SVM) 39](#_Toc202215)

[3.8. Perancangan *Unified Modeling Languange* (UML) 42](#_Toc202216)

[3.8.1. *Usecase Diagram* 42](#_Toc202217)

[3.8.2. *Activity Diagram* 43](#_Toc202218)

[3.8.2.1. *Activity* *Diagram* Masukkan Citra 43](#_Toc202219)

[3.8.2.2. *Activity* *Diagram* Ekstraksi Fitur 43](#_Toc202220)

[3.8.2.3. *Activity* *Diagram* Simpan Fitur 44](#_Toc202221)

[3.8.2.4. *Activity* *Diagram* Proses Uji 45](#_Toc202222)

[3.8.3. *Sequence Diagram* 45](#_Toc202223)

[3.8.3.1. *Sequance* *Diagram* Masukkan Citra 45](#_Toc202224)

[3.8.3.2. *Sequence Diagram* Ektraksi Fitur 46](#_Toc202225)

[3.8.3.3. S*equance* *Diagram* Simpan Fitur 46](#_Toc202226)

[3.7. Rancangan Penyimpanan Data 47](#_Toc202227)

[3.8. Perancangan Antarmuka Menu Utama 47](#_Toc202228)

[3.9. Perancangan Antarmuka Menu Identifikasi 48](#_Toc202229)

[310. Perhitungan Akurasi 48](#_Toc202230)

**DAFTAR PUSTAKA**

# 

# DAFTAR TABEL

Tabel 2.1 Contoh SVM pendekatan *One Against All* ...................................... 21

Tabel 2.2 Simbol-simbol dalam *Flowchart* ..................................................... 23

Tabel 2.3 Simbol *Usecase* *Diagram* ................................................................ 24

Tabel 2.4 Simbol *Sequence* *Diagram*............................................................... 25

Tabel 2.5 Simbol *Activity* *Diagram* ................................................................. 26

Tabel 3.1. Waktu Penelitian ............................................................................. 30

Tabel 4.1. *Range* nilai HSV ............................................................................. 37

Tabel 4.2. Nilai bias ......................................................................................... 41

# 

# DAFTAR GAMBAR

Gambar 2.1 Indeks skala warna kematangan pisang raja ................................ 8

Gambar 2.2 Pisang Raja Mentah ...................................................................... 8

Gambar 2.3 Pisang Raja Cukup Matang .......................................................... 9

Gambar 2.4 Pisang Raja Matang ...................................................................... 9

Gambar 2.5 Pisang Raja Matang Sekali ........................................................... 10

Gambar 2.6 Pisang Kepok Mentah .................................................................. 10

Gambar 2.7 Pisang Kepok Cukup Matang ....................................................... 11

Gambar 2.8 Pisang Kepok Matang .................................................................. 11

Gambar 2.9 Pisang Kepok Matang .................................................................. 12

Gambar 2.10(a) Koordinat citra digital; (b) Ilustrasi digitalisasi citra (pixel pada koordinat x=10, y=3 memiliki nilai 110) ......................................................... 13

Gambar 2.11 Representasi warna RGB pada citra digital................................ 14

Gambar 2.12 Model Warna HSV ..................................................................... 14 Gambar 2.13 Ilustrasi perubahan nilai piksel pada proses *thresholding .............................................................................................................................* 16

Gambar 2.14 Area Blo ..................................................................................... 17

Gambar 2.15 Penentuan garis pemisah (a); dan garis pemisah terbaik (b) ........................................................................................................................... 18

Gambar 2.16 Ilustrasi *multiclass* klasifikasi visualisasi ...................................19 Gambar 2.17 Diagram alir proses pengujian SVM ............................................20

Gambar 2.18 *Syntax* membaca gambar ..............................................................21

Gambar 2.19 *Syntax* menyimpan gambar ........................................................ 22

Gambar 4.1 Gambaran umum sistem ............................................................... 32

Gambar 4.2 *Flowchart* masukan citra buah pisang .......................................... 33

Gambar 4.3 *Flowchart* Ektraksi Fitur .............................................................. 33

Gambar 4.4 *Flowchart* Identifikasi Citra ......................................................... 34

Gambar 4.5 Nilai matriks hasil konversi RGB ke HSV................................... 37

Gambar 4.6 Citra buah pisang hasil dari konversi ke HSV ............................. 37

Gambar 4.7 Nilai matriks konversi citra HSV ke citra hitam putih ................. 38

Gambar 4.8 Citra buah pisang hasil masking................................................... 38

Gambar 4.9.Citra buah pisang hasil crop deteksi citra .................................... 39

Gambar 4.10 *Usecase Diagram .......................................................................* 42

Gambar 4.11. *Activity* *Diagram* Masukkan Citra ............................................ 43

Gambar 4.12 *Activity* *Diagram* Ekstraksi Fitur ............................................... 43

Gambar 4.13 *Activity* *diagram* simpan fitur ..................................................... 44

Gambar 4.14 *Activity* *diagram* identifikasi ...................................................... 45

Gambar 4.15 *Sequence diagram* masukkan citra ............................................. 45

Gambar 4.16 *Sequence diagram* ekstraksi fitur ............................................... 46

Gambar 4.17 *Sequence diagram* simpan fitur .................................................. 47 Gambar 4.18 Antarmuka Menu Utama ............................................................ 48 Gambar 4.19 Antarmuka Menu Klasifikasi ..................................................... 48

# 

# BAB I

# PENDAHULUAN

## 1.1.Latar Belakang

Pisang *(Musa Paradisiaca)* adalah tanaman buah yang kaya akan sumber vitamin, mineral dan karbohidrat. Buah ini sangat memasyarakat karena dapat dikonsumsi kapan saja dan di segala tingkatan usia dari bayi hingga orang tua (Indarto dkk., 2017). Muna merupakan salah satu kabupaten penghasil buahbuahan tropis yang cukup banyak di Sulawesi Tenggara. Menurut Badan Badan Pusat Statistika Provinsi Sulawesi Tenggara, Kabupaten Muna pada tahun 2018 pisang merupakan komoditas buah terbesar yang banyak diproduksi dibandingkan buah lainnya dengan jumlah 285.518 kw atau 28.551 ton.

Tiap buah memiliki ciri untuk dapat ditentukan jenis dan kematangannya, misalkan saja ukuran dan warnanya. Pada buah pisang digunakan ciri tersebut untuk melakukan klasifikasi. Saat ini, klasifikasi jenis dan kematangan pisang masih dilakukan petani pisang secara manual (Dendy dkk., 2011). Salah satu permasalahan pascapanen pada buah pisang yang diproduksi jumlah banyak adalah dalam hal penyortiran buah pisang yang akan dipasarkan. Selama ini buah pisang diidentifikasi tingkat kematangannya berdasarkan analisis warna kulit buah secara visual mata manusia. Proses identifikasi seperti ini memiliki beberapa kelemahan di antaranya yaitu membutuhkan tenaga lebih banyak untuk memilah, dan tingkat persepsi kematangan buah yang dihasilkan bisa berbeda karena manusia dapat mengalami kelelahan, tidak selalu konsiten, dan penilaian manusia juga bersifat subjektif. Kelemahan-kelemahan tersebut akan mempengaruhi waktu yang dibutuhkan dalam memilah dan mengidentifikasi tingkat kematangan buah pisang (Sigit dkk., 2015).

Pada tahun 2018, Sularida dkk., melakukan penelitian yang berjudul “*Identifikasi Tingkat Kematangan Buah Pisang Menggunakan Metode Ektraksi*

*Ciri Statistik Pada Warna Kulit Buah*”, penelitian ini mengidentifikasi tingkat

1

kematangan buah pisang menggunakan informasi warna citra pada kulit buah pisang. Penelitian ini memiliki 3 tahapan utama yaitu *preprocessing* yang terdiri dari *resize*, konversi citra RGB ke citra l\*a\*b dan citra label,ekstraksi fitur menggunakan metode segmentasi citra RGB dan HSV dan deteksi menggunakan metode klasifikasi *Euclidean Distance*. Dalam proses deteksi, metodologi yang diusulkan diuji pada *database* citra buah pisang*.* Hasil percobaan menunjukkan bahwa metode yang diusulkan mencapai akurasi sampai dengan 90%.

Fiturmerupakan karakteristik unik dari suatu objek. Fitur dibedakan menjadi dua yaitu fitur “alami” merupakan bagian dari gambar, misalnya kecerahan dan tepi objek. Sedangkan fitur “buatan” merupakan fitur yang diperoleh dengan operasi tertentu pada gambar, misalnya histogram tingkat keabuan (Gualtieri dkk., 1985). Sehingga ekstraksi fitur adalah proses untuk mendapatkan ciri-ciri pembeda yang membedakan suatu objek dari objek yang lain (Putra, 2010).

Salah satu metode yang sering digunakan untuk identifikasi adalah *MultiSupport Vector Machine* (M-SVM). *Multi Support Vector Machine* (M-SVM) adalah metode yang dikembangkan dari metode *Support Vector Machine* (SVM). Metode SVM awalnya memisahkan kelas biner (k=2) (Ma & Guo, 2014). Menurut Lingras & Butz (2007) penelitian yang melibatkan kasus di bidang kehidupan nyata cenderung mencakup kasus *multiclass,* Sehingga M-SVM dapat diterapkan pada kasus tersebut*.* Kelebihan *Multi Support Vector Machine* (M-SVM) sendiri yaitu dapat digunakan untuk tujuan klasifikasi linear dan non linear, klasifikasi dapat dilakukan lebih dari dua kelas, klasifikasi ini mudah dikenali dan memberikan hasil yang benar terkait dengan pendekatan klasifikasi lainnya serta klasifikasi yang cukup bagus ketika melatih gambar yang memiliki beberapa *noise* (Yun, 2015).

Berdasarkan latar belakang tersebut, dalam penelitian ini akan dibuat sistem Implementasi Tingkat Kematangan Buah Pisang Menggunakan Metode *Multi Support Vector Machine* (M-SVM) untuk mendapatkan hasil identifikasi tingkat kematangan pada kulit buah pisang yang lebih baik. Pada penelitian akan tingkat kematangan buah pisang diklasifikasikan menjadi 4 kelas yaitu mentah, cukup matang, matang, dan sangat matang.

## 1.2.Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan sebelumnya, dapat dirumuskan permasalahan dalam penelitian ini yaitu bagaimana mengidentifikasi tingkat kematangan tanaman buah pisang raja dan pisang kepok dengan metode *Multi Support Vector Machine* (M-SVM).

## 1.3.Batasan Masalah

Adapun batasan masalah dalam penelitian ini yaitu:

1. Data citra digital buah pisangberjumlah 160 data dengan format JPG diambil dari kebun pisang yang berlokasi di Kabupaten Muna, Kecamatan Kabangka Desa Oensuli. Seluruh citra tersebut dibagi menjadi 2 kelompok, yaitu data citra latih dan data citra uji.
2. Citra digital buah pisang yang diambil terdiri dari 2 jenis, yakni pisang raja dan pisang bugis.
3. Citra digital didapatkan dengan memetik buah pisang dan diambil citranya dengan latar belakang putih.
4. Alat akuisisi citra menggunakan kamera *smartphone*.
5. Jarak kamera dengan daun saat pengambilan citra adalah 30 cm.
6. Proses identifikasi menggunakan citra RGB yang telah melalui proses segmentasi dengan ukuran 300 x 225 piksel.
7. Jumlah kelas yang digunakan sebanyak 4.

## 1.4.Tujuan Penelitian

Tujuan dari penelitian ini yaitu bagaimana mengimplementasikan metode *Multi Support Vector Machine* (M-SVM) untuk menentukan tingkat kematangan buah pisang raja dan pisang kepok*.*

## 1.5.Manfaat Penelitian

Manfaat dari penelitian ini yaitu diharapkan dapat memberikan hasil identifikasi dengan benar dan tingkat akurasi baik sehingga memberi kemudahan bagi para petani untuk penyortiran buah pisang.

## 1.6.Sistematika Penulisan

###### BAB I PENDAHULUAN

Bab ini berisi tentang latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, sistematika penulisan dan tinjauan pustaka.

###### BAB II LANDASAN TEORI

Bab ini berisi tentang pengertian istilah-istilah, konsep dasar citra digital, RGB dan HSV, *Multi Support Vector Machine* (M-SVM), fungsi-fungsi Python yang digunakan dalam penelitian ini, teori tentang *flowchart, Unified Modelling Languange* (UML) dan *Rational Unified Process* (RUP).

###### BAB III METODOLOGI PENELITIAN

Bab ini memuat metodologi penelitian yang meliputi metode pengumpulan data, uraian metode *Rational Unified Process* (RUP) untuk pengembangan sistem dan uraian waktu penelitian.

###### BAB IV ANALISIS DAN PERANCANGAN SISTEM

Bab ini memuat analisis serta rancangan sistem yang akan dibuat. Adapaun rancangan sistem meliputi rancangan proses, rancangan UML dan rancangan antarmuka sistem.

###### BAB V IMPLEMENTASI DAN PENGUJIAN

Bab ini menyajikan implementasi dan pengujian dari sistem yang dibangun. Bagian implementasi menguraikan tentang implementasi secara detail dan runut dari sistem yang dibangun berdasarkan hasil analisis dan rancangan pada bab sebelumnya sedangkan bagian pengujian menguraikan pengujian sistem serta pembahasan hasil pengujian sistem tersebut.

###### BAB VI PENUTUP

Bab ini memuat kesimpulan dari penelitian yang telah dilakukan serta saran untuk penelitian selanjutnya.

## 1.7.Tinjauan Pustaka

Beberapa penelitian yang dijadikan rujukan dalam penelitian ini yaitu: Chamidah (2018) yang melakukan penelitian K-Means sebagai ekstraktor ciri pada klasifikasi data dengan algoritma *Support Vector Machine* (SVM). Data diperoleh dari *UCI Machine Learning Repository*. Data tersebut adalah data *liver* sebanyak 583 *record* yang terdiri dari klasifikasi *liver* sebanyak 167 *record*, dan *nonliver* sebanyak 416 *record*. Data *breast cancer* dengan total 569 *record* yang terdiri dari 357 *record* berupa kanker jinak dan 212 *record* kanker ganas. Data *heart disease* dengan total 303 *record* yang terdiri dari 160 data sehat dan 143 data sakit. Terdapat 4 tahap pada penelitian ini, pertama adalah praproses dilakukan dengan melakukan transformasi *min-max* ke dalam *range* 0-1. Kedua adalah ekstraksi dan seleksi ciri dilakukan untuk mendapatkan pola tersembunyi dari tiap kelas secara terpisah dengan Algoritma K-*Means*. K-*Means* mengelompokkan ciri-ciri yang paling berpengaruh terhadap suatu kelas dengan mencari jarak terdekat antara pusat (*centroid*) dengan anggota *cluster*-nya. Ketiga rekonstruksi ciri yang berguna untuk mengetahui kecocokan data dengan pola baru yang telah ditemukan. Dan tahap keempat adalah klasifikasi SVM penelitian ini memanfaatkan *hyperplane* untuk pemodelan. Dari penelitian tersebut didapatkan akurasi terbesar mencapai 96,17%.

Penetilian kedua dilakukan oleh Teny Handayani (2017) yang berjudul Identifikasi Penulisan Melalui Pola Penulisan Tangan Menggunakan Algoritma Support Vector Machine. Pada penelitian ini dilakukan dengan menggunakan data citra tulisan tangan dari 47 responden, dengan pembagian data latih dan data uji secara acak 70%:30%. Responden menjadi kelas target, sehingga dari data diperoleh 47 kelas. Pada eksperimen pertama, identifikasi penulis melalui pola tulisan tangan dilakukan dengan membandingkan pola tulisan dari setiap responden secara berpasangan. Support Vector Machine kernel linear berhasil mengidentifikasi penulis dengan rata-rata akurasi 99%. Pada eksperimen ke dua, identifikasi penulis dilakukan dengan menggunakan keseluruhan data dengan menghasilkan rata-rata akurasi 93,5%.

Penelitian ke tiga dilakukan oleh Selvia dkk (2017) yang berjudul Otomatisasi Klasifikasi Kematangan buah Mengkudu Berdasarkan Warna dan Tektur. Pada penelitian ini mengklasifikasikan data berdasarkan fitur Warna dan Fitur Tekstur. Berdasarkan penelitian hasil klasifikasi diperoleh SVM kernel polynomial untuk fitur tektur dan warna didapatkan secara berurutan 77,22% dan 83,30%, sedangkan SVM kernel linear untuk fitur tektur dan warna didapatkan secara berurutan 81,11% dan 87,22%.

Dari ketiga penelitian tersebut, diketahui dua hal penting tentang metode *Support Vector Machine.* Pertama*,* bahwa metode *Support Vector Machine* dapat digunakan untuk melakukan klasifikasi pada citra daun dan menghasilkan hasil klasifikasi yang lebih baik. Kedua, SVM merupakan salah satu metode klasifikasi yang baik dalam pemecahan masalah untuk dua kelas.

# BAB II

# LANDASAN TEORI

## 2.1. Pisang

Tanaman pisang (*Musa Paradisiaca*) merupakaan tanaman hortikultura penting karena kaya akan nilai gizi dan mempunyai nilai ekonomis yang tinggi (Suhartanto dkk., 2012). Buah pisang mempunyai kandungan gizi sangat baik, yaitu menyediakan energi cukup tinggi dibandingkan dengan buah-buahan lain. Pisang kaya mineral seperti kalium, magnesium, fosfor, besi, dan kalsium. Pisang juga mengandung vitamin yaitu C, B kompleks, B6 dan serotonin yang aktif sebagai neuro transmitter dalam kelancaran fungsi otak. Bahkan, kulit pisang yang selama ini hanya dianggap sebagai limbah ternyata dapat dimanfaatkan untuk menyerap limbah logam berat (Rizkika dkk., 2012).

Pisang terbagi menjadi 2 kelompok yaitu kelompok pisang konsumsi segar dengan contoh pisang raja bulu dan kelompok pisang konsumsi olahan dengan contoh pisang kepok.

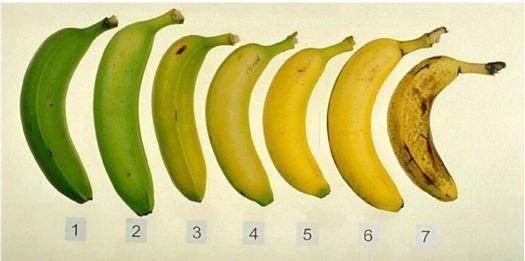
## 2.2.Kriteria Kematangan Buah Pisang

Pascapanen petani seringkali menentukan buah pisang berdasarkan pengalaman dengan ciri-ciri fisik pada buah, meliputi bentuk buah, ukuran, dan warna kulit buahnya. Beberapa tanda atau ciri sering digunakan sebagai kriteria untuk menentukan buah pisang dapat dipanen salah satunya yaitu berdasarkan warna kulit buah. Perubahan skala warna kulit buah pisang digunakan sebagai petunjuk untuk mengetahui tahapan pematangan buah pisang. Indeks skala warna kulit buah pisang (Sutowijoyo, 2013). Menurut Kader (2008), derajat kekuningan kulit buah pisang dinilai dengan angka antara 1 sampai 7 seperti pada Gambar 2.1, nilai derajat kekuningan kulit buah tersebut adalah:

1: Hijau 5 : Kuning dengan ujung hijau

2: Hijau dengan sedikit kuning 6 : Kuning penuh

3: Hijau kekuningan Kuning dengan bintik coklat 4. Kuning lebih banyak dari hijau



###### Gambar 2.1 Indeks skala warna kematangan pisang

### 2.2.1. Pisang Raja

Pisang raja memilki bentuk buah yang bulat memanjang dan ujung yang meruncing. Warna buah ini ketika masak yaitu kuning agak merah di sertai ujung yang hijau. Daging buah pisang ini kuning dengan rasa yang manis dan aroma harum.

##### 2.2.1.1.Pisang Raja Mentah

Seluruh permukaan kulit buah pisang raja mentah berwarna hijau dengan semburat atau sedikit warna kuning dan tekstur buah masih keras. Seperti pada Gambar 2.2.



###### Gambar 2.2 Pisang Raja Mentah

##### 2.2.1.2.Pisang Raja Cukup Matang

Permukaan kulit buah pisang raja warna hijau lebih dominan dari pada kuning dan tekstur buah mulai lunak. Seperti pada Gambar 2.3.



###### Gambar 2.3 Pisang Raja Cukup Matang

##### 2.2.1.3.Pisang Raja Matang

Seluruh permukaan kulit buah pisang raja matang berwarna kuning bagian ujung masih hijau, terdapat bintik coklat dikulit, tandan masih hijau, bentuk pisang mulai membulat. Seperti pada Gambar 2.4.



###### Gambar 2.4 Pisang Raja Matang

##### 2.2.1.4.Pisang Raja Matang Sekali

Permukaan buah pisang raja matang sekali berwarna kuning dengan banyak bercorak coklat, tandan berwarna hijau dan tekstur lunak sekali. Seperti pada Gambar 2.5.



###### Gambar 2.5 Pisang Raja Matang Sekali

### 2.2.2. Pisang Kepok

Pisang kepok merupakan salah satu buah pisang yang enak dimakan setelah setelah diolah terlebih dahulu. Pisang kepok memiliki buah yang sedikit pipih dan kulit yang tebal, jika sudah matang warna kulit buahnya akan menjadi kuning. Pisang kepok memiliki banyak jenis, namun yang lebih dikenal adalah pisang kepok putih dan pisang kepok kuning. Warna buahnya sesuai dengan nama jenis pisangnya, yaitu putih dan kuning (Prabawati dkk, 2008).

##### 2.2.2.1.Pisang Kepok Mentah

Permukaan kulit buah pisang kepok mentah berwarna hijau gelap, tandan pisang berwarna hijau dan tekstur pisang keras. Seperti pada Gambar 2.6.



###### Gambar 2.6 Pisang Kepok Mentah

##### 2.2.2.2.Pisang Kepok Cukup Matang

Permukaan kulit pisang kepok cukup matang yaitu berwarna hijau terang dan mulai muncul bercak kuning. Ujung pisang masih berwarna hijau tua. Seperti pada Gambar 2.7.



###### Gambar 2.7 Pisang Kepok Cukup Matang

##### 2.2.2.3.Pisang Kepok Matang

Permukaan kulit buah pisang kepok matang berwarna kuning langsat dan permukaannya sedikit berwarana hijau. Seperti pada Gambar 2.8.



###### Gambar 2.8 Pisang Kepok Matang

##### 2.2.2.4.Pisang Kepok Matang Sekali

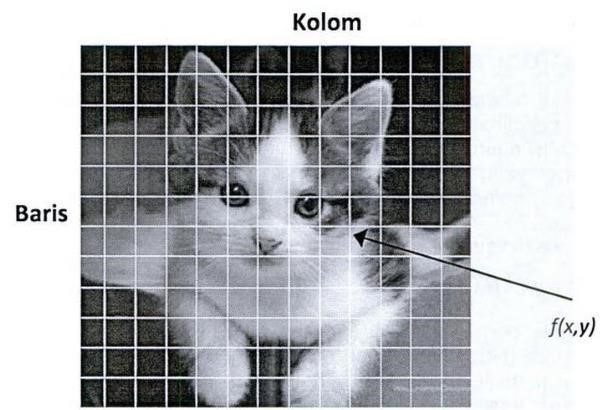
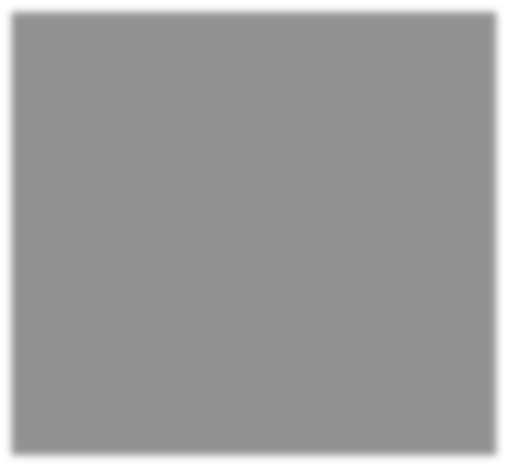
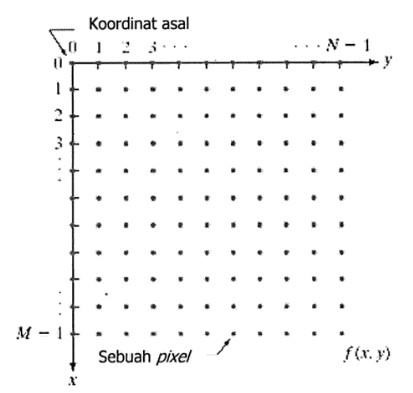
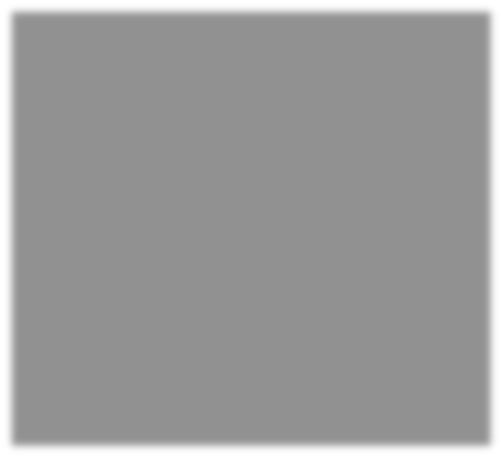
Seluruh permukaan kulit buah pisang kepok matang sekali berwarna kuning cerah dan berwarna kuning dengan bercak kecoklatan. Seperti pada Gambar 2.9.



###### Gambar 2.9 Pisang Kepok Matang

## 2.3. Citra Digital

Suatu citra dapat didefinisikan sebagai fungsi *f(x,y)* berukuran M baris dan N kolom, dengan *x* dan *y* adalah koordinat spasial, dan amplitudo *f* di titik koordinat *(x,y)* dinamakan intensitas atau tingkat keabuan dari citra pada titik tersebut (Putra, 2010). Apabila *x*, *y* dan besaran *f* mempunyai nilai terbatas dengan kuantitas diskrit, maka disebut sebagai citra digital. Citra berwarna (*color images*) dikenal dengan nama citra spektral, karena warna pada citra disusun oleh tiga komponen warna yang disebut komponen RGB, yaitu: merah (*red*), hijau (*green*), dan biru (*blue*). Intensitas suatu titik pada citra berwarna merupakan kombinasi dari tiga intensitas yaitu derajat keabuan merah (𝑓𝑚𝑒𝑟𝑎ℎ(*x,y*)), hijau (𝑓ℎ𝑖𝑗𝑎𝑢(*x, y*)), dan biru (𝑓𝑏𝑖𝑟𝑢(*x,y*)). Dalam citra berwarna, jumlah warna mulai dari 16, 256, 65.536 atau 16 juta warna yang masing-masing dipresentasikan oleh 4, 8, 16 atau 24 bit data untuk setiap piksel (Gonzalez dan Woods, 2008).



(a) (b)

###### Gambar 2.10 (a) Koordinat citra digital; (b) Ilustrasi digitalisasi citra (pixel pada koordinat x=10, y=3 memiliki nilai 110) (Putra, 2010)

## 2.4.Segmentasi

Segmentasi citra adalah proses membagi citra digital menjadi beberapa daerah atau kelompok, dengan masing-masing daerah terdiri dari sekumpulan piksel. Segmentasi citra menyederhanakan dan mengubah representasi citra ke sesuatu yang lebih bermakna dan lebih mudah untuk dianalisis. Segmentasi citra digunakan untuk mencari objek yang ingin dicari dan batas-batas bentuk objek seperti garis, kurva dalam citra (Munandar, 2017).

### 2.4.1. Konversi citra RGB ke HSV

Citra RGB memiliki warna tertentu, warna tersebut adalah merah (*Red*), hijau (*Green*) dan biru (*Blue*). Jika masing-masing warna memiliki *range* 0 - 255, maka totalnya adalah 2553 = 16.581.375 (16 K) variasi warna berbeda pada gambar, dimana variasi warna ini cukup untuk gambar apapun. Karena jumlah bit yang diperlukan untuk setiap piksel, gambar tersebut juga disebut gambar-bit warna (Kusumanto dan Tompunu, 2011). Sebuah jenis warna, dapat dibayangkan sebagai sebuah vektor di ruang dimensi 3 yang biasanya dipakai dalam matematika, koordinatnya dinyatakan dalam bentuk tiga bilangan, yaitu komponen-x, komponen-y dan komponen-z. Misalkan sebuah

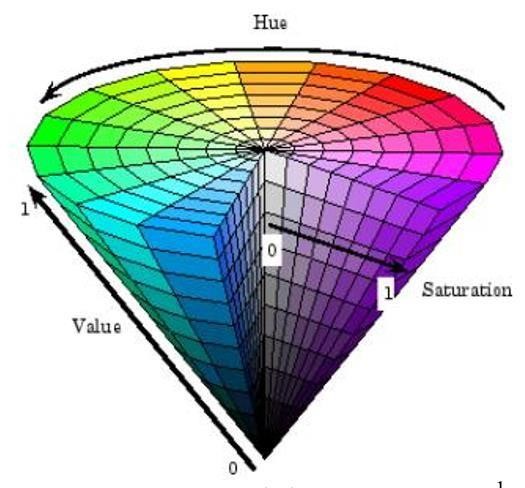
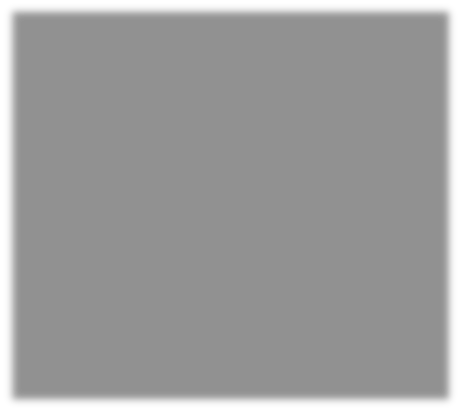
vektor dituliskan sebagai r = (x,y,z). Untuk warna, komponen-komponen tersebut digantikan oleh komponen *R(ed), G(reen), B(lue).* Jadi, sebuah jenis warna dapat dituliskan sebagai berikut: warna = RGB (30, 75, 255). Putih = RGB (255,255,255), sedangkan untuk hitam= RGB(0,0,0). Bentuk representasi warna dari sebuah citra digitial dapat dilihat pada Gambar 2.11 (Muchtar, 2012).



###### Gambar 2.11 Representasi warna RGB pada citra digital

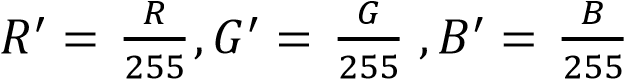
**(Kadir dan Susanto, 2012)**

Model warna HSV mendefinisikan warna dalam terminologi *Hue, Saturation, dan Value. Hue* menyatakan warna sebenarnya, seperti merah, violet, dan kuning. *Hue* digunakan untuk membedakan warna-warna dan menentukan kemerahan (*redness*), Kehijauan (*greenness*) dari cahaya (Rafael C. Gonzalez & Woods, 2002). Nilai *hue* antara 0 sampai 1 berarti warna antara merah melewati kuning, hijau, cyan, biru dan magenta dan kembali menjadi merah. Nilai *saturation* antara 0 sampai 1 berarti dari tidak tersaturutasi (keabuan) sampai tersaturisasi penuh (tidak putih) (Muhammad et al,2013).



###### Gambar 2.12 Model Warna HSV (Syahid dkk, 2016)

Ruang warna HSV diperoleh dengan menggunakan konversi RGB sbb : Nilai R, G dan B dibagi dengan diperoleh range nilai dari 0 hingga 1 dinyatakan pada persamaan 2.1 (Hidayat, 2015).

 (2.1)

Kemudian dihitung nilai maks minimumnya dengan menggunakan persamaan 2.2 (Hidayat, 2015).

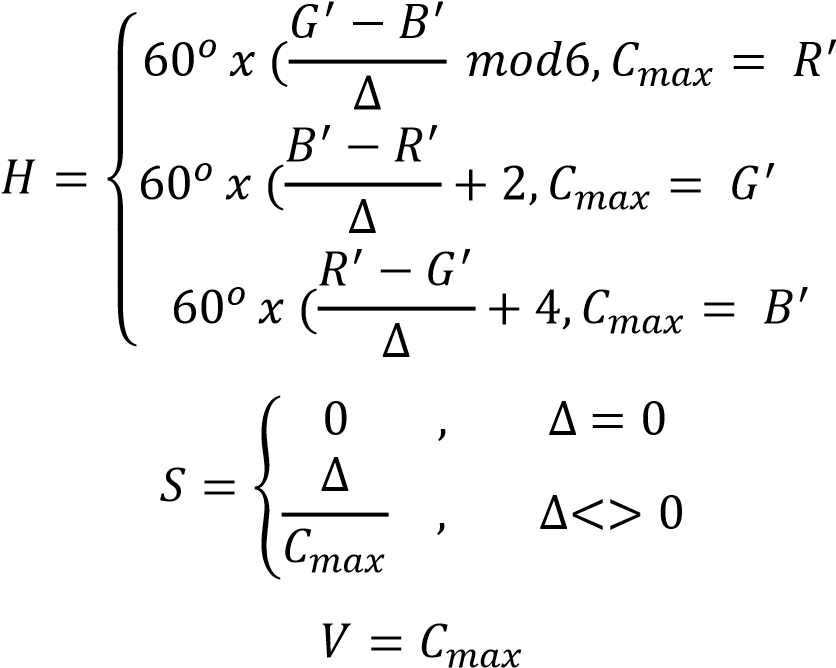


 (2.2)



Ruang warna HSV mempunyai elemen *Hue* (H), *Saturation* (S) dan *Value* (V).

Adapun konversinya dinyatakan dengan persamaan 2.3, 2.4 dan 2.5 (Hidayat, 2015).

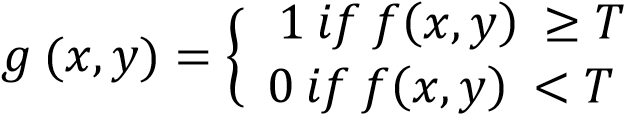
 (2.3)

(2.4)

(2.5)

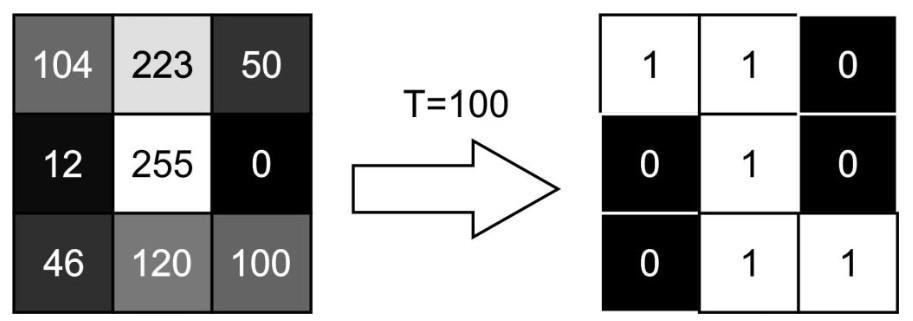
### 2.4.2. *Thresholding*

Citra biner adalah citra yang memiliki dua nilai tingkat keabuan yaitu hitam dan putih. Secara umum proses binerisasi citra *grayscale* untuk menghasilkan citra biner adalah sebagai berikut (Putra, 2004).

 (2.6)

dengan *g(x,y)* adalah citra biner dari citra *grayscale f(x,y)* dan *T* menyatakan nilai

ambang.



###### Gambar 2.13 Ilustrasi perubahan nilai piksel pada proses *thresholding*

### 2.4.3. *Masking image*

*Masking image* adalah proses memisahkan gambar dari latar belakangnya, baik untuk menyebabkan gambar menonjol sendiri atau menempatkan gambar di atas latar belakang lain (Veaux, 2015). *Masking Image* dapat membuat bagian mana pun sepenuhnya tidak terlihat atau sebagian terlihat.

### 2.4.4. *Blob Detection*

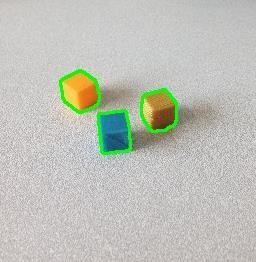
*Blob* atau *Binary Large Object* adalah sekumpulan piksel yang saling terhubung dan membentuk suatu objek. Deteksi *blob* yaitu mendeteksi kumpulan titik-titik piksel yang saling terhubung dan memiliki warna berbeda (lebih terang atau lebih gelap) dari latar belakang dan menyatukannya dalam suatu *region*. Kelebihan utama dari teknik ini ialah memiliki fleksibilitas yang tinggi dan kemampuan yang sangat baik. Namun batas dari teknik ini ialah perbedaan *background* dan *foreground* yang harus jelas (Adaptive Vision, n.d).

*Blob Analysis* terdiri dari *Blob extraction* untuk mengisolasi *Blob* (objek) dalam citra biner, dimana *blob* terdiri dari kelompok piksel yang saling berhubungan ditunjukkan pada Gambar 2.14. Piksel yang berhubungan disebut *connectivity*, yang pikselnya dapat bertetanggaan atau tidak. Analisis *blob* termasuk dalam analisis terhadap luas area dan bentuk objek *blob* dari suatu citra yang menjadi fokus deteksi. Dalam menentukan nilai *blob,* ada beberapa hal yang harus diketahui untuk menghasilkan sebuah *blob* yang optimal. Penentuan luas *blob* pada setiap objek pada proses segmentasi *foreground* perlu dianalisis karena nilai *blob* pada tiap objek akan berbeda. Hal ini dipengaruhi oleh fitur objek seperti ukuran, jenis, dan teknik pengambilan data video. Prosesnya dimulai dari penandaan area *foreground* yang dianggap objek, kemudian pengumpulan data area menjadi *blob* seperti posisi piksel awal, panjang terhadap sumbu x dan sumbu y dan luas area piksel.

Metode *blob detection* terdiri dari langkah-langkah berikut:

1. Ekstraksi – proses yang diterapkan untuk mendapatkan wilayah objek tertentu yang sedang dicari. Proses ini biasanya dilakukan menggunakan segmentasi warna.
2. Perbaikan - wilayah yang diekstraksi diperbaiki dengan menghilangkan *noise* yang ada. Proses ini biasanya dilakukan menggunakan teknik transformasi wilayah.

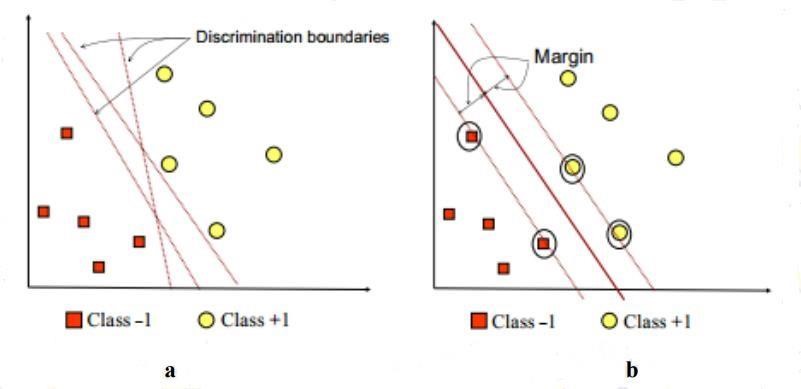
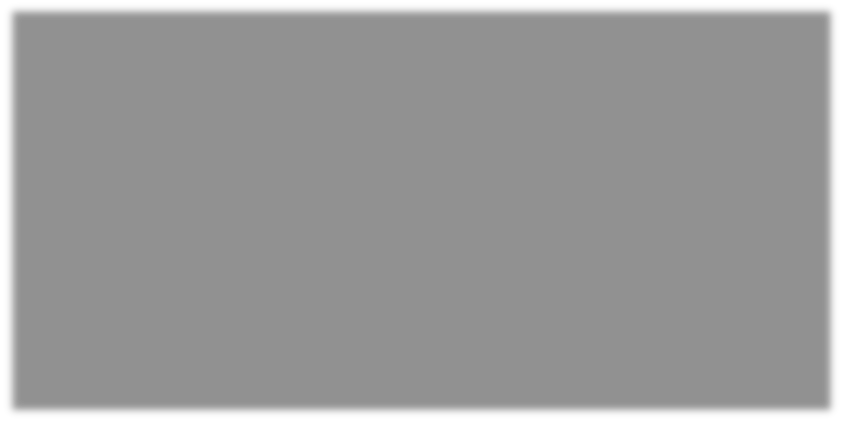
Analisis – proses ini dilakukan untuk mengekstrak informasi yang dibutuhkan.



###### Gambar 2.14. Area *blob*

## 2.5. M-SVM (*Multi* *Support Vector Machine*)

SVM saat pertama kali diperkenalkan oleh Vapnik (1995), hanya dapat mengklasifikasikan data ke dalam dua kelas (klasifikasi biner). SVM merupakan metode *machine learning* yang bertujuan menemukan *hyperplane* terbaik yang memisahkan dua buah kelas pada *input space.* Cara menemukan *hyperplane* ditunjukan pada gambar 2.15 (Nugroho dkk, 2003).



**Gambar 2.15 Penentuan garis pemisah (a); dan garis pemisah terbaik (b).**

###### (Nugroho dkk, 2003)

Gambar 2.15 menunjukan beberapa *pattern* yang merupakan anggota dari dua buah kelas : +1 dan -1. *Pattern* yang tergabung pada kelas -1 disimbolkan dengan warna merah (kotak), sedangkan *pattern* pada kelas +1, disimbolkan dengan warna kuning (lingkaran). Problem klasifikasi dapat diterjemahkan dengan usaha menemukan garis (*hyperplane*) yang memisahkan antara kedua kelompok tersebut. Berbagai alternatif garis pemisah (*discrimination boundaries*) ditunjukkan pada gambar 2.15 (a). *Hyperplane* pemisah terbaik antara dua kelas dapat ditemukan dengan mengukur *margin hyperplane* tersebut dan mencari titik maksimalnya. *Margin* adalah jarak antara *hyperplane* tersebut dengan *pattern* terdekat dengan masing-masing kelas. *Pattern* paling dekat ini disebut dengan *support vector.* Garis tebal pada gambar 2.15 (b) menunjukkan *hyperplane* yang terbaik, yaitu yang terletak tepat pada tengah-tengah kedua kelas, sedangkan titik merah dan kuning yang berada dalam lingkaran hitam adalah *support vector.* Pada SVM, terdapat parameterC berperan sebagai parameter *penalty,* yang menentukan seberapa besar nilai *penalty* yang akan kita berikan terhadap kesalahan klasifikasi.

Ada beberapa jenis kernel SVM, dengan tipe kernel yang paling umum digunakan adalah kernel linear (Muchtar, 2015), dengan fungsi kernel seperti pada persamaan 2.7



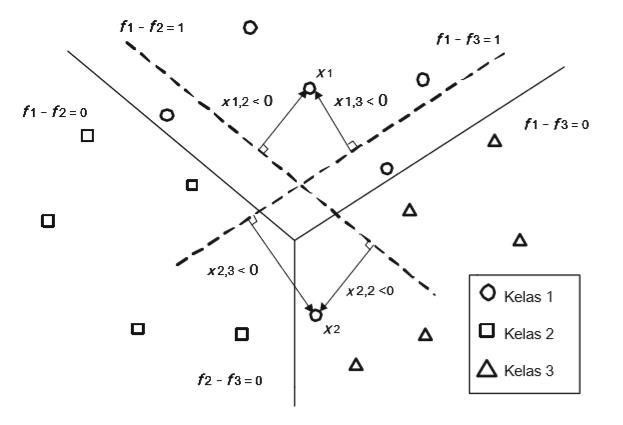
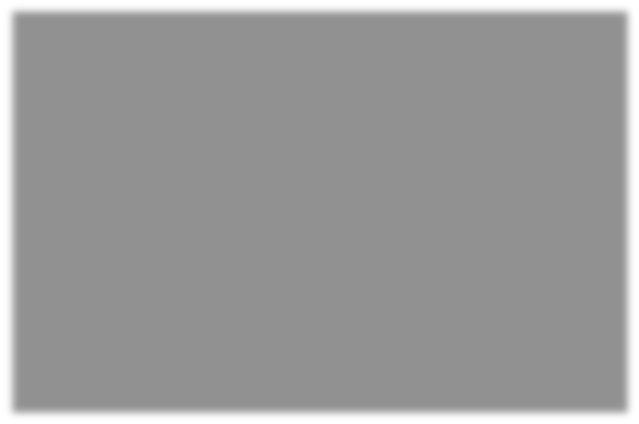
*Subject to*

 (2.7)

Dengan  adalah data *input*,adalah *output* yang nilainya =1 atau -1, w dan b adalah parameter yang kita cari nilainya, dimana w adalah normal bidang dan b adalah posisi bidang relatif. Dalam kasus yang tidak *feasible* dimana beberapa data mungkin tidak bisa dikelompokkan secara benar maka dapat digunakan kernel *rbf*. Dalam menyelesaikan klasifikasi *rbf*, SVM memecahkan masalah *dimensionality disaster* dengan memperkenalkan fungsi kernel yang saat ini digunakan adalah kernel polinomial dalam Persamaan 2.8 (Qian dkk, 2018).

(2.8)

Penelitian lebih lanjut untuk mengembangkan SVM sehingga bisa mengklasifikasi data yang memiliki lebih dari dua kelas, terus dilakukan.

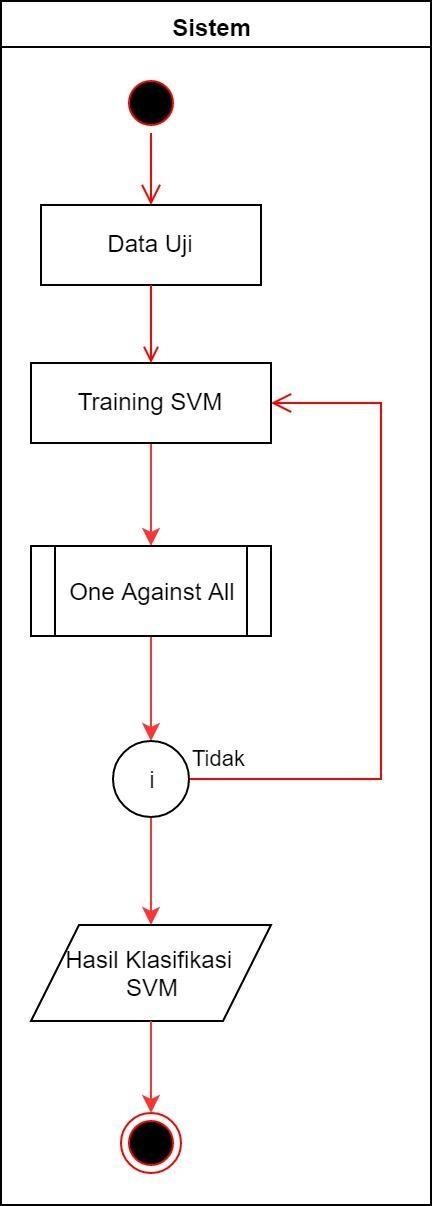


**Gambar 2.16 Ilustrasi *multiclass* klasifikasi visualisasi (Wang dan Xue, 2014).**

Pada gambar 2.16 merupakan ilustrasi *multiclass* di mana tiga kelas yang direpresentasikan sebagai lingkaran, persegi panjang, dan segitiga. Garis tebal mewakili beberapa batasan keputusan.

Ada dua pilihan untuk mengimplementasikan *multiclass* SVM yaitu dengan menggabungkan beberapa SVM biner atau menggabungkan semua data yang terdiri dari beberapa kelas ke dalam sebuah bentuk permasalah optimasi. Metode yang umum digunakan untuk mengimplementasikan *multiclass* SVM dengan menggunakan metode *one-against-all*, dengan menggunakan metode ini, dibangun *k* buah model SVM biner (*k* adalah jumlah kelas). Setiap model klasifikasi ke-*i* dilatih dengan menggunakan keseluruhan data, untuk mencari solusi permasalahan

(Banyal dkk, 2016). Gambar 2.17 akan menampilkan diagram alir proses *One Against All.*



###### Gambar 2.17 Diagram alir proses pengujian SVM

Pendekatan *one-against-all* bekerja dengan cara menjadikan kelas dari data ke-*i* bernilai positif dan untuk data yang tidak berasal dari kelas ke-*i* bernilai negatif. Pada Tabel 2.1 akan dijelaskan contoh permasalahan klasifikasi dengan menggunakan tiga kelas (Sembiring, 2007).

###### Tabel 2.1 Contoh SVM pendekatan *One Against All*

|  |  |  |
| --- | --- | --- |
|  |  | Hipotesis *Kernel* |
| Kelas 1 | Bukan kelas 1 |  |
| Kelas 2 | Bukan kelas 2 |  |
| Kelas 3 | Bukan kelas 3 |  |

## 2.6. Python

Python adalah bahasa pemrograman *interpretatif* multiguna dengan filosofi perancangan yang berfokus pada tingkat keterbacaan kode. Python diklaim sebagai bahasa yang menggabungkan kapabilitas, kemampuan, dengan sintaksis kode yang sangat jelas dan dilengkapi dengan fungsionalitas pustaka standar yang besar serta komprehensif.

Python merupakan salah satu bahasa pemrograman yang populer di dunia kerja Indonesia. Python juga merupakan bahasa pemrograman yang relatif mudah untuk pemula. Selain itu di akademik pun banyak akademisi yang menggunakan python untuk menyelesaikan penelitiannya di bidang robotika, *data science*, ekonomi, antariksa dan berbagai macam bidang lainnya. Python secara *default* telah terpasang di beberapa sistem operasi berbasis Linux seperti Ubuntu, Linux Mint, dan Fedora. Untuk sistem operasi lain, sudah tersedia *installer* yang disediakan untuk sistem operasi tersebut.

*1. Syntax* membaca gambar

Fungsi **cv2.imread ()** untuk membaca gambar. Gambar harus berada dalam direktori kerja.

import numpy as np import cv2

img=cv2.imread(‘contoh.jpg’,0)

###### Gambar 2. 18 *Syntax* membaca gambar

*2. Syntax* menyimpan gambar

Fungsi **cv2.imwrite ()** untuk menyimpan gambar.

* Pertama adalah nama file.
* Kedua adalah gambar yang ingin disimpan.

cv2.imwrite (‘contoh.jpg’,img)

###### Gambar 2. 19 *Syntax* menyimpan gambar

Python memiliki kelebihan tersendiri dibandingkan dengan bahasa pemrograman lainnya terutama dalam hal penanganan modul, ini yang membuat beberapa programmer menyukai Python. Selain itu Python merupakan salah satu produk yang bersifat open source, free, dan multiplatform. Beberapa feature yang dimiliki Python antara lain.

1. Memiliki library yang luas, dalam distribusi Python telah disediakan modul-modul siap pakai untuk berbagai keperluan.
2. Memiliki tata bahasa yang jernih dan mudah dipelajari.
3. Memiliki aturan layout source code yang memudahkan pengecekan, pembacaan kembali, dan penulisan ulang source code tersebut.
4. Bersifat OOP.
5. Memiliki sistem pengelolaan memory yang auto, garbage collection, seperti Java.
6. Bersifat modular, mudah dikembangkan dengan menciptakan modulmodul baru, dimana modul-modul tersebut dapat dibangun dengan bahasa Python maupun C/C++.

## 2.7. *Flowchart*

Ada dua *tool* yang sering digunakan untuk membantu menyusun dokumen logika pemrograman, yaitu *flowchart* dan *pseudocode* (kode semu). *Flowchart* adalah simbol-simbol pekerjaan yang menunjukkan bagan aliran proses yang sering terhubung. Jadi, setiap simbol *flowchart* melambangkan pekerjaan dan instruksinya. Simbol-simbol *flowchart* adalah standar yang ditentukan oleh *American National Standard Institute Inc* (Community, 2012).

Simbol–simbol yang digunakan dalam *flowchart* ditunjukkan pada Tabel 2.2 berikut:

###### Tabel 2.2 Simbol-simbol dalam *Flowchart*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Simbol** | | |  | | **Keterangan** |
| 1. |  | | |  | | Simbol *Start* atau *End* yang mendefinisikan awal atau akhir dari sebuah *flowchart.* |
| 2. |  |  | |  |  | Simbol pemrosesan yang terjadi pada sebuah alur kerja. |
|  | |  |
| 3. |  | | |  | | Simbol *Input/Output* yang mendefinisikan masukkan dan keluaran proses. |
| 4. |  | | |  | | Simbol untuk memutuskan proses lanjutan dari kondisi tertentu. |
| 5. |  | | |  | | *Storage* menyatakan *input* berasal dari *storage* atau *output* di simpan ke *storage*. |
| 6. |  | | |  | | *Preparation* merupakan simbol persiapan yang digunakan untuk memberi nilai awal suatu besaran. |
| 7. |  |  |  |  |  | Simbol yang menyatakan bagian dari program (subprogram). |
|  | |  |
| 8. |  | | |  | | Simbol untuk menghubungkan antar proses atau antar simbol. |

## 2.8.*Unified Modelling Languange* (UML)

UML disebut sebagai bahasa pemodelan bukan metode. Kebanyakan metode terdiri paling sedikit prinsip, bahasa pemodelan dan proses. Bahasa pemodelan (sebagian besar grafik) merupakan notasi dari metode yang digunakan untuk mendesain secara cepat. Bahasa pemodelan merupakan bagian terpenting dari metode. Ini merupakan bagian kunci tertentu untuk komunikasi. UML merupakan bahasa standar untuk penulisan *blue print software* yang digunakan untuk visualisasi, spesifikasi, pembentukan dan pendokumentasian alat-alat dari sistem perangkat lunak (Shalahudin, 2011).

Pada kenyataannya, pendapat orang–orang tentang UML berbeda satu sama lain. Hal ini dikarenakan oleh sejarahnya sendiri dan oleh perbedaan persepsi tentang apa yang membuat sebuah proses rancang–bangun perangkat lunak efektif (Sukamto dkk., 2013).

### 2.8.1. *Usecase* *Diagram*

Menggambarkan sejumlah *external actors* dan hubungannya ke *usecase* yang diberikan oleh sistem. *Usecase* adalah deskripsi fungsi yang disediakan oleh sistem dalam bentuk teks sebagai dokumentasi dari *usecase symbol* namun dapat juga dilakukan dalam *activity diagram*. *Usecase* digambarkan hanya yang dilihat dari luar oleh *actor* (keadaan lingkungan sistem yang dilihat *user* dan bukan bagaimana fungsi yang ada di dalam sistem (Shalahudin, 2011).

**Tabel 2.3 Simbol *Usecase* *Diagram***

|  |  |  |
| --- | --- | --- |
| **No.** | **Simbol** | **Keterangan** |
| 1. |  | Aktor, Menunjukkan *user* yang akan menggunakan sistem. |
| 2. |  | *Usecase,* Menunjukkan proses yang terjadi pada sistem. |
| 3. |  | *Undirectional Association,* Menunjukkan hubungan antara aktor dengan dan *usecase* atau antar *usecase.* |
| 4. | << include >> | *Include,*Relasi *use case* tambahan ke sebuah *use case* yang ditambahkan karena memerlukan *use* *case* ini untuk menjalankan fungsinya atau sebagai syarat dijalankan *use case* ini. |

### 2.8.2. *Sequence Diagram*

Diagram sekuen adalah diagram yang menggambarkan kelakuan objek pada sekuen dengan mendeskripsikan waktu hidup objek dengan *message* yang dikirimkan dan diterima antar objek. Menggambarkan kolaborasi dinamis antara sejumlah objek. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara objek juga interaksi antara objek, sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem (Shalahudin, 2011).

###### Tabel 2.4 Simbol *Sequence* *Diagram*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Sim** | | | **bol** | | **Keterangan** |
| 1. |  |  | |  |  | Objek, Menunjukkan objek yang yang terdapat di diagram *sequence.* |
|  | |  |
| 2. |  | | |  | | Pesan ke objek sendiri, Menunjukkan pesan yang diproses pada objek itu sendiri. |
| 3. |  | | |  | | Pesan objek, Menunjukkan pesan yang disampaikan ke objek lain dalam diagram *sequence.* |
| 4. |  | | |  | | Garis hidup*/lifeline*,Menyatakan kehidupan suatu objek. |
| 5. |  | |  |  | | Waktu aktif,Simbol yang mendeskripsikan objek dalam keadaan aktif dan berinteraksi dengan objek lain. |
|  |

### *2.8.3. Activity* *Diagram*

Menggambarkan rangkaian aliran dari aktivitas, digunakan untuk mendeskripsikan aktivitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktivitas lainnya seperti *usecase* atau interaksi (Shalahudin, 2011). Diagram aktivitas sistem tidak menggambarkan aktivitas yang dilakukan aktor, namun menggambarkan aktivitas yang dapat dilakukan sistem.

###### Tabel 2.5 Simbol *Activity* *Diagram*

|  |  |  |
| --- | --- | --- |
| **No.** | **Simbol** | **Keterangan** |
| 1. |  | *Start* atau kondisi awal, Menunjukkan awal dari suatu diagram aktivitas. |
| 2. |  | *End* atau kondisi akhir, Menunjukkan akhir dari suatu diagram aktivitas. |
| 3. |  | Kondisi transisi, Menunjukkan kondisi transisi antar aktivitas. |
| 4. |  | Swimlane, Menunjukkan aktor dari diagram aktivitas yang dibuat. |
| 5. |  | Aktivitas, Menunjukkan aktivitasa ktivitas yang terdapat pada diagram aktivitas. |
| 6. |  | *Decision* atau pengecekan kondisi, Menunjukkan pengecekan terhadap suatu kondisi. |

## 2.9. Metodologi Pengembangan Sistem (*Rational Unified Process*)

RUP *(Rational Unified Process)*adalah metode pengembangan sistem yang melakukan pendekatan pengembangan perangkat lunak secara berulang-ulang (*iterative*), fokus pada arsitektur (*architecture-centric*), serta lebih diarahkan berdasarkan penggunaan kasus (*use case* *driven*). RUP merupakan proses rekayasa perangkat lunak dengan pendefinisian yang baik (*well defined*) dan penstrukturan yang baik (*well structured*). Adapun 4 tahapan kerja dari RUP sebagai berikut (Sukamto dan Shalahudin, 2013).

1. Fase *Inception* (Permulaan)

Pada tahap ini dilakukan pemodelan proses bisnis yang dibutuhkan (*business modeling*) dan pendefinisian kebutuhan sistem yang akan dibuat (*requirements*).

1. Fase *Elaboration* (Perluasan/Perencanaan)

Pada tahap ini dilakukan pengecekan terhadap arsitektur sistem yang diinginkan, dan terhadap kemungkinan resiko yang terjadi dari arsitektur yang dibuat. Tahap ini lebih memfokuskan pada analisis dan desain sistem serta implementasi sistem yang masih berupa *prototype*.

1. Fase *Construction* (Konstruksi)

Tahap ini difokuskan pada pengembangan komponen dan fitur-fitur sistem. Pada tahap ini lebih difokuskan pada implementasi dan pengujian sistem yang berupa implementasi perangkat lunak pada kode program.

1. Fase *Transition* (Transisi)

Tahap ini lebih difokuskan pada *deployment* atau instalasi sistem agar dapat dimengerti oleh *user*. Aktivitas pada tahap ini meliputi pelatihan *user*, pemeliharaan sistem serta pengujian sistem untuk mengetahui jika sistem telah sesuai dengan harapan *user*.

# BAB III

# METODOLOGI PENELITIAN

## 3.1 Metode Pengumpulan Data

Beberapa metode pengumpulan data yang digunakan dalam membangun sistem Implementasi Metode *Multi Support Vector Machine* Untuk Menentukan Tingkat Kematangan Buah Pisang Raja dan Pisang Kepok pada penelitian ini yaitu sebagai berikut:

1. Wawancara

Metode wawancara dilakukan dengan dosen Fakultas Pertanian Universitas Haluoleo yaitu Bapak **Dr. La Ode Afa, S.M., M.Si.**, **Eko A. Johan, S.P., M.P.** dan **Sakir, S.TP., M.Sc** untuk memperoleh informasi mengenai tingkat kematangan buah pisang raja dan pisang kepok.

1. Kepustakaan

Metode kepustakaan dilakukan untuk mencari literatur atau sumber pustaka yang berkaitan dengan kematangan buah pisang, metode *Multi Support Vector Machine,* metode ekstraksi fitur statistik yang digunakan dalam penelitian ini yaitu nilai rata-rata piksel untuk ruang warna RGB pada Python, dan beberapa teori lainnya terkait pembuatan sistem pada Python.

1. Pengambilan Data

Data citra digital buah pisang diperoleh dari akuisisi citra, data tersebut merupakan data citra digital yang diambil dari kebun pisang yang berlokasi di Desa Oensuli, Kecamatan Kabangka, Kabupaten Muna.

## 3.2 Metode Pengembangan Sistem

Pengembangan sistem Identifikasi Tingkat Kematangan Buah Pisang Menggunakan Metode *Backpropagation* pada penelitian ini menggunakan metode *Rational Unified Proccess* (RUP). Berikut uraian kegiatan yang dilakukan pada setiap tahapan pengembangan sistem dalam penelitian ini.

28

29

### 3.2.1 *Inception*

Pada tahapini dilakukan proses identifikasi sistem yang diawali dengan analisis kebutuhan sistem. Kebutuhan sistem yang diperlukan dalam penelitian ini meliputi data citra digital buah pisang*,* sumber pustaka yang berkaitan dengan metode *Multi Support Vector Machine*, Ekstraksi Fiturdan beberapa teori lainnya terkait pembuatan sistem pada Python. Adapun pengumpulan data terkait kebutuhan sistem tersebut telah diuraikan pada sub bab 3.1. Hasil dari tahapan ini berupa kebutuhan sistem dan analisisnya akan digunakan pada tahapan selanjutnya, yaitu *elaboration*.

### 3.2.2 *Elaboration*

Pada tahap *elaboration* dilakukan proses analisis dan perancangan sistem yang akan dibuat. Proses analisis dan perancangan sistem ini meliputi:

1. Perancangan *Unified Modeling Languange* (UML) yang terdiri dari *usecase diagram, activity diagram* dan *sequence diagram*.
2. Perancangan antarmuka sistem yang meliputi perancangan antarmuka menu utama dan perancangan antarmuka menu klasifikasi.

### 3.2.3 *Construction*

Proses yang dilakukan dalam tahap *construction* ini adalah membangun sistem Identifikasi Tingkat Kematangan Buah Pisang Menggunakan Metode *Multi Support Vector Mechine* secara utuh, mulai dari *interface* (antarmuka) sistem sampai implementasi hasil perancangan menggunakan Python 3.7.0. Implementasi ini meliputi proses *input* citra buah pisang, *resize* citra, ekstraksi fitur dengan konversi citra RGB ke HSV, metode *Masking Image* dan *Blob Detection*, klasifikasi pada citra uji menggunakan metode *Multi Support Vector Mechine*.

### 3.2.4 *Transition*

Pada tahap *transition* dilakukan proses pengujian terhadap sistem Identifikasi Tingkat Kematangan Buah Pisang untuk mengetahui jika sistem yang dibangun telah sesuai dengan perancangan yang dibuat pada tahap *elaboration* dan sistem telah bisa melakukan klasifikasi tingkat kematangan buah pisang dengan benar. Selain itu, pengujian sistem juga dilakukan untuk melakukan perbaikan terhadap segala masalah yang muncul selama pengujian.

## 3.3 Waktu Penelitian

Penelitian ini dilakukan sejak Juni 2019 hingga Desember 2019. Waktu yang diperlukan dalam penelitian ini secara detail ditunjukkan pada Tabel 3.1.

###### Tabel 3.1. Waktu Penelitian

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Uraian** |  |  | **Tahun (2019)** | | |  |  |
| **Jun** | **Jul** | **Agt** | **Sep** | **Okt** | **Nov** | **Des** |
| 1. | *Inception* |  |  |  |  |  |  |  |
| 2. | *Elaboration* |  |  |  |  |  |  |  |
| 3. | *Constraction* |  |  |  |  |  |  |  |
| 4. | *Transition* |  |  |  |  |  |  |  |

## 3.4 Analisis Permasalahan

Salah satu permasalahan pascapanen pada buah pisang yang diproduksi secara skala besar atau industri adalah dalam hal penyortiran buah pisang. Selama ini buah pisang diidentifikasi tingkat kematangannya berdasarkan analisis warna kulit buah secara visual mata manusia. Proses identifikasi seperti ini memiliki beberapa kelemahan di antaranya yaitu membutuhkan tenaga lebih banyak untuk memilah, dan tingkat persepsi kematangan buah yang dihasilkan bisa berbeda karena manusia dapat mengalami kelelahan, tidak selalu konsiten, dan penilaian manusia juga bersifat subjektif. Kelemahan-kelemahan tersebut akan mempengaruhi waktu yang dibutuhkan dalam memilah dan mengidentifikasi tingkat kematangan buah pisang. Dengan demikian, dibutuhkan alat bantu yang dapat mengidentifikasi tingkat kematangan buah pisang secara tepat.

## 3.5 Rancangan Sistem

Pada aplikasi ini terdapat 2 menu utama yaitu menu ekstraksi fitur dan menu proses uji. Pada menu proses uji diimplementasikan metode *Multi Support Vector Machine* (M-SVM) untuk mengklasifikasi citra kulit buah pisang. Adapun perancangan proses terbagi atas 2 yaitu perancangan proses citra latih dan perancangan proses citra uji yang ditunjukkan pada Gambar 4.1.

31

Mulai

Pilih Menu Ektraksi

Fitur

Klasifikasi

*Multi Support*

*Vector M*

*a*

*chine*

Database

Mulai

Masukan

Citra Uji

Hasil Identifikasi

Selesai

Masukan

Citra Latih

Ektraksi FItur

Pilih Menu Proses Uji

Masukan Data Latih

###### Gambar 3.1 Gambaran umum sistem

Pada gambaran umum terdapat dua menu utama yaitu data latih dan data uji. Pada data latih memilih menu ektraksi fitur, masukan citra latih, ektraksi fitur, klasifikasi data latih menggunakan metode *multi support vector machine*, dan data latih disimpan ke *database* dalam *file* .csv. Pada data uji memili menu proses Uji, masukan citra latih, masukan citra uji, proses citra uji, menampilkan hasil identifikasi dan sistem selesai.

 Masukan Citra

Pada masukan citra, digunakan dataset citra yang telah diakuisisi sebelumnya. Dataset diakuisisi dari buah pisang yang memiliki tingkat kematangan yang berbeda-beda. Dataset yang diperolah sebanyak 160 citra dan diakuisisi pada siang hari. Citra diakuisisi menggunakan kamera *smartphone* dengan jarak dari pisang ke kamera sebesar 30 cm. adapun proses masukan citra dapat dilihat pada Gambar 4.2 berikut:



Mulai



Masukan Citra



Buah Pisang



300



X



225



Selesai



Tampil



###### Gambar 3.2 *Flowchart* masukan citra buah pisang

Sistem mulai menekan tombol masukan citra buah pisang, citra ditampilkan dan sistem selesai.

 Ektraksi Fitur

Pada tahap ektraksi fitur dilakukan proses segmentasi untuk memisahkan *background* dan objek. Tahapan yang dilakukan pada proses ekstraksi citra ditunjukkan pada Gambar 4.3 berikut:



Mulai



Konversi ke Citra HSV



Citra Buah Pisang



RGB



300



X



225



Konversi ke Citra Mask



Deteksi Citra Buah Pisang



Selesai



Crop Citra Deteksi



###### Gambar 3.3 *Flowchart* Ektraksi Fitur

Citra buah pisang RGB akan dikonversi ke citra HSV, konversi ke citra *masking image,* deteksi area buah pisang dan crop citra buah pisang.

 Identifikasi Buah

Tahapan terakhir penelitian ini adalah proses identifikasi citra buah. Proses identifikasi menggunakan data *training* yang di klasifikasi menggunankan metode *Multi Support Vector Machine* dengan tujuan menghasilkan satu model yang akan digunakan sebagai penentuan kematangan buah pisang. Metode yang digunakan untuk proses identifikasi adalah hitung menggunakan metode *Multi Support Vector Machine*. Tahapan yang dilakukkan pada proses identifikasi daun ditunjukkan pada Gambar 4.4 berikut:

Mulai

Mulai

Nilai Fitur Uji

Nilai Fitur Latih

Hitung nilai fitur

menggunakan metode

*Multi*

*Support Vector M*

*a*

*chine*

Simpan ke

Database

Hitung nilai dengan

menggunakan

*Multi Support*

*Vector M*

*a*

*chine*

Hasil Identifikasi

Selesai

###### Gambar 3.4 *Flowchart* Identifikasi Citra

Nilai fitur latih akan dihitung menggunakan metode *multi support vector machine* dan simpan ke *database.* Nilai fitur uji akan dibandingkan dengan nilai fitur latih yang telah tersimpang ke *database* dengan menggunakan metode *multi support vector machine*, dan sistem akan menampilkan hasil identifikasi.

## 3.5 Perancangan Proses Citra Latih

Dalam penelitian ini, untuk merancang bangun proses citra latih menggunakan metode *Multi Support Vector Machine* (M-SVM)dalam mengidentifikasi tingkat kematangan berdasarkan kulit buah pisang. Proses utama yaitu *input* citra*,* ekstraksi fitur, dan simpan fitur hasil segmentasi kedalam *file* .csv, fitur tersebuat akan digunakan dalam proses pengidentifikasian menggunakan metode *Multi Support Vector Machine* (M-SVM). Citra latih diperoleh dari proses akuisisi citra.

## 3.6 Perancangan Proses Citra Uji

Perancangan proses citra uji merupakan proses yang bertujuan untuk mengidentifikasi tingkat kematangan citra kulit buah pisang. Tahap awal pada citra uji, yaitu *input* citra, *resize* citra dengan ukuran 300 x 225 piksel, selanjutnya konversi citra RGB ke citra HSV, dan ekstraksi fitur yang terdiri dari proses segmentasi menggunakan metode thresholding, konversi ke citra RGB. Hasil akhir dari proses ekstraksi fitur akan disimpan dalam *database* yang nantinya akan digunakan pada proses pengujian.

#### 3.6.1 *Resize*

*Resize* merupakan proses mengubah dan menyamakan citra masukan, dengan mengubah resolusi horizontal dan vertikal citra masukan tersebut. Kinerja sistem akan melambat apabila terdapat keanekaragaman ukuran citra masukan. Saat mengubah ukuran gambar, perlu mempertahankan rasio agar gambar tidak terdistorsi. Misalnya ketika menambah atau mengurangi lebar gambar, ukuran tinggi juga perlu disesuaikan untuk memastikan rasio gambar tetap terjaga.

#### 3.6.2 Segmentasi Objek

Segmentasi objek merupakan proses pemisahan antara *background* dan objek pada citra. Hal ini dilakukan agar proses selanjutnya akan mendapatkan hasil yang lebih baik. Adapun ilustrasi proses segmentasi objek menggunakan metode *thresholding* dapat dilihat pada sub bab 2.4.2.

#### 3.6.3 Konversi citra RGB ke HSV

Proses segmentasi warna**,** salah satunya dapat dilakukan dengan cara mengkonversi ruang warnacitra yang semula RGB (*Red, Green, Blue*) menjadi ruang warna HSV (*Hue, Saturation, Value*)**.** Komponen *Hue*merupakan komponen yang merepresentasikan warna dari berbagai panjang gelombang cahaya**.**

Untuk mentransformasi dari RGB ke HSV. Diasumsikan koordinatkoordinat R, G, B [0,1] adalah berurutan merah, hijau, biru dalam ruang warna RGB, dengan max adalah nilai maksimum dari nilai *red, green, blue*, dan min adalah nilai minimum dari *nilai red, green, blue.*  Untuk memperoleh sudut *hue*[0,360] yang tepat untuk ruang warna HSV (Swedia dan Cahyanti, 2010). Adapun persamaan yang digunakan dalam konversi RGB ke HSV dapat dilihat pada sub bab 2.4.1.

Berikut langkah-langkah konversi nilai RGB ke HSV dari citra matriks 2x2:

1. Menentukan citra RGB yang menjadi objek deteksi. Misal: Nilai R = 22, G = 24, B = 23
2. Mencari nilai maksimal dan minimal dari nilai R, G dan B



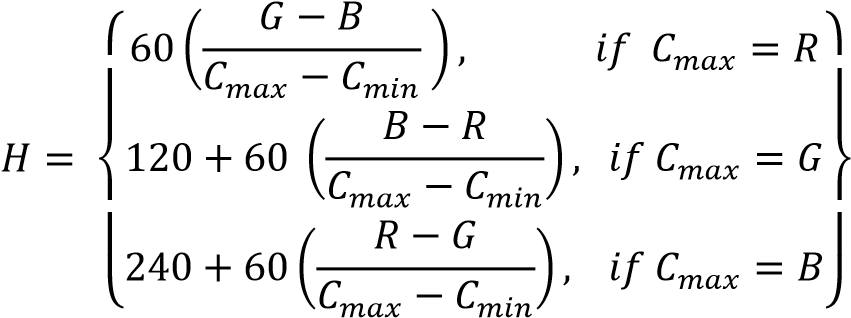


(4.1)

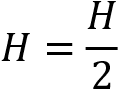
Sehingga



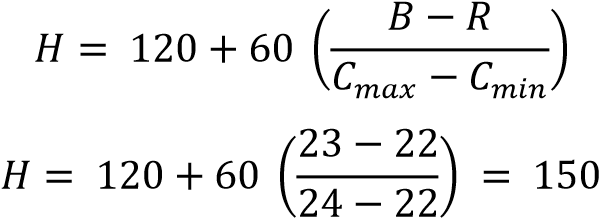
1. Menentukan nilai *H*

 (4.2)

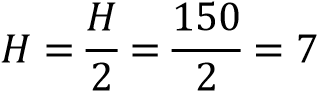
Jika tipe data 8 bit maka:

 (4.3)

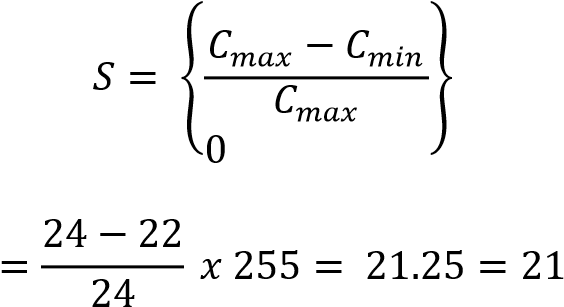
Karena, maka:



Karena tipe data Gambar yang digunakan 8 bit maka:

 (4.4)

1. Menentukan nilai *S*



1. Menentukan nilai *V*

𝑉 = 𝐶𝑚𝑎𝑥 (4.5)

𝑉 = 24

Sehingga diperoleh nilai HSV 75, 21 dan 24 dari nilai RGB 22, 24, dan 23. Contoh hasil konversi nilai RGB ke HSV menggunakan Gambar 8 bit dengan matriks 2x2 ditunjukkan pada Gambar 4.5.

RGB HSV

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | 22 24 23 | 23 25 23 | | 27 29 26 | 31 33 32 | |  | |  |  | | --- | --- | | 75 21 24 | 60 20 25 | | 50 26 29 | 75 15 33 | |

###### Gambar 4.5 Nilai matriks hasil konversi RGB ke HSV

Contoh citra pisang yang telah dikonversi ke HSV ditunjukkan pada:



###### Gambar 4.6 Citra buah pisang hasil dari konversi ke HSV

#### 3.6.4 *Masking Image*

Pada bagian ini, citra HSV dikonversi ke citra hitam putih melalui proses *masking*. Hasil proses *masking* memiliki dua nilai piksel yaitu 0 dan 255. Nilai *threshold* digunakan dengan mengambil rentang nilai *lower* dan *upper* yang optimal dari HSV. Berdasarkan hasil eksperimen yang dilakukan pada kondisi pisang belum matang, cukup matang, matang dan matang sekali maka nilai *lower* dan *upper* yang optimal dari HSV ditunjukkan pada Tabel 4.1.

###### Tabel 4.1. *Range* nilai HSV

|  |  |  |  |
| --- | --- | --- | --- |
| *Value* | *H* | *S* | *V* |
| Minimal | 0 | 48 | 33 |
| Maksimal | 77 | 255 | 212 |

Konversi citra HSV ke citra hitam putih menggunakan *threshold* pada Tabel 4.1 dengan matriks 5x5 ditunjukkan pada Gambar 4.7.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ***MxN*** | **1** | **2** | **3** | **4** | **5** |
| **1** | 0 0 13 | 2 255 46 | 33 92 75 | 37 110 58 | 35 111 30 |
| **2** | 3 226 62 | 2 229 88 | 4 212 88 | 20 174 69 | 15 245 25 |
| **3** | 3 207 79 | 4 183 110 | 3 146 124 | 7 169 119 | 21 127 76 |
| **4** | 4 196 95 | 2 213 102 | 2 226 96 | 7 212 107 | 4 255 55 |
| **5** | 6 198 90 | 2 188 103 | 10 175 92 | 6 201 75 | 0 0 20 |



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *MxN* | 1 | 2 | 3 | 4 | 5 |
| 1 | 0 | 255 | 255 | 255 | 0 |
| 2 | 255 | 255 | 255 | 255 | 0 |
| 3 | 255 | 255 | 255 | 255 | 255 |
| 4 | 255 | 255 | 255 | 255 | 255 |
| 5 | 255 | 255 | 255 | 255 | 0 |

**Gambar 4.7 Nilai matriks konversi citra HSV ke citra hitam putih**  Contoh citra pisang proses *masking* ditunjukkan ditunjukkan pada Gambar 4.8 berikut.



###### Gambar 4.8 Citra buah pisang hasil masking

#### 3.6.5 *Blob detection*

Proses selanjutnya adalah menganalisis luas area objek untuk dilakukan *bounding box*. *Blob detection* akan menganalisis luas area dan bentuk objek *blob* dari suatu citra yang menjadi fokus deteksi. Hasil *find contour* dan *bounding box* ditunjukkan pada Gambar 4.9.



###### Gambar 4.9.Citra buah pisang hasil crop deteksi citra

#### 4.4.6 *Multi Support Vector Mechine* (M-SVM)

Pada proses klasifikasi digunakan metode *Multi Support Vector Machine* (M-SVM). Metode tersebut digunakan untuk mengklasifikasikan tingkat kematangan buah pisang. RBF merupakan kernel yang akan digunakan pada proses klasifikasi. Berikut ini merupakan contoh perhitungan *Multi Support Vector Machine* (M-SVM) dengan menggunakan kernel RBF.

*Output* dari proses *training* adalah *alpha* (α) dan nilai bias yang akan digunakan untuk proses uji. Berikut alur sistem dari proses pelatihan SVM.

**Step1.** Data fitur latih, data target latih, *C*, dan Gamma () di gunakan sebagai masukan. Penentuan nilai awal *C* = 1 dan = 10-1.

**Step 2.** Menghitung fungsi kernel RBF menggunakan persamaan berikut:

 4.6

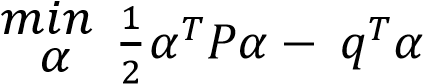
Dimana:

*x* = Data fitur latih

 = Gamma

**Step 3.** Pada tahap ini data masukan akan digunakan untuk mencari nilai α optimal

menggunakan persamaan *Quadratic Programming* pada persamaan 4.7



(4.7)



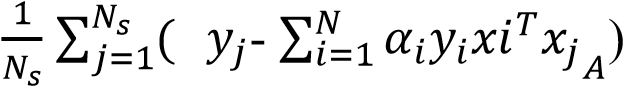
Dimana:

*P* =  dimana *k* = 2 ).

*q* = vector dengan ukuran *N\*1* (dimana *N* = jumlah data latih) *A* = Target *b* = 0

*G* = Batas bawah nilai , *G*= 0 *h* = Batas atas nilai , *h* = C

**Step 4.** Setelah nilai  optimal didapatkan, selanjutnya mencari nilai bias optimal untuk digunakan pada proses uji dengan persamaan 4.8

b =  (4.8)

= target dari *support* vector ke *j* (dimana *j* = 1...*Ns*)

= Fitur dari *support vector*

 = Fitur dari data latih

= optimal yang didapatkan dari *Quadratic Programming*

= Jumlah *support vector*

*N* = Jumlah data latih

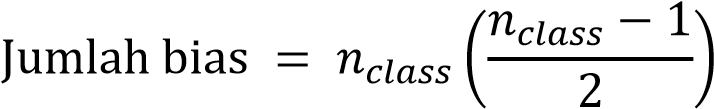
= Target dari data latih

*Output* dari nilai optimal bias merupakan model pelatihan SVM yang akan digunakan pada proses uji. Berikut nilai bias hasil proses *training*.

###### Tabel 4.2. Nilai bias

|  |  |
| --- | --- |
| **No.** | **Bias** |
| 1. | 0,57896412 |
| 2**.** | 0,51618445 |
| 3. | 0,2364117 |
| 4. | 2,65760561 |
| 5. | 2,4659829 |
| 6. | 1,38860911 |

Terdapat 6 nilai bias yang diperoleh dari proses *training* pada Tabel 3.2. Hal ini disebabkan karena model SVM yang digunakan adalah *multi-*class *OAA* (*oneagainst-all*) dan jumlah kelas yang digunakan ada 4. Berikut persamaan dalam penentuan jumlah bias yang digunakan:

 (4.9)

**Step 5.** Model SVM dari hasil *training* yaitu nilai optimal dari  dan bias akan disimpan untuk digunakan pada proses uji.

**Step 6.** Proses nilai selanjutnya yaitu *C* =2 sampai *C*dan  = 10-1 sampai 

10-5.

**Step 7.** Analisis model dari SVM diperoleh nilai *C* dan  yang optimal.

## 4.5 Perancangan *Unified Modeling Languange* (UML)

### 4.5.1 *Usecase Diagram*

Gambar 4.10 merupakan gambar *usecase* diagram yaitu gambar yang mendeskripsikan apa yang seharusnya dilakukan oleh sistem. Terdapat satu aktor pada Gambar 4.10 yaitu *User* yang merupakan pengguna sistem. Pada sistem ini *user* dapat mengelola data masukan, ektraksi fiturdan klasifikasi.

User

Pilih Menu

Ektraksi Fitur

Klasifikasi dengan

*Multi*

*Support Vector Mechine*

Proses Uji

Ektraksi Fitur

Konversi citra RGB ke citra HSV

<<

include

>>

<<

include

>>

Konversi citra HSV ke citra mask

Blob detection

<<

include

>>

<<

include

>>

Proses Uji

<<

include

>>

Crop citra

<<

include

>>

Gambar 4.10 *Usecase Diagram*

Sistem pada penelitian ini menyediakan tiga proses utama yaitu *preprocessing,* ekstraksi fitur, dan proses uji. Menu masukan citra digunakan untuk memasukkan citra dari folder citra ke dalam aplikasi. Menu ekstraksi fitur terdiri konversi citra RGB ke citra HSV, mask image, blob detection.Dan menu proses uji yang terdiri atas masukancitra uji, sertaidentifikasi menggunakan metode *Multi Support Vector Machine*.

### 4.5.2 *Activity Diagram*

##### 4.5.2.1 *Activity* *Diagram* Masukkan Citra

Gambar 4.11 merupakan aktivitas untuk memasukkan citra yang akan diproses ke dalam aplikasi. Untuk memasukkan citra, sebelumnya *user* memilih menu Masukkan Citra kemudian sistem akan meminta *user* untuk memilih citra yang akan dimasukkan ke dalam sistem.

Activity

:

masukan citra

Sistem

User

Phase

Menekan tombol

masukan citra

Meminta user untuk

memilih citra yang

akan di masukan

Melakukan dan

menampilkan hasil

citra resize

###### Gambar 4.11. *Activity* *Diagram* Masukkan Citra

##### 4.5.2.2 *Activity* *Diagram* Ekstraksi Fitur

Gambar 4.12 merupakan aktivitas untuk melakukan ekstraksi fitur citra. Untuk melakukan tahapan ekstraksi fitur, sebelumnya *user* memilih menu Proses, kemudian sistem akan menampilkan hasil ekstraksi fitur.

Activity

:

ektraksi fitur

Sistem

User

Phase

Menekan tombol

masukan proses

menampilkan hasil

citra RGB

menampilkan hasil

blob detection dan

crop citra hasi

deteksi

menampilkan hasil

konversi citra RGB

ke citra HSV

menampilkan hasil

masking image

###### Gambar 4.12 Activity Diagram Ekstraksi Fitur

##### 4.5.2.3 *Activity* *Diagram* Simpan Fitur

Gambar 4.13 merupakan aktivitas untuk menyimpan nilai fitur citra. Untuk melakukan tahapan simpan fitur, sebelumnya *user* memilih menu proses, kemudian sistem akan menyimpan nilai fitur dalam *file* .csv.

ActIvity

:

Simpan Fitur

Sistem

User

Phase

Menekan tombol

simpan fitur

Melakukan proses

simpan fitur

###### Gambar 4.13 *Activity* *diagram* simpan fitur

##### 4.5.2.4 *Activity* *Diagram* Proses Uji

Gambar 4.14 merupakan aktivitas untuk melakukan identifikasi terhadap citra uji. Untuk melakukan tahapan identifikasi, sebelumnya *user* memilih menu Identifikasi, kemudian sistem akan menampilkan *form* Identifikasi, lalu *user* memilih tombol masukkan citra uji, maka sistem akan menampilkan citra uji yang telah dipilih, setelah itu *user* memilih tombol identifikasi, maka sistem akan menampilkan hasil identifikasi.

ActIvity

:

identifikasi

Sistem

User

Phase

Menekan tombol

Proses Uji

Menampilkan form

proses uji

Menekan tombol

pilih pada Path CSV

Menampilkan

database yang telah

dipilih

Menekan tombol

pilih pada Path

Gambar

Menampilkan data

uji yang di pilih

Menekan tombol

Proses

Menampilkan hasil

identifikasi citra

###### Gambar 4.14 *Activity* *diagram* identifikasi

### 4.5.3 *Sequence Diagram*

##### 4.5.3.1 *Sequance* *Diagram* Masukkan Citra

Gambar 4.15 merupakan *sequence* *diagram* masukan citra pada sistem Implementasi Metode *Multi Support Vector Machine* Untuk Menentukan Tingkat Kematangan Buah Pisang Raja dan Pisang Bugis.

**User**

Antarmuka

Axes citra asli

1

.

Pilih menu masukan citra

2

.

Menampilkan file dan lokasi citra

3

.

Memilih lokasi dan file citra

4

.

Menampilkan citra yang telah dipilih dan telah ter

-

resize

###### Gambar 4.15 Sequence diagram masukkan citra

User memilih menu masukan citra, sistem menampilkan file dan lokasi citra, user memilih lokasi dan file citra dan sistem menampilkan citra latih yang telah dipilih.

##### 4.5.3.2 *Sequence Diagram* Ektraksi Fitur

Gambar 4.16 merupakan *sequence* *diagram* ekstraksi fitur pada sistem Implementasi Metode *Multi Support Vector Machine* Untuk Menentukan Tingkat Kematangan Buah Pisang Raja dan Pisang Bugis.

User

Antarmuka

Axes Citra RGB

Axes

citra HSV

Axes citra mask

1

.

Pilih menu proses

2

.

Menampilkan citra RGB

3

.

Menampilkan hasil konversi citra ke HSV

4

.

Menampilkan hasil citra mask

Axes deteksi

citra

5

.

Menampilkan hasil citra deteksi pisang

Axes crop citra

6

.

Menampilkan hasil citra crop

###### Gambar 4.16 Sequence diagram ekstraksi fitur

User memilih menu proses, sistem menampilkan citra RGB, menampilkan hasil konversi citra ke HSV, menampilkan hasil citra mask, menampilkan citra deteksi pisang dan menampilkan hasil crop citra.

##### 4.5.3.3 S*equance* *Diagram* Simpan Fitur

Gambar 4.17 merupakan *sequence* *diagram* simpan fitur pada sistem

Implementasi Metode *Multi Support Vector Machine* Untuk Menentukan Tingkat

Kematangan Buah Pisang Raja dan Pisang Bugis

User

Antarmuka

Sistem

2

.

Menampilkan fitur dari hasil ektraksi fitur

1

.

Pilih menu proses

###### Gambar 4.17 *Sequence diagram* simpan fitur

User memilih menu proses dan sistem menampilkan fitur dan hasil ektraksi fitur.

## 4.6 Rancangan Penyimpanan Data

Penyimpanan data pada sistem ini menggunakan format .csv pada Python. File dengan ekstensi .csv adalah *file* yang berada dalam format wadah data biner yang digunakan oleh program Python. Ekstensi dikembangkan oleh Anaconda dan *file* CSV dikategorikan sebagai *file* data yang mencakup variabel, fungsi, *array*, dan informasi lainnya.

## 4.7 Perancangan Antarmuka Menu Utama

Pada perancangan antarmuka Sistem identifikasi tingkat kematangan buah yang ditunjukkan Gambar 4.18, terdapat tombol ‘Masukkan citra’, ‘Resize’, ‘Ekstraksi’, ’Simpan Database’ dan ‘Pengujian’. Pada tombol ‘Pilih’, citra buah pisang yang kita pilih akan tampil pada axes. Jika tombol proses dipilih, akan ditampilkan 5 proses ekstraksi fitur adalah fitur yang akan digunakan dalam proses identifikasi dan citra hasil ekstraksi fitur akan disimpan ke dalam database dan akan melalu proses klasifikasi menggunakan metode Multi Support Vector Machine

(MSVM). Dan jika memilih tombol ‘Ektraksi Fitur’, maka akan menampilkan form Ektraksi Fitur.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Form Pelatihan**   |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | |  |  |  |  | | --- | --- | --- | --- | | |  | | --- | | Hasil citra label | | |  | | --- | | **Pilih** | |   Pilih Citra | | | | | | | | | | | | | **Proses** | |  | **Restar** | | | | Hasil citra RGB |  | Hasil citra HSV |  | | Hasil masking image | | |  | Hasil Deteksi Citra | |  | Hasil crop | | |

###### Gambar 4.18 Antarmuka Menu Utama

## 4.8 Perancangan Antarmuka Menu Identifikasi

Peracangan antarmuka tombol ‘Proses Uji’ ditunjukkan pada Gambar 4.19. Pada *form* Identifikasi akan ditampilkan hasil klasifikasi citra uji buah pisang, hasil klasifikasi akan berupa tingkat kematangan buah pisang.

**Form Pelatihan**

Pilih Citra Latih

Pilih Citra Uji

**Pilih**

**Proses**

Hasil pencarian citra latih

**Restar**

Hasil Deteksi

Hasil pencarian citra uji

**Pilih**

###### Gambar 4.19 Antarmuka Menu Klasifikasi

## 4.9 Perhitungan Akurasi

Perhitungan akurasi digunakan untuk mengetahui sejauh mana sistem dapat menguji apakah metode yang digunakan sudah cukup baik. Perhitungan akurasi pada sistem Identifikasi Tingkat Kematangan Buah Pisang menggunakan metode *Multi Support Vector Machine* (M-SVM) menggunakan rumus pada Persamaan (4.9) (Achban, 2017).

% keakuratan = 𝑇𝑜𝑡𝑎𝑙 𝑃𝑒𝑛𝑔𝑒𝑛𝑎𝑙𝑎𝑛 𝑦𝑎𝑛𝑔 𝐵𝑒𝑛𝑎𝑟 x 100% (4.9)

𝑇𝑜𝑡𝑎𝑙 𝐶𝑖𝑡𝑟𝑎 𝑈𝑗𝑖

###### DAFTAR PUSTAKA

Achban, A. (2017). *Pengenal Dorsal Hand Vein Menggunakan Local Binary Pattern.* Kendari:Fakultas Teknik, Universitas Halu Oleo.

Adaptive Vision. n.d. “*Blob* Analysis.” [Online]. Available:

http://docs.adaptive.vision.com/current/studio/machine\_vision\_guide/*Blob* Analysis.html. [Accessed: 04-May-2019].

Agaputra, D. M., Wardani, R. R. K., Siswanto, E., 2013, *Pencarian Citra Digital Berbasiskan Konten dengan Ektraksi Fitur HSV, ACD, dan GLCM*. Institut Teknologi Harapan Bangsa, Telematika, Vol. 8, No. 2.

Chamidah, N. (2018). K-Means Sebagai Ekstraktor Ciri Pada Klasifikasi Data

Dengan Algoritma Support Vector Machine (Svm). *Simetris: Jurnal Teknik*

*Mesin, Elektro Dan Ilmu Komputer*, *9*(2), 889–896.

<https://doi.org/10.24176/simet.v9i2.2433>

Chaniago, D., Hidayat, B., Wibowo, S.A. 2011, *Klasifikasi Buah Pisang Berdasarkan Jenis Dan Kematangan Berbasis Pengolahan Citra Dengan*

*Kamera Digital Classification Of Banana Fruit Based On Type And Ripeness Using Image Pr ocessing With A Digital Camera*. Universitas Telkom, Teknik Komunikasi, Fakulas Teknik Elektro.

Community. (2012). *Index Lengkap Syntax.* Yogyakarta: MediaKom.

Gonzales, R. C., dan Woods, R. E., 2002, *Digital Image Processing Second edition. New Jersey: Prentice Hall.*

Gonzalez, R. C., dan Woods, R. E., 2008, *Digital Image Processing Third Edition*, New Jersey: Pearson Education,Inc.

Harianto, A.D., 2017. *Prototipe Pemilah Kematangan Buah Pisang Berdasarkan Warnanya Menggunakan Conveyor*. Yogyakarta: Universitas Sanata Dharma, Teknik Elektro.

Hindhayani, T. (2017). Identifikasi Penulisan Melalui Pola Tulisan Tangan Menggunakan Algoritma Support Vector Machine. Jurnal Muara: Sains Teknologi, Kedokteran, dan Ilmu Kesehatan, Vol. 1, No. 1: hlm 210-217.

Ilmiah Teknologi Sistem Informasi 3 (1) 17-23.

Kader, AA. 2018. Kader AA. 2008. *Maturity indices - banana ripening chart* [Internet]. [diunduh 2013 Jan 20]. Tersedia pada: [http://postharvest.](http://postharvest/) ucdavis. edu/ Produce/ProduceFacts/Fruit/banana.shtml.

Kusuma, S. F., Pawening, R. E., & Dijaya R. (2017) , Otomatisasi Klasifikasi Kematangan Buah Mengkudu Berdasarkan Warna dan Tekstur. Register:

Jurnal

Kusumanto, RD., dan Tompunu, A. N., 2011, *Pengolahan Citra Digital Untuk Mendeteksi Obyek Menggunakan Pengolahan Warna Model Normalisasi Rgb*. *Seminar Nasional Teknologi Informasi & Komunikasi Terapan*.

Semarang.

Limin, nina, Sari, J., & Purnama, I. P. (2019). *Identifikasi Tingkat Kematangan Buah Pisang Menggunakan Metode Ektraksi Ciri Statistik Pada Warna Kulit Buah*. Ultimatics : Jurnal Teknik Informatika, 10 (2), 98-102.

Lingras, P., & Butz, C. (2007). Rough set based 1-v-1 and 1-v-r approaches to support vector machine multi-classification. *Information Sciences*, *177*(18),

3782–3798. <https://doi.org/10.1016/j.ins.2007.03.028>

Ma, Y., & Guo, G. (2014). Support vector machines applications. In *Support Vector*  *Machines Applications* (Vol. 9783319023007). https://doi.org/10.1007/978-3-319-02300-7

Munandar, H., 2017, *Identifikasi Landmark Orbital Cephalometry Menggunakan Metode Fuzzy C-Means Clustering*. Kendari: Fakultas Teknik, Universitas Halu Oleo.

Muchtar, M. (2015). *Penggabungan Fitur Dimensi Fraktal dan Lacunarity untuk Klasifikasi Daun*.

Nugroho, A. S., Witarto, A. B., & Handoko, D. (2003). *Support Vector Machine*.

Putra, D., 2010, *Pengolahan Citra Digital*, C. V Andi Offset, Yogyakarta.

Prabawati, S., Suyanti dan Setyabudi, D.A. (2008). *Teknologi Pascapanen dan*

*Pengolahan Buah Pisang. Balai Besar Penelitian dan Pengembangan*

*Pascapanen Pertanian.* Dalam seminar Badan Litbang Pertanian. Departemen Pertanian, Bogor

Qian, R., Wu, Y., Duan, X., Kong, G., & Long, H. (2018). SVM multi-classification optimization research based on multi-chromosome genetic algorithm. *International Journal of Performability Engineering*, *14*(4), 631–638. https://doi.org/10.23940/ijpe.18.04.p5.631638

Rahmat, Ericks, Swedia., Cahyanti, Margi, 2010. *Algoritma Transformasi Ruang Warna*.

Rizkika K. 2012. *Kulit Pisang Pemurni Air*. Jakarta (ID): PT. Trubus Swadaya.

Sugianti, Sigit., Wibowo, Feri., 2015. *Klasifikasi Tingkat Kematangan Buah Pepaya (Carica Papaya L) California (Calina-IPB 9) dalam ruang warna HSV dalam Algoritma K-Nearest Neightbors*. Purwokerto: Universitas Muhammadiah Purwokerto, Teknik Informatika.

Suhartanto, R., Sobir, Heri H., 2012. *Teknologi Sehat Budidaya Pisang: dari Benih sampai Pasca Panen*. Bogor: Pusat Kajian Holtikultura Tropika LPPM-IPB.

Sugianti, Sigit., Wibowo, Feri., 2015. *Klasifikasi Tingkat Kematangan Buah Pepaya (Carica Papaya L) California (Calina-IPB 9) dalam ruang warna HSV dalam Algoritma K-Nearest Neightbors*. Purwokerto: Universitas Muhammadiah Purwokerto, Teknik Informatika.

Sutowijoyo, Danang., 2013. *Kriteria Kematangan Pasca Panen Pisang Raja Bulu dan Pisang Kepok*. Bogor: Institut Pertanian Bogor.

Sembiring, K. (2007). *Machine untuk Pendeteksian Intrusi pada Jaringan Latar*

*Belakang*. (September 2007)

Sukamto, R. A., & Shalahudin, M. (2013). *Rekayasa Perangkat Lunak.* Bandung:

Informatika.