

Pengenalan Tingkat Kematangan Tomat Berdasarkan Citra Warna Pada Studi Kasus Pembangunan Sistem Pemilihan Otomatis

Mochamad Angga Anggriawan^{#1}, Muhammad Ichwan^{*2}, Dina Budhi Utami^{#3}

[#]Teknik Informatika, Institut Teknologi Nasional Bandung
Jalan P.H.H. Mustafa No.23 Bandung

¹1927angga@gmail.com

³dinabusoft@gmail.com

^{*}Teknik Informatika, Institut Teknologi Nasional Bandung
Jalan P.H.H. Mustafa No.23 Bandung

²ichwan@itenas.ac.id

Abstract — Tomatoes sorting process by humans relatively less consistent in determining the ripe of tomato and unripe of tomato. This is due to several factors like human fatigue and difference perception of each person. To solve this problem, this research proposes a computer based system which can do the sorting process that more consistent level of accuracy than by human. The sorting process of this system involve several phases like tomatoes image acquisition followed by cropping image process, HSV color image conversion and hitogram HSV. Results HSV color conversion as a classification input to determine ripeness level of tomatoes into ripe and unripe categories using Learning Vector Quantization (LVQ) method. The system is built by several supporting components such as camera as an image acquisition and Belt Conveyor as a material transfer. The result of the ripeness level recognition system test based on the color image with Learning Vector Quantization (LVQ) method was obtained the accuracy percentage of 83,75% from 80 with tomato image acquisition done statically and 83,33% from 48 with tomato image acquisition on belt conveyor.

Keywords — Belt Conveyor, Camera, LVQ, Ripeness Level, Tomatoes.

I. PENDAHULUAN

Tingkat kematangan buah dapat dilihat dari beberapa aspek yaitu dilihat dari warna, dilihat dari bentuk dan dilihat dari aroma buah tersebut. Pada aspek warna, kematangan buah dapat dilihat dengan menyesuaikan warna buah yang telah matang dan buah yang akan diuji tingkat kematangannya [1]. Namun, proses klasifikasi tingkat kematangan tersebut juga dapat menyebabkan ketidakseragaman yang disebabkan oleh beberapa faktor seperti kelelahan yang dialami manusia, perbedaan persepsi antara yang satu dengan yang lainnya, dan keragaman visual manusia [2]. Maka untuk memeriksa tingkat kematangan buah, perlu adanya suatu sistem yang dapat mengenali buah sesuai dua kategori yaitu telah matang serta belum matang

dan melakukan pemilihannya secara otomatis dengan *automated sorting system* [3].

Berdasarkan permasalahan tersebut muncul suatu ide dalam membuat suatu sistem yang dapat memisahkan antara buah yang sudah matang dan belum matang dengan memanfaatkan citra warna dari buah tersebut. Buah tersebut berupa tomat yang akan dikenali melalui tahapan *image pre-processing* dengan meletakkan tomat tersebut pada *Belt Conveyor*. Tahapan *image processing* akan melakukan penyesuaian warna dari tomat yang telah matang terhadap tomat yang akan diuji tingkat kematangannya [4]. Sistem yang dibuat juga memiliki sistem pengontrol yang akan memisahkan tomat yang telah matang dengan tomat yang belum matang pada suatu wadah yang berbeda. Sistem yang dibangun diharapkan dapat menjadi solusi untuk mengenali tingkat kematangan tomat dan dapat di terapkan pada kehidupan bermasyarakat [5].

Sistem yang dibangun memiliki tiga subsistem yang dikerjakan oleh orang yang berbeda. Subsistem tersebut diantaranya mengenai proses pengenalan tingkat kematangan tomat berdasarkan citra warna, pengenalan ukuran tomat berdasarkan citra digital serta proses pemilihan tomat berdasarkan tingkat kematangan dan ukuran tomat. Adapun keterangan lebih lanjut mengenai informasi pembagian sistem yang dibangun direpresentasikan pada Tabel I.

TABEL I
PEMBAGIAN KERJA PER-SUB BAGIAN

Nama Anggota Tim	Bagian	Sistem Besar Yang Dibangun
Mochamad Angga Anggriawan (152013073)	Pengenalan Tingkat Kematangan tomat Berdasarkan Citra Warna	<i>Automated Sorting System</i> pada tomat

TABEL I (LANJUTAN)
PEMBAGIAN KERJA PER-SUB BAGIAN

Nama Anggota Tim	Bagian	Sistem Besar Yang Dibangun
Sandha Rineka Putra (152013061)	Pengenalan Ukuran Tomat Berdasarkan Citra Digital	<i>Automated Sorting System</i> pada tomat
Gian Permana (152013054)	Penyeleksian Tomat Berdasarkan Tingkat Kematangan dan Ukuran	

Fokus subsistem yang dibangun pada penelitian yang dilakukan yaitu mengenai subsistem pengenalan tingkat kematangan tomat berdasarkan citra warna.

A. Rumusan Masalah

Dalam rangka mengurangi permasalahan pada saat memilih tomat yang telah matang dan tomat yang belum matang, perlu adanya suatu sistem yang mampu mengurangi dampak keakurasian pemilihan tingkat kematangan tomat berdasarkan warna tersebut secara otomatis^[3]. Sistem yang dibangun menggunakan bantuan perangkat kamera. Berdasarkan identifikasi yang ditetapkan maka muncul berbagai masalah yang akan ditemui sebagai berikut:

- Bagaimana cara melakukan tahapan pengolahan citra warna terhadap citra tomat yang diambil oleh kamera pada *belt conveyor* secara bergerak.
- Bagaimana cara melakukan klasifikasi terhadap citra warna hasil tahapan *image-processing* kedalam kategori matang dan belum matang.
- Bagaimana cara mengetahui kecepatan conveyor yang optimal untuk mendapatkan informasi mengenai kematangan buah berdasarkan citra warna.
- Bagaimana cara mengetahui tingkat akurasi sistem tersebut untuk mendapatkan informasi mengenai kematangan buah berdasarkan citra warna dengan perbandingan antara citra yang tidak bergerak dan citra yang bergerak.

B. Tujuan

Pada sistem ini, terdapat penelitian dengan tujuan melakukan pengukuran performa sistem pengklasifikasian tomat berdasarkan citra warna untuk mengenali tingkat kematangan pada studi kasus pembangunan sistem pemilihan otomatis.

C. Ruang Lingkup

Penelitian pada sistem pemilihan otomatis untuk mengenali kematangan tomat berdasarkan citra warna memiliki ruang lingkup sebagai berikut :

- Sistem akan mengklasifikasi tingkat kematangan tomat kedalam kategori matang atau belum matang berdasarkan satu paramater yaitu citra warna dari tomat yang akan melalui proses konversi ruang warna HSV (*Hue, Saturation, Value*) dan Histogram.

- Objek penelitian berasal dari satu jenis tomat yaitu varietas 'tw'.
- Membandingkan 2 kamera dengan spesifikasi *Megapixel* berbeda untuk proses pengambilan citra warna agar hasil yang di dapatkan optimal.
- Proses pengambilan citra tomat dilakukan dengan cara manual.
- Proses pengambilan citra warna dari arah tampak atas objek tomat.
- Proses pengambilan citra warna dilakukan dengan memberikan pencahayaan yang tetap.
- Proses pengambilan citra warna dilakukan dengan memberikan latar *solid* berwarna hitam.
- Proses pengambilan citra warna dilakukan dengan meletakkan tomat diatas *Belt Conveyor*.

II. LANDASAN TEORI

A. Ekstraksi Ciri

Ekstraksi ciri merupakan proses pengideksan suatu *database* citra dengan isinya secara matematik, setiap ekstraksi ciri merupakan *encode* dari vektor n dimensi yang disebut dengan vektor ciri [6]. Komponen vektor ciri dihitung dengan pemrosesan citra dan teknik analisis serta digunakan untuk membandingkan citra yang satu dengan citra yang lain.

B. Pemotongan Citra

Pemotongan citra (*cropping citra*) merupakan cara mendapatkan area warna tomat tertentu yang diamati (*area of interest*), yang bertujuan untuk mempermudah penganalisaian citra dan memperkecil ukuran penyimpangan citra [7].

C. Ruang Warna H-S-V

HSV mendefinisikan warna dalam terminologi *Hue, Saturation* dan *Value*. Keuntungan *HSV* adalah terdapat warna-warna yang sama dengan yang ditangkap oleh indra manusia [8]. Sedangkan warna yang dibentuk model lain seperti *RGB* merupakan hasil campuran dari warna-warna primer.

Ruang *HSV* memiliki 3 karakteristik pokok, yaitu *Hue, Saturation* dan *Value* dengan definisi sebagai berikut :

- Hue : Menyatakan warna sebenarnya, seperti merah, violet, dan kuning dan digunakan menentukan kemerahan (*redness*), kehijauan (*greeness*), dan lainnya.
- Saturation : Sering disebut chroma, adalah kemurnian atau kekuatan warna.
- Value : Menyatakan kecerahan dari warna. Nilainya berkisar antara 0-100 %. Apabila nilainya 0 maka warnanya akan menjadi hitam, semakin besar nilai maka semakin cerah dan muncul variasi-variasi baru dari warna tersebut.

Proses untuk mendapatkan nilai dari setiap warna yang ingin ditampilkan melalui proses perhitungan dengan melakukan konversi ruang warna R-G-B (*Red, Green, Blue*) ke ruang warna H-S-V (*Hue, Saturation, Value*). Cara untuk

mendapatkan setiap nilai H-S-V yang terdefinisi dengan rumus yang ditunjukkan pada Rumus 1 sampai Rumus 6, berikut.

Rumus 1 yang akan ditunjukkan adalah rumus untuk mencari nilai maksimum dari tiga parameter ruang warna RGB yang akan dimasukkan kedalam variabel maks.

$$\text{maks} = \text{MAX}(R,G,B) \quad \dots\dots\dots (1)$$

dimana :
MAX adalah fungsi untuk mencari nilai maksimum dari beberapa angka
R adalah nilai warna merah
G adalah nilai warna hijau
B adalah nilai warna biru

Rumus 2 yang akan ditunjukkan adalah rumus untuk mencari nilai minimum dari tiga parameter ruang warna RGB yang akan dimasukkan kedalam variabel min.

$$\text{min} = \text{MIN}(R,G,B) \quad \dots\dots\dots (2)$$

dimana :
MIN adalah fungsi untuk mencari nilai minimum dari beberapa angka
R adalah nilai warna merah
G adalah nilai warna hijau
B adalah nilai warna biru

Rumus 3 yang akan ditunjukkan adalah rumus untuk mencari selisih nilai maksimum dan nilai minimum dari tiga parameter ruang warna RGB yang akan dimasukkan kedalam variabel min.

$$\text{delta} = \text{maks} - \text{min} \quad \dots\dots\dots (3)$$

dimana :
maks adalah nilai hasil dari fungsi MAX(R,G,B)
min adalah nilai hasil dari fungsi MIN(R,G,B)

Rumus 4 yang akan ditunjukkan adalah rumus untuk mendapatkan nilai *value* parameter ruang warna HSV dari hasil konversi tiga parameter ruang warna RGB yang akan dimasukkan kedalam variabel V.

$$V = \frac{\text{maks}}{255} \times 100 \quad \dots\dots\dots (4)$$

dimana :
maks adalah nilai hasil dari fungsi MAX(R,G,B)

Rumus 5 yang akan ditunjukkan adalah rumus untuk mendapatkan nilai *saturation* parameter ruang warna HSV dari hasil konversi tiga parameter ruang warna RGB yang akan dimasukkan kedalam variabel S.

$$\begin{aligned} S &= 0 && , \text{ Jika } V=0 \\ S &= \frac{\text{maks}-\text{min}}{\text{maks}} \times 100 && , \text{ Jika } V>0 \end{aligned} \quad \dots\dots\dots (5)$$

dimana :
maks adalah nilai hasil dari fungsi MAX(R,G,B)
min adalah nilai hasil dari fungsi MIN(R,G,B)
V adalah nilai warna *Value*

Rumus 6 yang akan ditunjukkan adalah rumus untuk mendapatkan nilai *hue* parameter ruang warna HSV dari hasil konversi tiga parameter ruang warna RGB yang akan dimasukkan kedalam variabel H.

$$\begin{aligned} H &= 0 && , \text{ Jika } S=0 \\ H &= 60 \times \frac{G-B}{\text{delta}} && , \text{ Jika maks } =R \\ H &= 60 \times \left(2 + \frac{B-R}{\text{delta}} \right) && , \text{ Jika maks } =G \\ H &= 60 \times \left(4 + \frac{R-G}{\text{delta}} \right) && , \text{ Jika maks } =B \\ \text{Jika } H < 0, &&& \text{ maka } H=H+360 \end{aligned} \quad \dots\dots\dots (6)$$

dimana :
maks adalah nilai hasil dari fungsi MAX(R,G,B)
delta adalah nilai selisih antara maks dan min
S adalah nilai warna *Saturation*
R adalah nilai warna merah
G adalah nilai warna hijau
B adalah nilai warna biru

Berdasarkan pada Rumus 1 sampai Rumus 6, merupakan perhitungan cara untuk mendapatkan nilai dari setiap warna H-S-V dan cara ini yang paling efektif dipakai jika mendapatkan *Saturation* bernilai 0 sehingga nilai *Hue* akan terdefinisi. Berikut contoh proses konversi ruang warna R-G-B ke H-S-V.

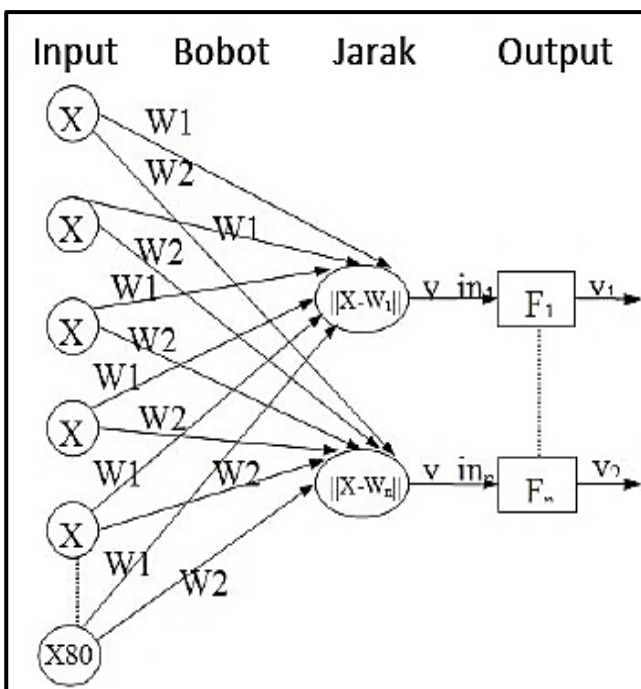
D. Histogram

Histogram merupakan sebuah diagram yang menunjukkan jumlah titik yang terdapat pada sebuah citra untuk setiap tingkat keabuan. Sumbu x (absis) pada histogram menunjukkan tingkat warna, sedangkan sumbu y (ordinat) menunjukkan frekuensi kemunculan titik. Salah satu cara yang dapat digunakan untuk memodifikasi histogram citra adalah perataan histogram (*histogram equalization*) [9]. *Histogram equalization* adalah sebuah proses yang mengubah distribusi nilai derajat keabuan pada sebuah citra sehingga menjadi seragam (*uniform*). Tujuannya untuk

memperoleh penyebaran histogram yang merata sehingga setiap derajat keabuan memiliki jumlah piksel yang relatif sama.

E. Learning Vector Quantization

Learning Vector Quantization (LVQ) merupakan suatu metode untuk melakukan pelatihan terhadap lapisan – lapisan kompetitif yang terawasi. Lapisan – lapisan kompetitif akan belajar secara otomatis untuk melakukan klasifikasi terhadap vector input yang diberikan [10]. Apabila vector input memiliki jarak yang sangat berdekatan, maka vektor – vektor input tersebut akan dikelompokkan dalam kelas yang sama. Contoh arsitektur LVQ terdapat notasi $X_1 \dots X_n$ yang menunjukkan sebagai nilai vektor input, sedangkan notasi $X - W_1 \dots X - W_k$ menunjukkan sebagai perhitungan nilai jarak dari nilai vektor input dan vektor bobot seperti terlihat pada Gambar 1.



Gambar 1. Arsitektur LVQ [11]

Pada Gambar 1 akan dicari perhitungan jarak antara kedua titik. Penghitungan jarak antara 2 titik dapat diselesaikan dengan Euclidean distance menggunakan Rumus 7.

$$C_j = \sqrt{\sum_{i=1}^n (x_i - w_j)^2} \dots\dots\dots (7)$$

dimana :

- C_j adalah nilai jarak Euclidean
- x_i adalah nilai vektor input
- w_j adalah nilai vektor bobot

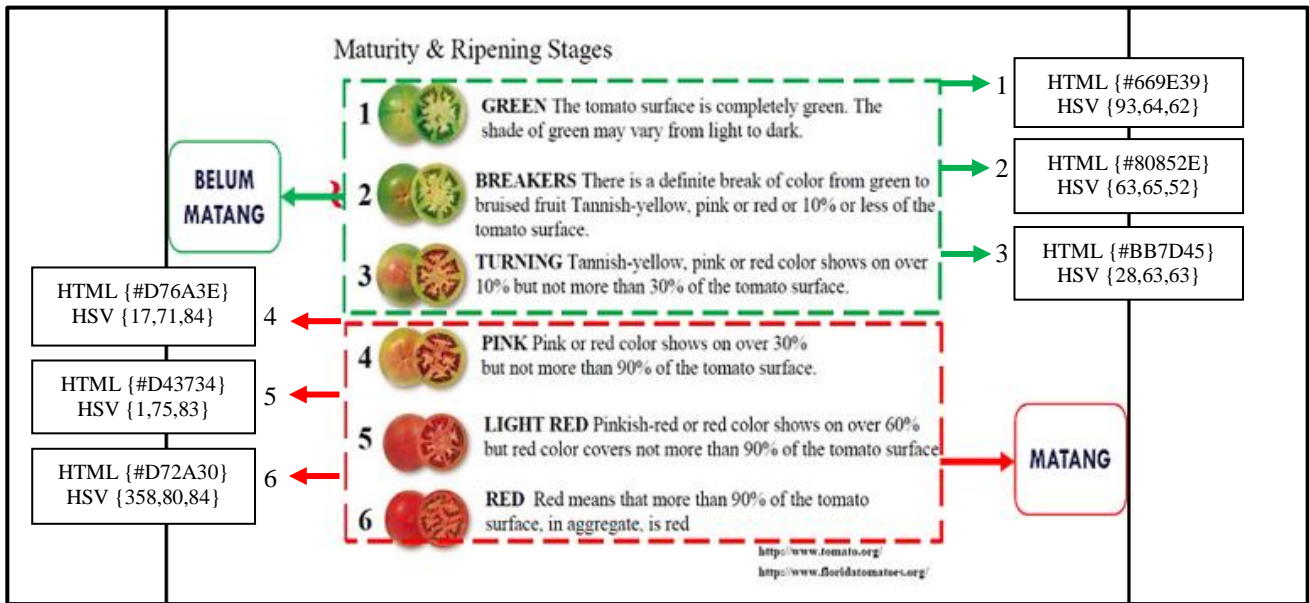
Berdasarkan definisi tersebut, terbentuklah langkah-langkah dalam penyelesaian algoritma LVQ. Langkah – langkah tersebut sebagai berikut :

- 1) Inisialisasi : bobot awal variabel input ke-j menuju kelas ke-i (W), maksimum epoch (MaxEpoch), error minimum yang diharapkan (Eps), Learning rate (α).
- 2) Masukkan :
 - a) Data Input : $x(m,n)$; dengan $i=1,2, \dots, n$ dan $j=1,2, \dots, m$
 - b) Target berupa kelas : $T(1,n)$; dengan $k=1,2, \dots, n$
- 3) Inisialisasi kondisi awal:
 - c) Epoch = 0
 - d) Err = 1
- 4) Kerjakan jika: (epoch < MaxEpoch), nilai error minimum tercapai atau nilai error=0 dan ($\alpha > Eps$)
 - e) Epoch = Epoch+1;
 - f) Kerjakan untuk $i = 1$ sampai n
 - Tentukan Jarak sedemikian hingga $\| x_{ij}-w_{ij} \|$ minimum (sebut sebagai C_j)
 - Perbaiki W_j dengan ketentuan:
 - i. Jika $T = C_j$ maka: $W_{ij}(\text{baru}) = w_{ij}(\text{lama}) + \alpha (x_{ij}-w_{ij}(\text{lama}))$ (3.29)
 - ii. Jika $T \neq C_j$ maka: $W_{ij}(\text{baru}) = w_{ij}(\text{lama}) - \alpha (x_{ij}-w_{ij}(\text{lama}))$ (3.30)

Alasan dari dipilihnya metode Learning Vector Quantization (LVQ) pada penelitian ini untuk proses pengenalan tingkat kematangan tomat berdasarkan citra warna adalah karena metode Learning Vector Quantization (LVQ) merupakan metode yang mampu melakukan klasifikasi untuk membedakan 2 atau lebih kelas kategori dengan melakukan pelatihan secara terbimbing. Maksud dari pelatihan terbimbing adalah proses pelatihannya dilakukan dengan cara memberikan pelabelan diawal untuk menentukan kelas kategori suatu objek. Objek yang akan diklasifikasikan tersebut berupa tomat. Tomat yang telah dilakukan proses pengolahan citra yang menghasilkan warna dari tomat akan menunjukkan tingkat kematangan. Warna yang dihasilkan oleh objek tomat terdiri dari 2 warna yaitu hijau dan merah. Warna yang telah dihasilkan tersebut akan dilakukan pelabelan atau asumsi bahwa tomat tersebut masuk kedalam kategori matang atau belum matang. Pelabelan atau asumsi tersebut diproses menjadi suatu nilai yang menjadi data pelatihan terhadap objek tomat dan tersimpan tersimpan pada database. Proses pelatihan yang telah dilakukan, kemudian diteruskan dengan proses pengujian dimana tomat yang diuji akan disesuaikan dengan data pelatihan untuk menentukan kategori dari tomat sehingga dapat mengenali tingkat kematangan.

F. Tomat

Tomat memiliki ukuran yang bermacam – macam serta memiliki tingkat kematangan yang dilihat dari warna kulit [12]. Tingkat Kematangan tomat dapat dilihat dari warna kulit dengan tingkatan level dan range warna seperti ditunjukkan pada Gambar 2, berikut.



Gambar 2. Objek Buah Tomat [13]

III. METODE PENELITIAN

A. Bahan dan Peralatan

Pada perancangan sistem ini, dibutuhkan beberapa alat, bahan, serta program aplikasi pendukung, yang dikelompokkan menjadi dua bagian, yaitu perangkat keras (*hardware*) dan perangkat lunak (*software*).

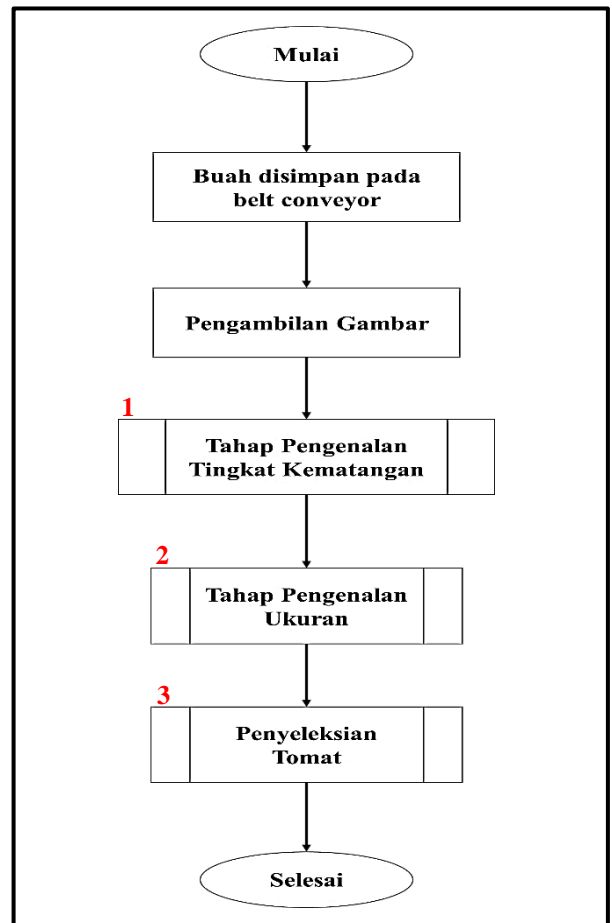
Perangkat keras (*hardware*) yang terdiri dari *Raspberry Pi*, Laptop, *Belt conveyor*, Kamera 12MP serta 5MP dan perangkat lunak (*software*) yang terdiri dari Windows 10 Pro, Netbeans, *OpenCV*, *Xampp* serta *MySQL*.

B. Desain Sistem

Secara garis besar, sistem ini memiliki tiga subsistem diantaranya subsistem pengenalan kematangan tomat, pengenalan ukuran tomat, serta penyeleksian tomat ke dalam beberapa kategori yang ditentukan.

Pada dasarnya subsistem yang dibangun terdiri dari dua bagian komponen yaitu perangkat lunak dan perangkat keras. Perangkat keras dibangun menggunakan kebutuhan perangkat seperti Laptop, Kamera dan *Belt Conveyor* sesuai dengan yang telah dipaparkan pada kebutuhan bahan dan peralatan. Perangkat lunak dibuat dengan menggunakan bahasa pemrograman *Java* untuk dapat menjalankan perangkat keras yang telah dirancang sesuai dengan skenario sistem yakni mengenali tingkat kematangan tomat berdasarkan citra warna.

Fokus pengerjaan sub-sistem yang dibangun pada penelitian ini, yakni mengenai proses pengenalan tingkat kematangan tomat berdasarkan citra warna yang ditandai dengan nomor 1. Berikut ini, akan direpresentasikan mengenai alur cara kerja sistem secara keseluruhan pada Gambar 3.



Gambar 3. Flowchart sistem keseluruhan

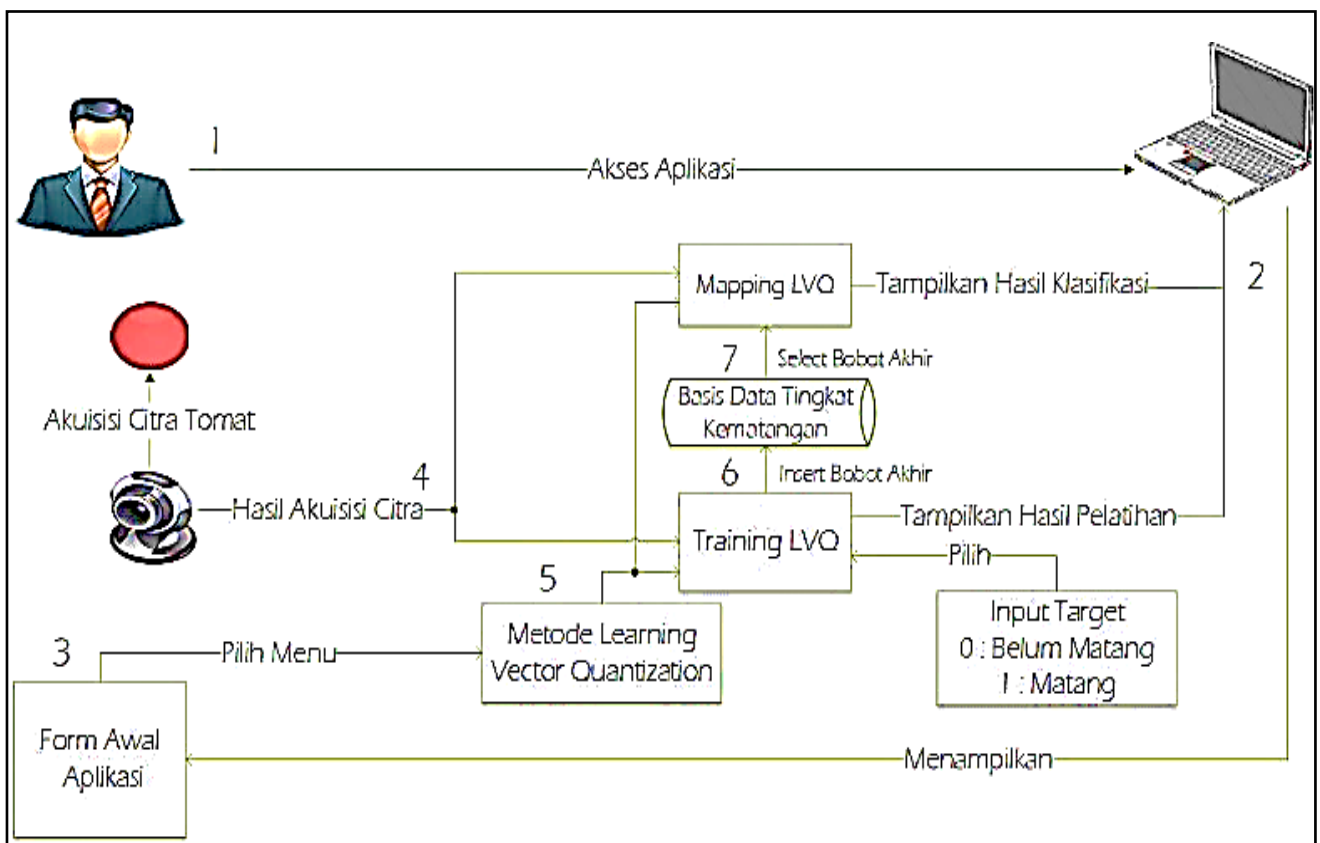
Berdasarkan Gambar 3, subsistem ini dimulai ketika kamera mengambil citra dari buah tomat. Citra tomat tersebut diproses melalui tahapan *image pre-processing* yaitu konversi ruang warna H-S-V (Hue-Saturation-Value) dan ekualisasi histogram untuk menentukan titik – titik dari warna hasil konversi H-S-V sehingga titik – titik tersebut akan ditampung dalam satu array sebanyak citra tomat atau vektor masukkan. Vektor – vektor masukkan yang telah tersimpan dalam satu array tersebut akan dilakukan pembelajaran kedalam 2 kelas sehingga dapat di klasifikasikan menjadi 2 kategori yaitu matang dan belum matang.

C. Perancangan Hardware

Melakukan konfigurasi untuk perancangan perangkat keras antara Laptop, Kamera dan *Belt Conveyor*. Perancangan yang pertama merupakan perancangan antara Laptop dan Kamera yang ditunjukkan dengan blok diagram pada Gambar 4 yaitu perancangan sistem untuk mengenali tingkat kematangan tomat berdasarkan citra warna.

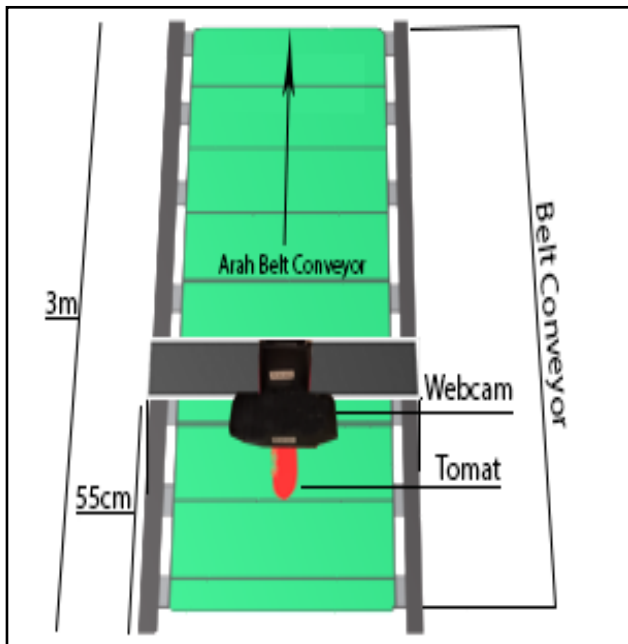
Berdasarkan Gambar 4, ditunjukkan blok diagram sistem untuk mengenali tingkat kematangan tomat berdasarkan citra warna menggunakan metode *Learning Vector Quantization (LVQ)* dengan penjelasan sebagai berikut :

- 1) Pengguna mengakses aplikasi
- 2) Sistem akan menampilkan form utama pada aplikasi untuk melakukan pengenalan tingkat kematangan tomat berdasarkan citra warna.
- 3) Aplikasi akan menunggu pilihan untuk melakukan pemilihan terhadap metode yang digunakan yaitu proses pelatihan atau proses klasifikasi.
- 4) Kamera telah mengakuisisi citra tomat untuk selanjutnya dilakukan proses pada metode *Learning Vector Quantization* dan langsung dapat melakukan proses klasifikasi jika memilih menu *mapping*. Namun jika yang dipilih adalah menu pelatihan, sistem akan melakukan pelatihan.
- 5) Sistem akan melakukan pelatihan terhadap citra tomat untuk mendapatkan nilai bobot akhir dari citra yang akan dilakukan pelatihan dengan memasukkan target pelatihan.
- 6) Hasil pelatihan mendapatkan bobot akhir yang disimpan di *database*
- 7) Melakukan pengambilan nilai bobot akhir yang tersimpan di *database* untuk dapat melakukan proses klasifikasi tingkat kematangan tomat kedalam kategori kelas matang dan belum matang.



Gambar 4. Blok Diagram Sistem

Perancangan yang kedua menunjukkan perancangan alat antara *Belt Conveyor* dengan Kamera yang ditunjukkan pada Gambar 5 untuk membantu melakukan pemrosesan sistem.



Gambar 5. Perancangan Alat

Pada Gambar 5, Kamera *webcam* diletakkan sejajar vertikal diatas terhadap *Belt Conveyor*. Hal ini dilakukan agar jangkauan Kamera *Webcam* dapat menangkap seluruh area *Belt Conveyor* untuk diletakkan nya tomat.

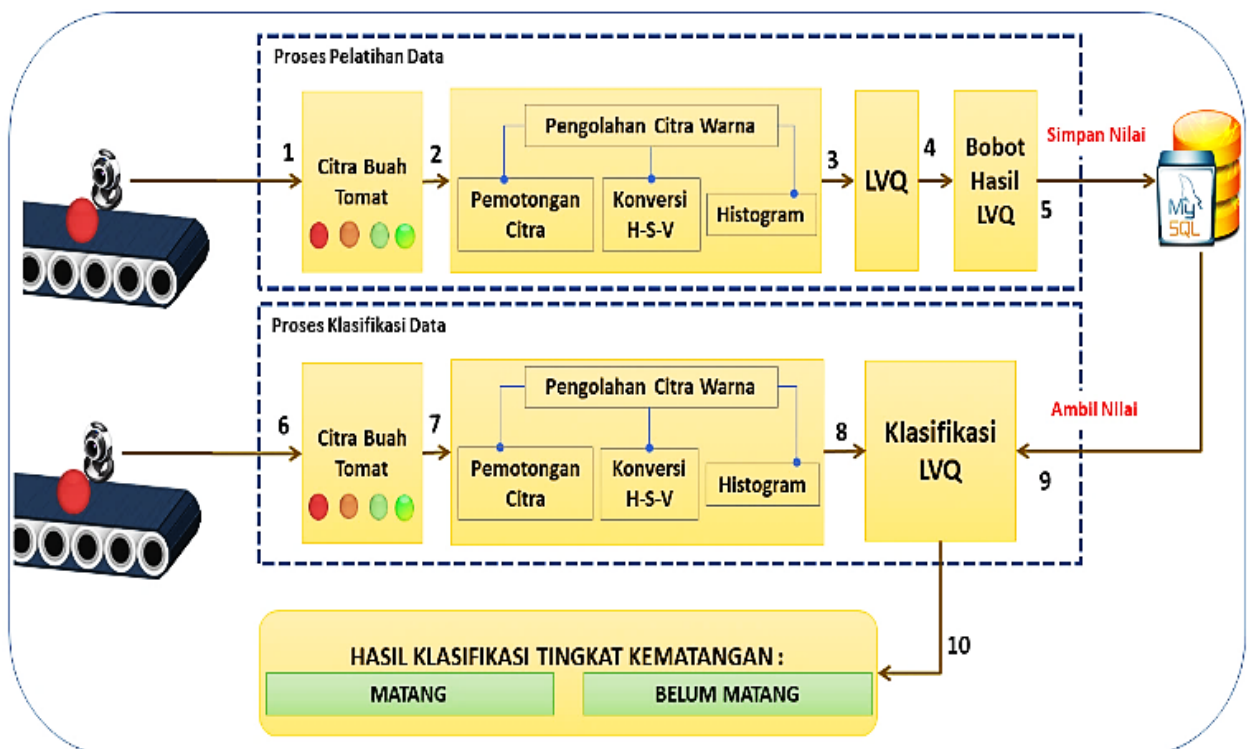
D. Perancangan Software

Melakukan perancangan program secara garis besar program dibuat menggunakan bahasa pemrograman *Java* dan *Library OpenCV* yang terdiri dari *training* dengan menggunakan metode *LVQ* untuk melakukan proses *insert* bobot akhir ke *database MySQL* dan *mapping* dengan menggunakan metode *LVQ* untuk melakukan proses *select* bobot akhir dari *database MySQL*.

Proses *training* (pelatihan) merupakan proses yang dilakukan untuk mendapatkan hasil kuantisasi citra atau pengolahan citra yang telah terbimbing berupa suatu nilai menggunakan metode *Learning Vector Quantization (LVQ)* dan disimpan pada *database*.

Proses *mapping* (pengujian) merupakan proses yang dilakukan untuk mendapatkan hasil klasifikasi tingkat kematangan menggunakan metode *Learning Vector Quantization (LVQ)* dengan mengambil nilai bobot yang tersimpan pada *database*.

Skema dari sistem untuk mengenali tingkat kematangan tomat berdasarkan citra warna untuk melakukan proses *training* (pelatihan) dan *mapping* (pengujian) akan ditunjukkan pada Gambar 6,



Gambar 6. Workflow Skema Sistem

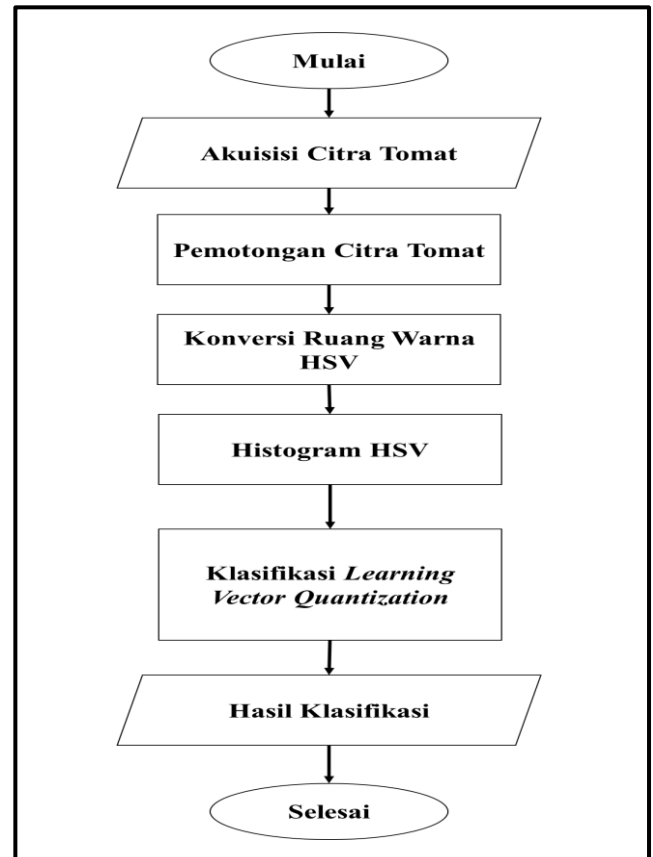
Berdasarkan Gambar 6, digambarkan mengenai skema dari sistem pengenalan tingkat kematangan buah tomat berdasarkan citra warna yang dibangun dengan penjelasan sebagai berikut :

- 1) Melakukan perancangan terhadap sistem yang dibuat.
- 2) Menuliskan *source code* yang dapat memproses sistem pengenalan tingkat kematangan buah tomat berdasarkan citra warna kedalam kategori matang atau belum matang.
- 3) Melakukan pengambilan citra warna menggunakan kamera untuk diproses pada tahap pelatihan selanjutnya.
- 4) Citra warna diambil dan dilakukan proses pelatihan
- 5) Pemrosesan citra warna tersebut akan melewati tahapan *pre-processing* dengan melakukan konversi ruang warna H-S-V dan pembuatan histogram yang sebelumnya melewati tahapan *image cropping* terlebih dahulu.
- 6) Nilai yang didapat dari pembuatan histogram akan melewati perhitungan jarak antara kedua titik dengan menggunakan rumus *euclidean distance*. Jarak tersebut merupakan parameter yang diimplementasikan terhadap metode *Learning Vector Quantization (LVQ)*.
- 7) Proses implementasi metode *Learning Vector Quantization (LVQ)* tersebut akan menghasilkan bobot nilai yang memisahkan *clustering* terhadap 2 kategori yaitu matang dan belum matang.
- 8) Bobot nilai yang telah didapatkan akan disimpan kedalam *database* pelatihan.
- 9) Melakukan pengambilan citra warna menggunakan kamera untuk di proses pada tahap pengujian selanjutnya.
- 10) Citra warna diambil dan dilakukan proses pengujian
- 11) Pemrosesan citra warna tersebut akan melewati tahapan *pre-processing* dengan melakukan konversi ruang warna H-S-V dan pembuatan histogram yang sebelumnya melewati tahapan *image cropping* terlebih dahulu. Nilai yang didapat dari pembuatan histogram akan melewati perhitungan jarak antara kedua titik dengan menggunakan rumus *euclidean distance*. Jarak tersebut merupakan parameter yang akan digunakan saat klasifikasi dengan mengimplementasikan metode *Learning Vector Quantization (LVQ)*.
- 12) Proses pengambilan nilai data latih pada *database*.
- 13) Proses terakhir adalah melakukan penyesuaian nilai warna dari proses – proses yang telah dilakukan sebelumnya antara nilai hasil pelatihan dengan nilai hasil pengujian. Penyesuaian tersebut akan menghasilkan buah tomat yang diuji akan masuk kedalam kategori matang atau belum matang.

E. Algoritma Sistem

Dari tiga subsistem yang berjalan pada sistem pemilih otomatis tomat, salah satu yang menjadi fokus pada

penelitian yang dilakukan yaitu mengenai subsistem pengenalan tingkat kematangan tomat berdasarkan citra warna. Pada subsistem ini terdapat dua kegiatan utama diantaranya tahap *pre-processing* dan tahap implementasi sistem dengan menggunakan metode *Learning Vector Quantization (LVQ)*. Pada Gambar 7, akan ditunjukkan algoritma sistem yang direpresentasikan dengan *flowchart*.



Gambar 7. Flowchart

Berdasarkan Gambar 7, digambarkan mengenai cara penelitian yang dilakukan terhadap sistem pengenalan tingkat kematangan tomat berdasarkan citra warna yang menghasilkan keluaran klasifikasi berdasarkan 2 kelas yang telah ditentukan yaitu matang atau belum matang. Selanjutnya, akan jelaskan secara lengkap tahapan alur dari sistem yang telah dibuat.

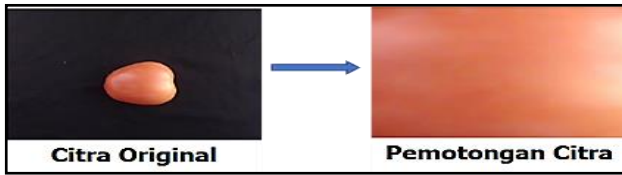
o Akuisisi Citra Tomat



Gambar 8. Akuisisi Citra Tomat

Berdasarkan Gambar 8, ditunjukkan proses untuk melakukan pengambilan citra dari objek tomat.

o Pemotongan Citra Tomat



Gambar 9. Pemotongan Citra Tomat

Berdasarkan Gambar 9, ditunjukkan proses melakukan pemotongan citra untuk mendapatkan area citra tertentu yang akan diamati.

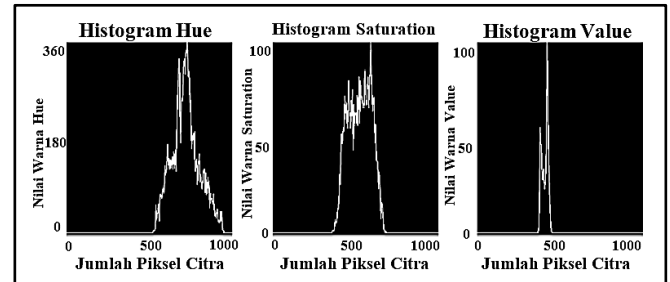
o Konversi Ruang Warna H-S-V



Gambar 10. Konversi Ruang Warna H-S-V

Berdasarkan Gambar 10, ditunjukkan proses melakukan konversi HSV untuk mendapatkan warna yang sesuai dengan warna yang ditangkap oleh penginderaan manusia.

o Histogram Ekualisasi

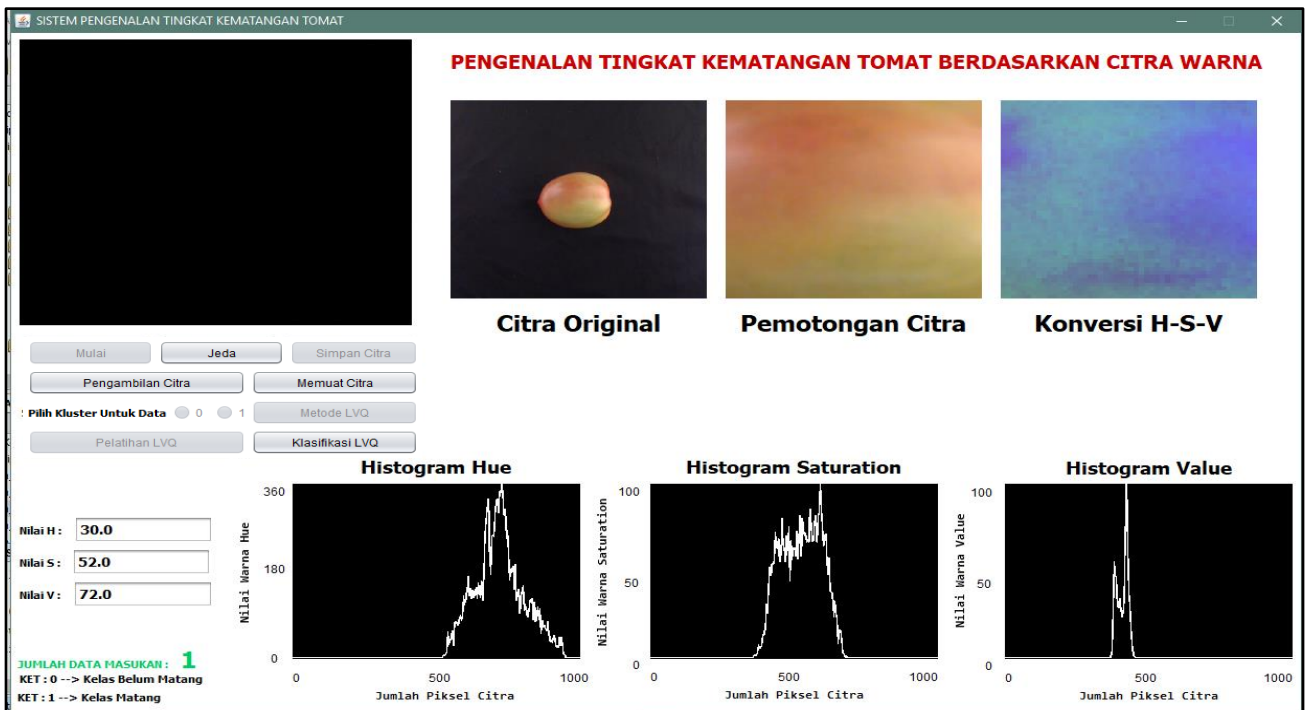


Gambar 11. Histogram Ekualisasi

Berdasarkan Gambar 11, ditunjukkan proses melakukan pembuatan histogram HSV untuk menempatkan titik warna warna hasil konversi RGB ke HSV.

IV. HASIL DAN PEMBAHASAN

Berdasarkan perancangan yang telah dibuat, didapatkan sebuah subsistem yang terdiri dari perangkat keras dan perangkat lunak untuk melakukan pengenalan tingkat kematangan tomat berdasarkan citra warna pada sistem pemilihan otomatis. Perangkat keras yang dibangun berupa rangkaian yang terdiri dari beberapa komponen yang sesuai dengan yang digambarkan pada Gambar 4 dan Gambar 5 mengenai perancangan perangkat keras subsistem. Selain perangkat keras, dibangun pula perangkat lunak dari subsistem untuk mengenali tingkat kematangan tomat berdasarkan citra warna yang dapat dilihat pada Gambar 12 dan Gambar 13.



Gambar 12. Tampilan Form Pre-Processing

Data Tomat Ke	Nilai H	Nilai S	Nilai V	C1	C2	Winner	Klasifikasi
1	14.0	55.0	86.0	3551,951	1575,111	1	Matang
2	47.0	32.0	67.0	2797,371	3626,825	0	Belum Matang
3	45.0	38.0	59.0	2175,823	3754,766	0	Belum Matang
4	47.0	48.0	49.0	2776,830	3407,150	0	Belum Matang
5	4.0	59.0	66.0	3867,721	2673,949	1	Matang
6	15.0	54.0	71.0	3643,128	2227,324	1	Matang
7	4.0	58.0	66.0	3750,713	2796,754	1	Matang
8	48.0	41.0	59.0	2618,439	4068,461	0	Belum Matang
9	13.0	54.0	78.0	3939,630	2253,449	1	Matang
10	6.0	61.0	73.0	3730,440	2545,449	1	Matang
11	30.0	52.0	72.0	3637,278	3261,942	1	Matang
12	16.0	55.0	81.0	3759,853	3527,227	1	Matang
13	18.0	50.0	78.0	3671,751	3037,153	1	Matang
14	56.0	27.0	58.0	2359,685	3705,407	0	Belum Matang
15	5.0	57.0	77.0	3639,257	1810,157	1	Matang
16	42.0	41.0	63.0	2635,621	3811,267	0	Belum Matang
17	3.0	62.0	72.0	4158,692	2581,055	1	Matang
18	46.0	40.0	58.0	2906,210	4138,775	0	Belum Matang
19	17.0	56.0	77.0	4318,113	2948,080	1	Matang
20	15.0	52.0	81.0	3628,530	2730,806	1	Matang
21	14.0	55.0	64.0	3681,656	2455,117	1	Matang
22	23.0	48.0	74.0	3619,373	3065,860	1	Matang
23	8.0	58.0	67.0	3450,022	2176,931	1	Matang

Gambar 13. Tampilan Form Pre-Processing

Berdasarkan Gambar 12, ditunjukkan hasil implementasi dari aplikasi sistem pengenalan tingkat kematangan tomat berdasarkan citra warna. Adapun fungsi dari tahapan pengolahan citra warna yang terdapat pada *form pre-processing* tersebut dengan alur sebagai berikut:

- 1) User yang menjalankan aplikasi tersebut akan menekan tombol *start*, dimana sistem pada aplikasi akan menampilkan tampilan hasil jangkauan dari kamera. Setiap pergerakan yang melewati kamera akan ditampilkan pada *form* dalam *panel webcam*.
- 2) User dapat menjalankan metode *Learning Vector Quantization* dengan memilih akan melakukan pelatihan (*training*) untuk mendapatkan bobot hasil perhitungan citra tomat masukkan dengan parameter pendukung yang telah diinisialisasikan serta melakukan pengujian (*mapping*) untuk dapat melakukan klasifikasi citra tomat kedalam kelas matang atau belum matang.
- 3) Pada tahap pelatihan user harus memasukkan target untuk inialisasi citra tomat masukkan kedalam kelas belum matang (0) atau matang (1) setelah itu bobot akhir hasil pelatihan disimpan kedalam *database*.
- 4) Pada tahap pengujian klasifikasi, sistem akan menampilkan hasil aplikasi citra tomat pengujian.
- 5) User dapat menekan tombol *capture* atau *load image* untuk memasukkan citra tomat yang akan dilakukan atau citra tomat yang akan dilakukan pengujian.

Berdasarkan Gambar 13, ditunjukkan *form* klasifikasi yang dibuat untuk dapat melakukan pengelompokkan data tomat yang masuk kedalam kelas matang atau belum matang. Pengelompokkan data dipengaruhi oleh hasil pelatihan karena akan melakukan pembimbingan terhadap data dengan metode *LVQ*. Proses klasifikasi adalah proses menghitung antara bobot akhir pada *database* dengan citra tomat masukkan yang akan diuji untuk mencari jarak terdekat atau terkecil dengan menggunakan *euclidean distance*.

A. Pengujian Akurasi Sub-Sistem



Setelah dilakukannya proses implementasi, maka perlu dilakukan pula pengujian yang diterapkan pada subsistem pengenalan tingkat kematangan tomat ditunjukkan pada Tabel II, Tabel III dan Tabel IV.

TABEL II
DATA LATIH TOMAT

Data Latih Citra Tomat				
No.	Citra Tomat Latih	Webcam	Target	Kondisi Nyata
1.		12MP	0	Belum Matang
2.		12MP	0	Belum Matang
3.		12MP	0	Belum Matang
4.		12MP	0	Belum Matang
5.		12MP	0	Belum Matang
6.		12MP	1	Matang
7.		12MP	1	Matang
8.		12MP	1	Matang
9.		12MP	1	Matang
10.		5MP	1	Matang
11.		5MP	0	Belum Matang
12.		5MP	0	Belum Matang
13.		5MP	0	Belum Matang
14.		5MP	0	Belum Matang
15.		5MP	0	Belum Matang
16.		5MP	1	Matang
17.		5MP	1	Matang
18.		5MP	1	Matang
19.		5MP	1	Matang
20.		5MP	1	Matang

Pada Tabel II, menunjukkan pengujian pada tahap *training* (pelatihan) untuk mendapatkan nilai bobot akhir. Selanjutnya, akan ditunjukkan pengujian pada tahap *mapping* (klasifikasi) pada Tabel III.



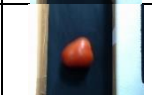










TABEL III
DATA UJI TOMAT STATIS

Data Uji Citra Tomat									
No.	Citra Uji Tomat		C1 (0)	C2 (1)	Webcam	Winner (Kelas Pemenang)	Klasifikasi	Kondisi Nyata	Hasil
1.			3552,0	1575,1	12MP	1	Matang	Matang	Cocok
2.			2797,4	3626,8	12MP	0	Belum Matang	Belum Matang	Cocok
3.			2175,8	3754,8	12MP	0	Belum Matang	Belum Matang	Cocok
4.			2776,8	3407,2	12MP	0	Belum Matang	Belum Matang	Cocok
5.			3867,7	2673,9	12MP	1	Matang	Matang	Cocok
6.			3643,1	2227,3	12MP	1	Matang	Matang	Cocok
7.			3750,7	2796,8	12MP	1	Matang	Matang	Cocok
8.			2618,4	4068,5	12MP	0	Belum Matang	Belum Matang	Cocok
9.			3939,6	2253,4	12MP	1	Matang	Matang	Cocok
10.			3730,4	2545,4	12MP	1	Matang	Matang	Cocok
11.			3637,3	3261,9	12MP	1	Matang	Belum Matang	Tidak Cocok
12.			3759,9	3527,2	12MP	1	Matang	Belum Matang	Tidak Cocok
...									
80.			3333,0	2231,1	5MP	1	Matang	Matang	Cocok




Pada Tabel III, menunjukkan pengujian pada tahap *mapping* (klasifikasi) untuk mendapatkan informasi tingkat kematangan tomat kedalam kategori matang dan belum

matang secara statis (tidak bergerak). Selanjutnya, akan ditunjukkan pengujian pada tahap *mapping* (klasifikasi) pada Tabel IV.

TABEL IIIV
DATA UJI TOMAT BELT CONVEYOR

Data Latih Citra Tomat Belt Conveyor								
No	Citra Tomat Latih	Webcam	C1 (0)	C2 (1)	Selang Waktu(s)	Klasifikasi	Kondisi Nyata	Hasil
1		12MP	4672,2	2435,8	8,97	Matang	Matang	Cocok
2		12MP	2781,0	2730,1	7,34	Matang	Matang	Cocok
3		12MP	4327,3	1753,8	6,71	Matang	Matang	Cocok
4		12MP	3187,8	2039,0	7,45	Matang	Matang	Cocok
5		12MP	1736,4	3544,7	8,51	Belum Matang	Belum Matang	Cocok
6		12MP	2316,7	4211,5	7,56	Belum Matang	Belum Matang	Cocok
7		12MP	2454,2	3430,3	8,69	Belum Matang	Belum Matang	Cocok
8		12MP	2144,7	3417,3	6,32	Belum Matang	Belum Matang	Cocok
9		12MP	3568,5	3446,3	5,55	Matang	Matang	Cocok
10		12MP	3327,6	2995,4	4,96	Matang	Matang	Cocok
11		12MP	4136,4	2891,3	5,51	Matang	Matang	Cocok
12		12MP	3399,9	3260,7	4,55	Matang	Matang	Cocok
13		12MP	2446,1	2946,1	5,73	Belum Matang	Belum Matang	Cocok

TABEL IVV (LANJUTAN)
DATA UJI TOMAT BELT CONVEYOR

Data Latih Citra Tomat Belt Conveyor								
No	Citra Tomat Latih	Webcam	C1 (0)	C2 (1)	Selang Waktu(s)	Klasifikasi	Kondisi Nyata	Hasil
14		12MP	2966,6	3252,6	3,56	Belum Matang	Belum Matang	Cocok
15		12MP	3006,8	3585,5	4,89	Belum Matang	Belum Matang	Cocok
16		12MP	2952,1	3827,1	5,32	Belum Matang	Belum Matang	Cocok
17		12MP	3168,9	3491,1	2,15	Belum Matang	Matang	Tidak Cocok
18		12MP	3659,9	2214,1	1,95	Matang	Matang	Cocok
19		12MP	4264,1	2606,4	2,45	Matang	Matang	Cocok
20		12MP	3412,9	2852,8	2,9	Matang	Matang	Cocok
...								
48		5MP	806,3	1159,5	2,37	Belum Matang	Matang	Tidak Cocok

Pada Tabel IV, menunjukkan pengujian pada tahap *mapping* (klasifikasi) untuk mendapatkan informasi tingkat kematangan tomat berdasarkan citra warna kedalam kategori matang atau belum matang yang dilakukan pada citra tomat yang diletakkan pada *Belt Conveyor* (bergerak).

Hasil pengukuran akurasi dilakukan untuk mengetahui kinerja dari sistem pengenalan tingkat kematangan tomat berdasarkan citra warna yang telah dibuat. Pengukuran tersebut dilihat dari beberapa aspek parameter diantaranya yaitu :

- Perbandingan akurasi antara citra tomat yang diambil dalam posisi tidak bergerak dan bergerak.
- Perbandingan akurasi sistem dengan menggunakan kamera 12MP dan 5 MP.

- Perbandingan akurasi sistem dalam menentukan keberhasilan pengenalan tomat dalam kelas matang dan kelas belum matang.
- Perbandingan akurasi sistem berdasarkan rata – rata waktu sistem untuk memproses dengan mengatur kecepatan conveyor.
- Perbandingan kecepatan conveyor yang paling optimal untuk melakukan pemrosesan citra tomat.
- Akurasi sistem secara keseluruhan.

Berdasarkan pada Tabel III dan Tabel IV yang telah ditunjukkan, maka dapat dilakukan analisis perbandingan akurasi sistem pengenalan tingkat kematangan tomat berdasarkan citra warna. Untuk menghitung tingkat akurasi

pengujian tersebut dengan menggunakan Rumus 8. Adapun analisis perbandingan akurasi ditunjukkan pada Tabel V.

$$\text{Tingkat Akurasi} = \frac{\text{Data Berhasil}}{\text{Jumlah Data}} \times 100\% \quad \dots\dots\dots (8)$$

dimana :

Jumlah Data adalah banyaknya data masukan
Data Berhasil adalah kecocokan data masukan

TABEL V
PENGUKURAN AKURASI SISTEM

Perbandingan Akurasi Antara Citra Tomat Yang Diambil Dalam Posisi Tidak Bergerak Dan Bergerak			
Aspek Pemandang	Jumlah Data	Data Berhasil	Tingkat Akurasi
Citra Tomat (Statis)	80	67	83,75%
Citra Tomat (Conveyor)	48	40	83,33%
Perbandingan akurasi sistem dengan menggunakan kamera 12MP dan 5 MP			
Aspek Pemandang	Jumlah Data	Data Berhasil	Tingkat Akurasi
Kamera 12MP (Statis)	40	38	95,00%
Kamera 5MP (Statis)	40	29	72,50%
Kamera 12MP (Conveyor)	24	23	95,83%
Kamera 5MP (Conveyor)	24	17	70,83%
Perbandingan Akurasi Sistem Dalam Menentukan Keberhasilan Pengenalan Tomat Dalam Kelas Matang Dan Kelas Belum Matang			
Aspek Pemandang	Jumlah Data	Data Berhasil	Tingkat Akurasi
Kamera 12MP Berkategori Matang (Statis)	26	26	100,00%
Kamera 12MP Berkategori Belum Matang (Statis)	14	12	85,71%
Kamera 5MP Berkategori Matang (Statis)	26	20	76,92%
Kamera 5MP Berkategori Belum Matang (Statis)	14	9	64,29%
Kamera 12MP Berkategori Matang (Conveyor)	12	11	91,67%
Kamera 12MP Berkategori Belum Matang (Conveyor)	12	12	100,00%
Kamera 5MP Berkategori Matang (Conveyor)	12	4	50,00%
Kamera 5MP Berkategori Belum Matang (Conveyor)	12	12	91,67%

Pada Tabel V menunjukkan hasil pengukuran akurasi dari sistem pengenalan tingkat kematangan tomat dengan beberapa aspek perbandingan. Berdasarkan perbandingan yang telah dilakukan, maka didapatkan tingkat akurasi sistem secara keseluruhan menggunakan metode *Learning Vector Quantization* dengan prosentasi sebagai berikut :

- Pengambilan Citra Secara Statis

$$\text{Tingkat Akurasi Sistem} = \frac{67}{80} \times 100\% = 83,75\%$$

- Pengambilan Citra Pada *Belt Conveyor*

$$\text{Tingkat Akurasi Sistem} = \frac{40}{48} \times 100\% = 83,33\%$$

Adapun pengukuran kecepatan optimal *Belt Conveyor* dilihat dari perbandingan rata – rata selang waktu tempuh untuk memproses citra tomat menggunakan kamera 12MP dan 5MP yang ditunjukkan pada Tabel VI.

TABEL VI
PENGUKURAN KECEPATAN PADA BELT CONVEYOR

Aspek Pemandang	Kategori Kecepatan Belt	Rata-Rata Waktu	Jumlah Data	Data Berhasil	Tingkat Akurasi
Kamera 12MP	Lambat	7,694s	8	8	100,00%
	Sedang	5,009s	8	8	100,00%
	Cepat	2,025s	8	7	87,50%
Kamera 5MP	Lambat	7,874s	8	8	100,00%
	Sedang	4,999s	8	5	62,50%
	Cepat	1,904s	8	4	50,00%

Berdasarkan Tabel VI, perbandingan tersebut didapatkan waktu terbaik untuk melakukan pemrosesan citra berdasarkan jarak lintasan yaitu 55cm. Waktu terbaik pada kamera 12MP adalah 5,009s dan pada kamera 5MP adalah 7,874s. Dari waktu pemrosesan tersebut dicari kecepatan belt conveyor paling optimal dengan menggunakan Rumus 9.

$$V = \frac{S}{t} \quad \dots\dots\dots (9)$$

dimana :

V adalah kecepatan optimal belt conveyor
S adalah jarak yang ditempuh untuk pemrosesan citra
t adalah selang waktu rata – rata untuk pemrosesan citra

Maka kecepatan optimal *belt conveyor* untuk melakukan pemrosesan citra dengan jarak 55 cm berdasarkan :

- Kamera 12MP: Waktu rata-rata yang didapatkan 5,009s

$$V = \frac{(55cm/100)}{5,009s} = 0,109m/s$$

- Kamera 5MP: Waktu rata-rata yang didapatkan 7,874s

$$V = \frac{(55cm/100)}{7,874s} = 0,069m/s$$

Berdasarkan penelitian yang telah dilakukan untuk mampu mengenali tingkat kematangan tomat berdasarkan citra warna menggunakan metode *Learning Vector Quantization (LVQ)* menghasilkan keterangan kelebihan dari metode *Learning Vector Quantization (LVQ)* tersebut. Kelebihannya yaitu metode *LVQ* melakukan proses pelatihan dengan memberikan asumsi awal terhadap kondisi dari tomat yang akan dikenali yaitu matang atau tidak matang. Jika proses asumsi awal terhadap kondisi tomat dilakukan secara teliti maka hasil yang didapatkan dari proses pengujian akan menghasilkan sistem yang mampu mengenali tingkat kematangan dengan tingkat akurasi yang baik seperti yang telah dilakukan pada penelitian tersebut. Namun, dari sekian kali percobaan dalam penelitian yang dilakukan tersebut terdapat pula hasil yang tidak memiliki kecocokan terhadap kondisi nyata dari tomat sehingga menyebabkan sistem gagal mengenali tingkat kematangan tomat secara tepat. Hal tersebut disebabkan karena metode *LVQ* hanya akan melakukan perhitungan untuk menyesuaikan warna yang telah diasumsikan sebelumnya. Pengaruh dari cahaya menyebabkan kamera yang menangkap citra tomat memiliki pergeseran warna. Pergeseran nilai warna tersebut mempengaruhi saat proses pengujian menggunakan metode perhitungan pada *LVQ* sehingga hal tersebut yang menyebabkan kegagalan sistem mengenali tingkat kematangan secara tepat karena warna yang dihasilkan oleh kamera tidak sama dengan kondisi sebenarnya.

V. PENUTUP

A. Kesimpulan

Berdasarkan hasil penelitian serta pengujian sistem yang telah dilakukan, tingkat akurasi atau kesesuaian sistem pengenalan tingkat kematangan tomat berdasarkan citra warna pada studi kasus pembangunan sistem pemilihan otomatis untuk jenis tomat bervariasi *tw* menghasilkan prosentase sebesar 83,75 % untuk pengambilan citra secara tidak bergerak atau diam dengan menggunakan metode *LVQ* mampu mengenali 67 citra tomat dari 80 citra tomat serta 83,33% untuk pengambilan citra secara bergerak pada *belt conveyor* dengan menggunakan metode *LVQ* mampu mengenali 40 citra tomat dari 48 citra tomat.

B. Saran

Berdasarkan hasil pengujian dari penelitian mengenai citra tomat yang didapatkan tersebut belum menghasilkan hasil yang sempurna. Maka untuk lebih meningkatkan sistem menjadi lebih baik salah satu caranya adalah dengan menggunakan kamera dengan resolusi yang besar dan pemberian cahaya yang tidak berubah – ubah sehingga pada saat proses pengujian dengan menggunakan perhitungan metode *LVQ* akan menghasilkan hasil yang baik dan tepat. Selain itu, pemilihan citra tomat serta asumsi awal kondisi kematangan tomat untuk data pelatihan harap perhatikan karena sangat berpengaruh jika menggunakan metode *LVQ* pada saat melakukan pengujian klasifikasi kelas kategori. Untuk melakukan otomatisasi agar mengurangi kesalahan saat pengujian pada *belt conveyor* sebaiknya pengembang yang akan melakukan penelitian tersebut memberikan deteksi objek atau penggunaan sensor untuk menangkap citra tomat.

DAFTAR PUSTAKA

- [1] Iswahyudi, Catur. *Prototype Aplikasi Untuk Mengukur Kematangan Buah Apel Berdasarkan Kemiripan Warna*. 2010.
- [2] Sulistiyo, Susanto Budi. *Pemutuan Buah Jeruk Siam Pontianak (Citrus nobilis var. microcarpa) dengan Teknik Pengolahan Citra*. 2008.
- [3] Seniwati, Erni. *Identifikasi Kematangan Jeruk Berdasarkan Citra Warna RGB Dan Jaringan Syaraf Tiruan Learning Vector Quantization*. 2014.
- [4] Prahara, Hidayat Wahyu. Sela, Enny Itje. *Tingkat Kematangan Buah Pepaya Dengan Jaringan Syaraf LVQ*. 2016.
- [5] Radityo, Dimas Rizki. *Alat Penyortir Dan Pengecekan Kematangan Buah Menggunakan Sensor Warna*. 2012.
- [6] Damayanti, Fitri. *Pengenalan Citra Wajah Menggunakan Two-Dimensional Linear Disriminant Analysis dan Support Vector Machine*. 2010.
- [7] Arhatin, Risti. *Memotong Citra, Koreksi Radiometrik dan Koreksi Geometrik*. 2010.
- [8] Fauzan, Abdi. *Ruang Warna Hue Saturation Value (HSV) serta Proses Konversinya*. 2015.
- [9] Hertiningsih, I. *Pengolahan Citra Digital*. 2009.
- [10] Qur'ani, Difla Yustisia. *Jaringan Syaraf Tiruan LVQ Untuk Aplikasi Pengenalan Tanda Tangan*. Yogyakarta: SNATI. 2010.
- [11] Karimah, Fatimatul. (2012) Sistem Cerdas(Metode ANN). [Online]. Tersedia: http://fatimatul-k--fst08.web.unair.ac.id/artikel_detail-43814-Sistem%20Cerdas-Metode%20ANN.html.
- [12] Smith, A. F. *The Tomato in America. University of Illinois Press*. ISBN 0-252-07009-7.
- [13] (2017) The Tomato. [Online]. Tersedia: <http://www.tomato.org>.