

PENGEMBANGAN APLIKASI PEMILIHAN BUAH TOMAT UNTUK BIBIT UNGGUL BERDASARKAN WARNA DAN UKURAN MENGGUNAKAN HSV DAN *THRESHOLDING*

Ahmad Zaky Maula¹, Cahya Rahmad², Ulla Delfana Rosiani³

Program Studi Teknik Informatika, Jurusan Teknologi Informasi,
Politeknik Negeri Malang

¹zackymaula@gmail.com, ²cahya.rahmad@yahoo.com, ³ullarosi@gmail.com

ABSTRACT

Tomato is the most widely cultivated fruit in the world. The low production of tomato in Indonesia at planting tomato regardless quality. Then be directed to increase tomato quality with planting of the seed is available from the selection of good tomato quality. Science and digital image processing technology make it possible for tomato quality selection with automatically, by using application tomato fruit selection for superior seed by color and size using color space Hue Saturation Value (HSV) method and thresholding processing. From the result of application has 3 (three) output from the color and size of the fruit is good tomato, half as nice tomato and not nice tomato.

Keyword : *Tomato, Color, Size, Hue Saturation Value (HSV), Quantization HSV, Thresholding, Otsu Threshold, Data Similarly.*

1. PENDAHULUAN

Buah tomat saat ini merupakan salah satu komoditas hortikultura yang bernilai ekonomi tinggi dan masih memerlukan penanganan serius, terutama dalam hal peningkatan hasilnya dan kualitas buahnya. Rendahnya produksi tomat di Indonesia kemungkinan disebabkan penanaman tomat tanpa memperhatikan kualitasnya (Wariantio, 2011:1). Maka perlu diarahkan untuk meningkatkan hasil dan kualitas buah tomat dengan menanam bibit unggul yang diperoleh dari pemilihan buah tomat yang berkualitas.

Perkembangan ilmu pengetahuan dan teknologi pengolahan citra digital memungkinkan untuk memilih produk pertanian dan perkebunan tersebut secara otomatis.

Berdasarkan uraian sebelumnya, maka perlu dibangun sebuah sistem pengolahan citra pemilihan kualitas buah tomat berdasarkan warna dan ukuran menggunakan ruang warna *Hue Saturation Value* (HSV) dan pengolahan *Thresholding*. Sehingga dapat memudahkan pemilihan dan meningkatkan kualitas dari buah tomat.

Pendukung pemilihan buah menggunakan bantuan smartphone Android yang memiliki kamera untuk mendapatkan warna dan ukuran dari buah tomat yang diidentifikasi. Dari hasil identifikasi yang

diperoleh menghasilkan 3 output yaitu bagus, setengah bagus, dan tidak bagus.

Rumusan Masalah

Berdasarkan latar belakang, permasalahan yang akan dibahas adalah bagaimanakah membangun aplikasi pengolahan citra digital dalam menentukan kualitas buah tomat yang digunakan untuk bibit unggul berdasarkan warna dan ukuran buah tomat menggunakan ruang warna *Hue Saturation Value* (HSV) dan pengolahan *Thresholding*?

Tujuan Penelitian

Tujuan penelitian ini adalah untuk mengetahui bagaimana proses pengolahan citra dengan menggunakan ruang warna HSV dan pengolahan *Thresholding* yang digunakan untuk mengukur tingkat kualitas warna maupun ukuran buah tomat.

Manfaat Penelitian

Manfaat penelitian ini adalah untuk meningkatkan hasil dan kualitas dari buah tomat.

2. LANDASAN TEORI

Tomat

Tomat (*Lycopersicon esculentum*) adalah tumbuhan dari keluarga Solanaceae, tumbuhan asli Amerika Tengah dan Selatan, dari Meksiko sampai Peru. Kata “tomat”

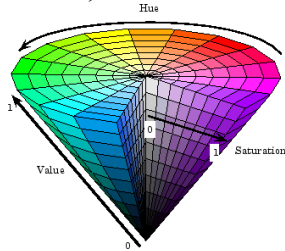
berasal dari kata dalam bahasa Nauhat, dimana tomat merupakan keluarga dekat dari kentang, menurut Sabiro (dalam Kusumaningrum, 2015:5).

Pengolahan Citra

Secara umum, istilah pengolahan citra digital menyatakan pemrosesan gambar berdimensi-dua melalui komputer digital, menurut Jain (dalam Kadir, 2013:2).

Hue Saturation Value (HSV)

H berasal dari kata “hue”, S berasal dari “saturation”, dan V berasal dari “value”



Gambar 2.1 Ruang HSV (Sumber: MATLAB)

Untuk mendapatkan nilai H, S, V berdasarkan R, G, dan B.

$$r = \frac{R}{R+G+B}, g = \frac{G}{R+G+B}, b = \frac{B}{R+G+B} \quad (2.1)$$

$$V = \max(r, g, b) \quad (2.2)$$

$$S = \begin{cases} 0, & \text{jika } V = 0 \\ 1 - \frac{\min(r, g, b)}{V}, & V = 0 \end{cases} \quad (2.3)$$

$$H = \begin{cases} 0, & \text{jika } S = 0 \\ \frac{60 \cdot (g-b)}{S \cdot V}, & \text{jika } V = r \\ 60 \cdot \left(2 + \frac{b-r}{S \cdot V}\right), & \text{jika } V = g \\ 60 \cdot \left(4 + \frac{r-g}{S \cdot V}\right), & \text{jika } V = b \end{cases} \quad (2.4)$$

$$H = H + 360 \text{ jika } H < 0 \quad (2.5)$$

Kuantisasi HSV

Jika menggunakan nilai H, S, dan V secara langsung, maka memerlukan banyak komputasi. Lebih baik dikuantisasi secara *non-equal intervals* ke 8x3x3 histogram 72 dimensi.

$$H = \begin{cases} 0 & \text{if } h \in [316, 20] \\ 1 & \text{if } h \in [21, 40] \\ 2 & \text{if } h \in [41, 75] \\ 3 & \text{if } h \in [76, 155] \\ 4 & \text{if } h \in [156, 190] \\ 5 & \text{if } h \in [191, 270] \\ 6 & \text{if } h \in [271, 295] \\ 7 & \text{if } h \in [296, 315] \end{cases} \quad (2.6)$$

$$S = \begin{cases} 0 & \text{if } s \in [0, 0.2] \\ 1 & \text{if } s \in [0.2, 0.7] \\ 2 & \text{if } s \in [0.7, 1] \end{cases} \quad (2.7)$$

$$V = \begin{cases} 0 & \text{if } v \in [0, 0.2] \\ 1 & \text{if } v \in [0.2, 0.7] \\ 2 & \text{if } v \in [0.7, 1] \end{cases} \quad (2.8)$$

Dari vektor 3 dimensi nilai H, S, dan V, akan dinormalisasi ke 1 dimensi.

$$G = 9H + 3S + V \quad (2.9)$$

Menurut Gopalan (2012:50), HSV bisa mempunyai beberapa variasi kuantisasi, dari 2x2x2=8 sampai 72x10x10=7200. Dari kuantisasi 7200 dimensi, Hue (355, 360, 5, 10, 15, ... 355), Saturation (0, 0,1, 0,2, ... 1.0), dan Value (0, 0,1, 0,2, ... 1.0).

Mengkonversi Jenis Citra

Dalam praktik, citra berwarna seringkali harus dikonversi ke dalam bentuk citra berskala keabuan (*greyscale*).

$$I = 0.2989 \times R + 0.5870 \times G + 0.1141 \times B \quad (2.10)$$

Mengkonversi citra berskala keabuan ke citra biner (0 atau 1), yaitu dengan memakai *threshold*.

$$b(i) = \begin{cases} 0, & i \geq a \\ 1, & i < a \end{cases} \quad (2.11)$$

Threshold Otsu

Metode ini menentukan nilai *threshold* secara otomatis dengan cara membedakan dua kelompok, yaitu objek dan latar belakang. Langkah-langkahnya:

Probabilitas nilai intensitas *i* histogram

$$p(i) = \frac{n_i}{N}, p(i) \geq 0, \sum_{i=1}^{256} p(i) = 1 \quad (2.12)$$

Pembobotan kedua kelas

$$w_1(t) = \sum_{i=1}^t p(i) \quad (2.13)$$

$$w_2(t) = \sum_{i=t+1}^L p(i) = 1 - w_1(t) \quad (2.14)$$

Rerata kedua kelas

$$m_1(t) = \sum_{i=1}^t i \cdot p(i) / w_1(t) \quad (2.15)$$

$$m_2(t) = \sum_{i=t+1}^L i \cdot p(i) / w_2(t) \quad (2.16)$$

Varian kedua kelas

$$\sigma_1^2(t) = \sum_{i=1}^t (1 - m_1)^2 \cdot \frac{p(i)}{w_1(t)} \quad (2.17)$$

$$\sigma_2^2(t) = \sum_{i=t+1}^L (1 - m_2)^2 \cdot \frac{p(i)}{w_2(t)} \quad (2.18)$$

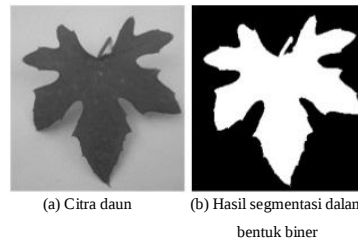
Varian total

$$\sigma^2(t) = \sigma_w^2(t) + \sigma_B^2(t) \quad (2.19)$$

Hitung BCV

$$\sigma_B^2 = w_1 \cdot [m_1(t) - m_T]^2 + w_2 \cdot [m_2(t) - m_T]^2 \quad (2.20)$$

BCV bertujuan mencari nilai *threshold* sebuah citra *greyscale*, sebagai acuan mengubah citra *greyscale* ke citra biner. Setiap citra memiliki nilai *threshold* yang berbeda-beda (Syafi'i, dkk. 2015:3). Proses binerisasi bisa menggunakan persamaan 2.11.



(a) Citra daun (b) Hasil segmentasi dalam bentuk biner

Gambar 2.2 Pemisahan objek dengan *backgorund* (Sumber: Kadir, 2013:336)

Mencari Luas Objek

Cara sederhana menghitung luas objek adalah dengan cara menghitung jumlah piksel pada objek tersebut.

Ukuran Kemiripan Data

Istilah ketidakmiripan terkadang juga dianggap sebagai ukuran jarak antara dua data. Berikut cara mencari kemiripan dan ketidakmiripan dua data dengan satu atribut, dengan tipe atribut interval dan rasio formula ketidakmiripan:

$$d = |x - y| \quad (2.21)$$

Formula kemiripan

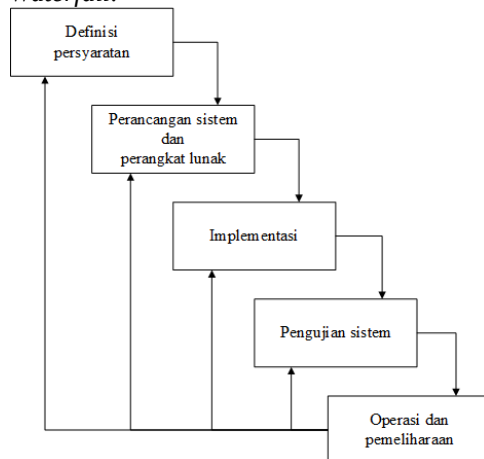
$$s = -d \quad (2.22)$$

Nilai s dan d tidak ada yang negatif.

3. METODOLOGI PENELITIAN

Rancangan Penelitian

Metode pembuatan perangkat lunak ini adalah dengan menggunakan metode *Waterfall*.



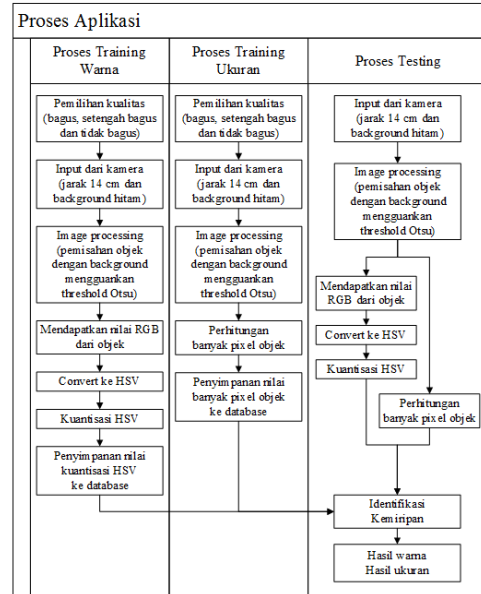
Gambar 3.1 Diagram *Waterfall* (Sumber: Sommerville, 2003:43)

Data

Komponen yang diperlukan untuk melakukan penelitian ini salah satunya adalah data. Data yang diperlukan sebagai objek pengujian aplikasi pemilihan buah tomat untuk bibit unggul yaitu buah tomat yang memiliki komposisi warna dan ukuran yang satu sama lain berbeda, serta image atau gambar yang didapat dari data sampel buah tomat.

Pengolahan Data

Sesuai dengan tujuan penelitian yang telah dikemukakan pada pendahuluan maka penelitian ini menggunakan kerangka penelitian sebagai berikut:



Gambar 3.2 Diagram blok proses aplikasi

Pada Gambar 3.2 merupakan proses dari aplikasi pemilihan buah tomat untuk bibit unggul berdasarkan warna dan ukuran menggunakan HSV (Hue Saturation Value) dan Thresholding. Terdapat 3 (tiga) proses dalam aplikasi ini, yaitu proses training warna, proses training ukuran dan proses testing.

Proses Training Warna

Proses training warna dilakukan untuk memperoleh nilai kuantisasi HSV dari nilai warna buah tomat menurut kualitas yang nantinya akan diuji melalui proses testing.

Proses Training Ukuran

Proses training ukuran dilakukan untuk memperoleh nilai banyak piksel objek dari nilai ukuran buah tomat menurut kualitas yang nantinya akan diuji melalui proses testing.

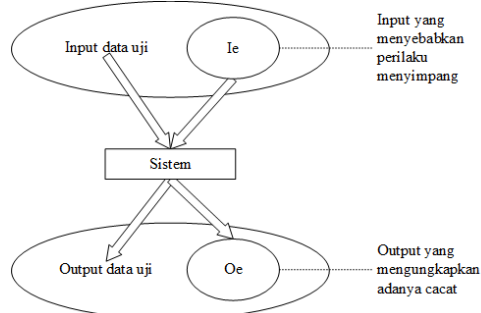
Proses Testing

Proses testing bertujuan untuk menguji aplikasi menggunakan nilai kuantisasi HSV dan nilai banyak piksel objek yang dihasilkan dari proses training warna maupun ukuran dengan melakukan identifikasi kualitas buah tomat dari segi warna maupun ukuran.

Metode Pengujian

Dalam hal pengujian dilakukan dengan pengujian unit, yang ditujukan untuk memastikan bahwa setiap fungsi berjalan

dengan benar dimana strategi pengujiannya menggunakan pengujian fungsional atau pengujian kotak hitam (black-box testing) merupakan pendekatan pengujian yang ujinya diturunkan dari spesifikasi program atau komponen.



Gambar 3.3 Pengujian kotak hitam (Sumber: Sommerville, 2003:87)

4. ANALISIS DAN PERANCANGAN

Analisis Sistem

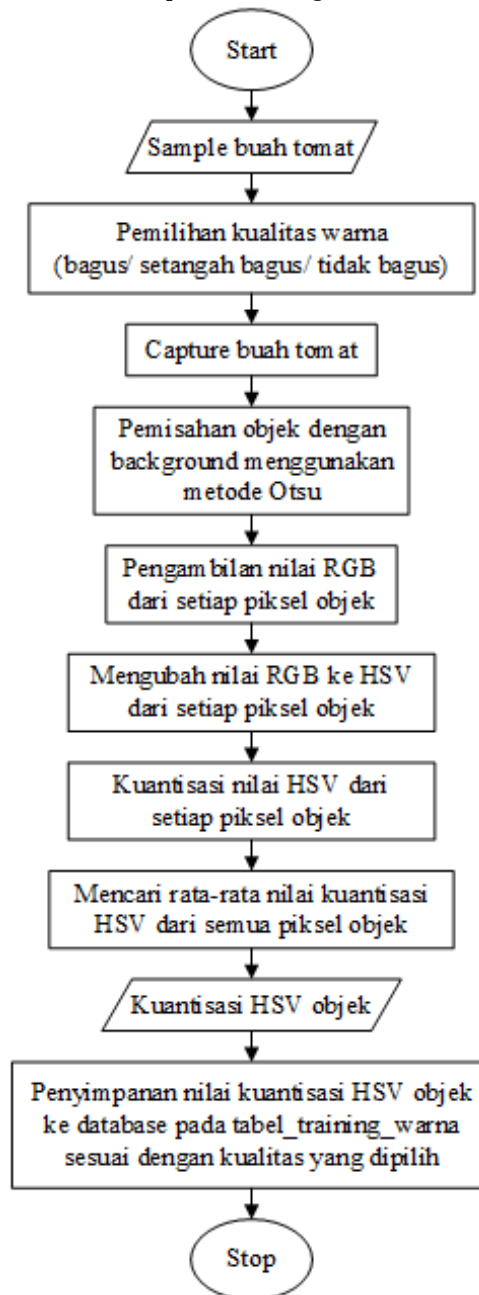
Proses yang terdapat dalam sistem yaitu proses *training* warna maupun ukuran dan proses *testing*.

Perancangan Proses

Pada perancangan proses untuk aplikasi pengolahan citra digital pemilihan buah tomat untuk bibit unggul berdasarkan warna dan ukuran menggunakan ruang warna HSV dan pengolahan *Thresholding* akan disajikan ke dalam bentuk *flowchart*. *Flowchart* merupakan sebuah diagram yang digunakan untuk menjelaskan alur sebuah sistem dengan menggunakan bentuk-bentuk atau simbol untuk menggambarkan aliran data dari proses-proses yang saling berhubungan.

Simbol-simbol dalam *flowchart* akan menjelaskan *input*, *process* dan *output* yang terjadi dalam sistem. Berikut adalah *flowchart* yang dibagi menjadi 3 (tiga) proses yaitu proses *training* warna, proses *training* ukuran dan proses *testing*.

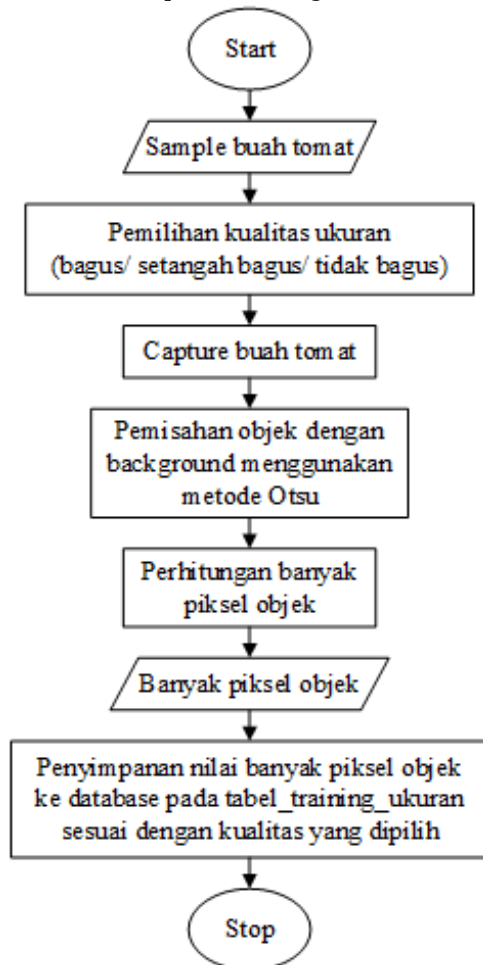
- Flowchart proses *training* warna



Gambar 4.1 Proses *training* warna

Dalam *training* warna pertama-tama yaitu pemilihan kualitas warna buah, kemudian *capture*, setelah itu hasil *capture* diproses dengan pemisahan objek dan *background* menggunakan *threshold Otsu*, kemudian nilai RGB dari objek dirubah ke HSV, setelah itu dihitung kuantisasi HSV, kemudian menghitung rata-rata kuantisasi HSV dari piksel-piksel objek, setelah itu rata-rata nilai kuantisasi HSV disimpan di tabel *training* warna.

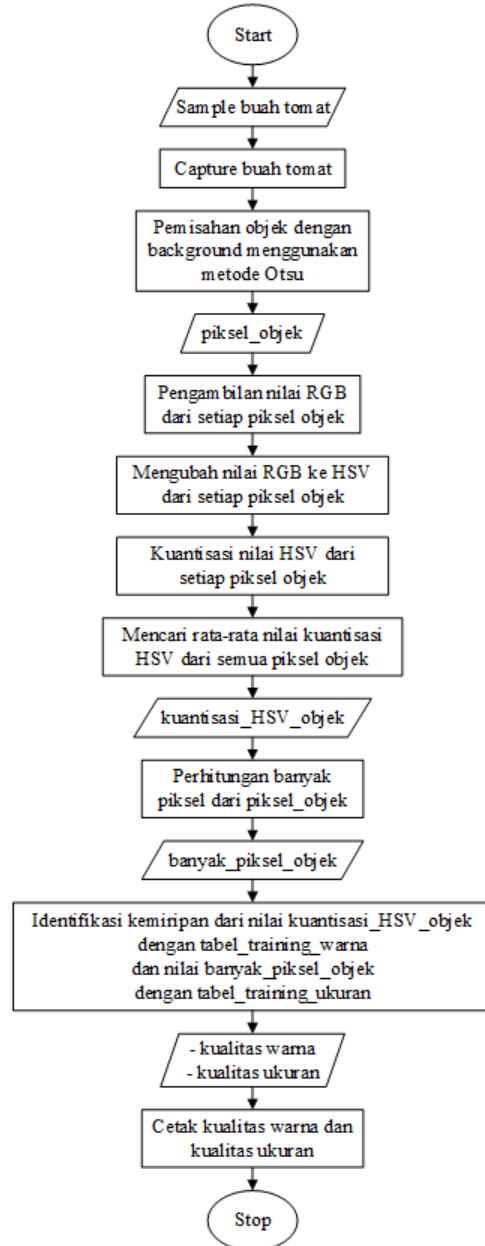
- Flowchart proses training ukuran



Gambar 4.2 Proses training ukuran

Dalam training ukuran pertama-tama yaitu pemilihan kualitas warna buah, kemudian *capture*, setelah itu hasil *capture* diproses dengan pemisahan objek dan *background* menggunakan *threshold Otsu*, kemudian menghitung banyak piksel objek, setelah itu nilai banyak piksel objek disimpan di tabel *training* ukuran.

- Flowchart proses testing



Gambar 4.3 Proses testing

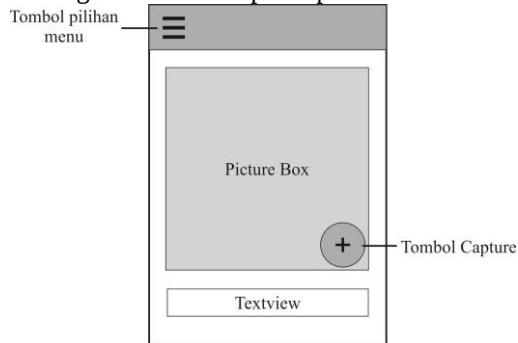
Dalam proses *testing* pertama-tama yaitu *capture* gambar, setelah itu hasil *capture* diproses dengan pemisahan objek dan *background*, kemudian nilai RGB dari objek dirubah ke HSV, setelah itu dihitung kuantisasi HSV, setelah itu menghitung banyak piksel objek. Kemudian nilai kuantisasi HSV dibandingkan dengan nilai yang ada pada tabel *training* warna dan nilai banyak piksel objek dibandingkan dengan nilai yang ada pada tabel *training* ukuran. Setelah nilai kuantisasi HSV dan nilai banyak piksel objek tersebut dibandingkan,

kemudian menampilkan hasil kualitas warna dan ukuran.

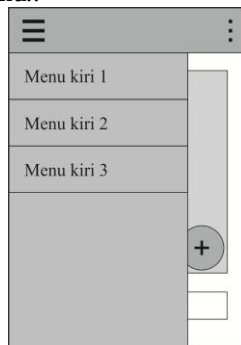
Perancangan Antar Muka

a. Halaman utama

Tampilan halaman utama merupakan halaman awal ketika menjalankan aplikasi. Pada halaman ini terdapat pilihan menu yang berisi menu halaman utama, menu pelatihan warna, menu pelatihan ukuran, dan menu tentang untuk melihat profil penulis.



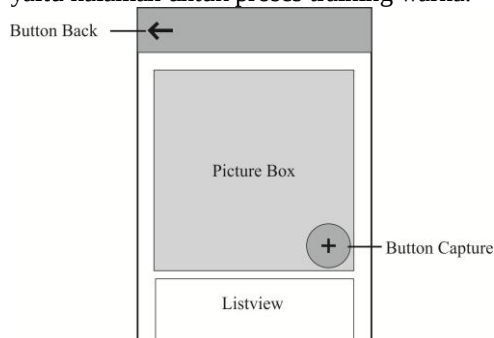
Gambar 4.3 Rancangan halaman utama. Berikut adalah perancangan pilihan menu yang ada pada halaman utama, terdapat 3 pilihan menu.:



Gambar 4.4 Rancangan pilihan menu pada halaman utama

b. Halaman *training* warna

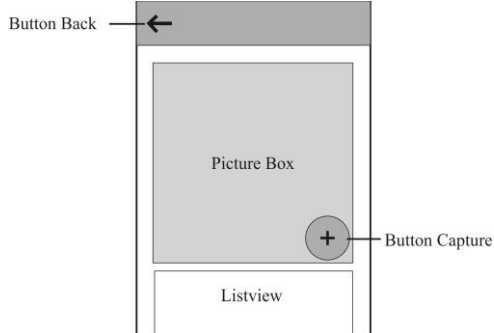
Halaman *training* warna merupakan hasil tampilan setelah memilih menu pelatihan warna pada pilihan menu di halaman utama. Halaman *training* warna yaitu halaman untuk proses *training* warna.



Gambar 4.5 Rancangan halaman *training* warna

c. Halaman *training* ukuran

Halaman *training* ukuran merupakan hasil tampilan setelah memilih menu pelatihan ukuran pada pilihan menu di halaman utama. Halaman *training* ukuran yaitu halaman untuk proses *training* ukuran.

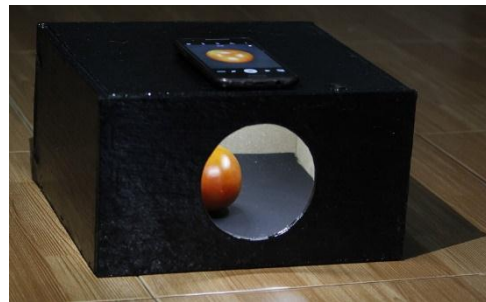


Gambar 4.6 Rancangan halaman *training* ukuran

Perancangan Mekanik

Perancangan mekanik diperlukan untuk pengambilan gambar tomat agar cahaya dan jarak stabil. Dalam perancangan ini diperlukan *box* yang sekiranya tidak membuat cahaya jadi memantul dan cahaya dari luar tidak masuk. Lampu LED agar pencahayaan stabil serta tinggi dari *box* disesuaikan 14 cm agar stabil. Tinggi dari *box* jika tidak stabil maka akan mempengaruhi ukuran dari buah tomat ketika proses *capture*.

Berikut adalah tampilan rancangan dari *box* yang akan dibuat, *box* berbentuk balok dan satu sisi ada lubang untuk memasukkan buah tomat.



Gambar 4.7 Perancangan mekanik box dan lampu LED

Pada bagian atas *box* terdapat lubang persegi yang digunakan kamera untuk tempat pengambilan objek. *Box*, lampu LED, tinggi *box* disusun sedemikian rupa untuk pengambilan objek secara maksimal.



Gambar 4.8 Lubang atas tengah untuk tempat pengambilan objek dari kamera

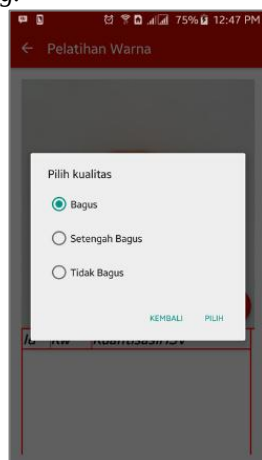
5. IMPLEMENTASI

Implementasi Training

Implementasi *training* ada 2 proses yaitu proses *training* warna dan proses *training* ukuran.

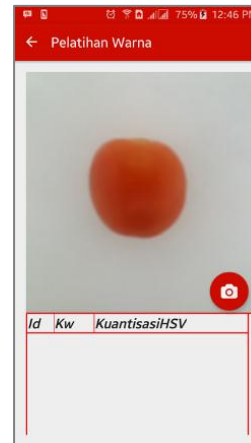
- Proses *training* warna

Pemilihan kualitas warna adalah proses awal dari *training* warna yang digunakan sebagai penentuan kualitas warna dari buah tomat yang akan di-*training*.



Gambar 5.1 Tampilan pemilihan kualitas

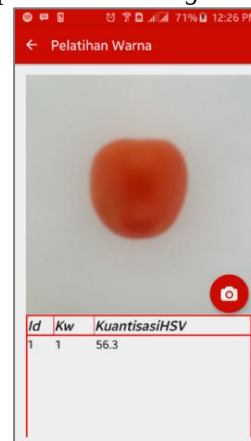
Pengambilan gambar adalah proses selanjutnya, pengambilan gambarnya dengan jarak 14 cm antara kamera dengan *background*, dan warna dari *background* yaitu putih. Dari proses pengambilan tersebut menghasilkan gambar dalam format *bitmap*.



Gambar 5.2 Tampilan pengambilan gambar

Pemisahan objek dengan *background* bertujuan untuk mendapatkan piksel-piksel dari objek saja. Proses pertama yaitu perhitungan metode *Otsu* yang bertujuan untuk memperoleh nilai *threshold* sebagai pemisah objek dan *background*. Potongan kode perhitungan metode *Otsu* ada pada Lampiran 3. Setelah mendapatkan *threshold* dari perhitungan metode *Otsu*, selanjutnya yaitu proses pemisahan antara objek dengan *background* menggunakan nilai *threshold* yang diperoleh dari perhitungan metode *Otsu*.

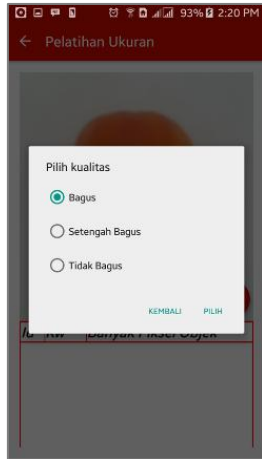
Setelah mendapatkan piksel-piksel dari objek, selanjutnya yaitu mengambil nilai RGB dari setiap piksel objek tersebut, kemudian nilai RGB tersebut dirubah ke nilai HSV (*Hue, Saturation, Value*). Setelah mendapatkan nilai HSV dari setiap piksel objek, selanjutnya yaitu menghitung kuantisasi HSV. Nilai kuantisasi HSV dari setiap piksel tersebut dihitung rata-rata dari semua nilai kuantisasi HSV piksel objek. Nilai rata-rata kuantisasi HSV dari objek tersebut selanjutnya yaitu disimpan ke database pada tabel *training* warna.



Gambar 5.3 Tampilan isi tabel *training* warna

- Proses *training* ukuran

Pemilihan kualitas ukuran adalah proses awal dari *training* ukuran yang digunakan sebagai penentuan kualitas ukuran dari buah tomat yang akan di-*training*.



Gambar 5.4 Tampilan pemilihan kualitas

Pengambilan gambar adalah proses selanjutnya, pengambilan gambarnya dengan jarak 14 cm antara kamera dengan *background*, dan warna dari *background* yaitu putih. Dari proses pengambilan tersebut menghasilkan gambar dalam format *bitmap*.



Gambar 5.5 Tampilan pengambilan gambar

Pemisahan objek dengan *background* bertujuan untuk mendapatkan piksel-piksel dari objek saja. Proses pertama yaitu perhitungan metode *Otsu* yang bertujuan untuk memperoleh nilai *threshold* sebagai pemisah objek dan *background*. Setelah mendapatkan *threshold* dari perhitungan metode *Otsu*, selanjutnya yaitu proses pemisahan antara objek dengan *background* menggunakan nilai *threshold* yang diperoleh dari perhitungan metode *Otsu*.

Setelah mendapatkan piksel-piksel dari objek, selanjutnya yaitu menghitung banyak piksel objek tersebut. Nilai banyak piksel dari objek tersebut selanjutnya yaitu disimpan ke database pada tabel *training* ukuran.

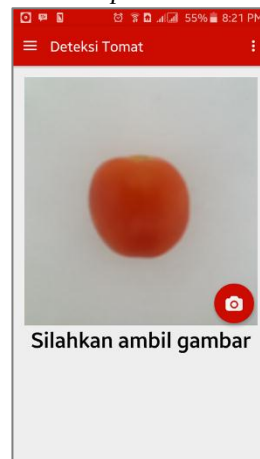


Gambar 5.6 Tampilan isi tabel *training* ukuran

Implementasi *Testing*

Implementasi proses *testing* yaitu menjelaskan tentang alur dari pengecekan kualitas warna maupun ukuran dari buah tomat yang akan di-*testing*.

Pengambilan gambar adalah proses awal dari proses *testing*, pengambilan gambarnya dengan jarak 14 cm antara kamera dengan *background*, dan warna dari *background* yaitu putih. Dari proses pengambilan tersebut menghasilkan gambar dalam format *bitmap*.



Gambar 5.7 Tampilan pengambilan gambar

Pemisahan objek dengan *background* bertujuan untuk mendapatkan piksel-piksel dari objek saja. Proses pertama yaitu perhitungan metode *Otsu* yang bertujuan untuk memperoleh nilai *threshold* sebagai pemisah objek dan *background*. Setelah

mendapatkan *threshold* dari perhitungan metode *Otsu*, selanjutnya yaitu proses pemisahan antara objek dengan *background* menggunakan nilai *threshold* yang diperoleh dari perhitungan metode *Otsu*.

Setelah mendapatkan piksel-piksel dari objek, selanjutnya yaitu mengambil nilai RGB dari setiap piksel objek tersebut, kemudian nilai RGB tersebut dirubah ke nilai HSV (*Hue, Saturation, Value*). Setelah mendapatkan nilai HSV dari setiap piksel objek, selanjutnya yaitu menghitung kuantisasi HSV. Nilai kuantisasi HSV dari setiap piksel tersebut dihitung rata-rata dari semua nilai kuantisasi HSV piksel objek. Nilai piksel-piksel objek yang didapatkan dari proses pemisahan objek tersebut, dihitung banyaknya piksel objek.

Proses identifikasi kemiripan bertujuan untuk memunculkan kualitas warna maupun ukuran dari data buah yang di-*testing*. Dari proses pengolahan warna objek maupun ukuran objek telah mendapatkan nilai kuantisasi HSV objek dan nilai banyak piksel objek, dan nilai-nilai tersebut akan dihitung kemiripannya. Untuk nilai kuantisasi HSV objek akan dihitung kemiripannya dengan database pada tabel *training* warna. Dan untuk nilai banyak piksel objek akan dihitung kemiripannya dengan database pada tabel *training* ukuran.

Berikut adalah tampilan dari hasil identifikasi warna dan ukuran pada aplikasi yang ada pada halaman utama:



Gambar 5.8 Tampilan hasil identifikasi warna dan ukuran

6. PENGUJIAN DAN PEMBAHASAN Pengujian

Dalam pengujian terdapat 4 bagian, yaitu pengujian untuk proses *training* warna, proses *training* ukuran, proses *testing* satu sisi, dan proses *testing* beberapa sisi.

a. Pengujian proses *training* warna



Gambar 6.1 Proses *training* warna
Tabel 6.1 Hasil *training* warna

Kuantisasi HSV	Kw
55,9093	1

b. Pengujian proses *training* ukuran



Gambar 6.2 Proses *training* ukuran
Tabel 6.2 Hasil *training* ukuran

Banyak Piksel Objek	Kw
101611	1

c. Pengujian proses *testing*



Gambar 6.3 Proses *testing*

Berikut adalah tabel hasil dari proses *testing*:

Tabel 6.3 Hasil *testing*

W				U	Kw
H	S	V	kHSV	Bpo	
15,8	0,8	0,7	63,4	87151	W:1 U:2

Keterangan:

- H : Hue
- S : Saturation
- V : Value
- kHSV : Kuantisasi HSV
- Bpo : Banyak Piksel Objek
- Kw : Kualitas
- W : Warna
- U : Ukuran
- Kw 1 : Kualitas Bagus
- Kw 2 : Kualitas Setengah Bagus
- Kw 3 : Kualitas Tidak Bagus

Pembahasan

Dari hasil keseluruhan pengujian, pada proses *training* warna telah dapat menampilkan nilai kuantisasi HSV dan kualitas warnanya dari data sampel buah tomat, untuk proses *training* ukuran telah dapat menampilkan nilai banyak piksel objek dan kualitas ukurannya dari data sampel buah tomat. Pada proses *testing* telah dapat menampilkan kuantisasi HSV, banyak piksel objek, kualitas warna maupun ukuran dari sampel *testing*.

7. KESIMPULAN

Kesimpulan

Berdasarkan hasil dari penelitian yang dilakukan dapat diambil beberapa kesimpulan sebagai berikut:

- a. Aplikasi pengolahan citra digital pemilihan buah tomat untuk bibit unggul berdasarkan warna dan ukuran menggunakan ruang warna Hue Saturation Value (HSV) dan pengolahan *Thresholding* ini, telah mampu menampilkan kualitas warna maupun ukuran dari buah tomat berupa kualitas bagus, setengah bagus dan tidak bagus.
- b. Aplikasi telah mampu untuk menampilkan nilai warna dari buah tomat berupa nilai Hue, Saturation, Value dan kuantisasi HSV, aplikasi telah mampu menampilkan nilai ukuran dari buah tomat berupa nilai banyak piksel objek.

Saran

Saran yang ditujukan untuk pengembangan penelitian ini adalah dalam pengambilan gambar, kamera tidak boleh

autofocus, karena bisa mempengaruhi ukuran buah tomat, tetapi untuk aplikasi ini jika fitur *autofocus* dihilangkan, kualitas gambar akan menurun, dan akan mempengaruhi dalam proses penghitungan dari objek. Maka diharapkan untuk pengembangan aplikasi, menggunakan kamera yang kualitasnya tetap bagus walaupun dihilangkan fitur *autofocus*.

Daftar Pustaka

- Gopalan, R., dan Niranjana, S., 2012. "Performance Efficiency of Quantization using HSV Colour Space and Intersection Distance in CBIR". *International Journal of Computer Applications*. 42(21), 48-53.
- Halim, A., et al., 2015. "Aplikasi Image Retrieval dengan Histogram Warna dan Multi-scale GLCM". 16(1), 41-50.
- Kadir, A dan Susanto, A., 2013. *Teori dan Aplikasi Pengolahan Citra*. Yogyakarta: Andi Offset.
- Kusumaningrum, S., 2015. *Identifikasi Kematangan Buah Tomat Berdasarkan Warna Menggunakan Metode Jaringan Syaraf Tiruan (JST)*. Program Studi Teknik Informatika: Laporan akhir Tidak Diterbitkan.
- Prasetyo, E., 2012. *Data Mining – Konsep dan Aplikasi Menggunakan MATLAB*. Yogyakarta: Andi Offset.
- Safaat, N., 2013. *Aplikasi Berbasis Android–Berbagai Implementasi dan Pengembangan Aplikasi Mobile Berbasis Android*. Bandung: Informatika Bandung.
- Sommerville, I., 2003. *Software Engineering (Rekayasa Perangkat Lunak)/Edisi 6/ Jilid 1*. Jakarta: Erlangga.
- _____. 2003. *Software Engineering (Rekayasa Perangkat Lunak)/Edisi 6/ Jilid 2*. Jakarta: Erlangga.
- Supriyadi, A., 2010. *Pengembangan Benih Tomat (Lycopersicum esculentum Mill) Bersertifikat Di UPTD BP2TPH Ngipiksari, Kaliurang, Yogyakarta*. Program Studi Agribisnis Holtikultura dan Arsitektur Pertamanan: Laporan akhir Tidak Diterbitkan.
- Syafi'i, S. I., et al., 2015. "Segmentasi Obyek Pada Citra Digital Menggunakan Metode Otsu

- Thresholding”. *Jurnal Informatika*, 13(1), 1-8.
- Wibowo, J.S., 2011. “Deteksi dan Klasifikasi Citra Berdasarkan Warna Kulit Menggunakan HSV”. *Jurnal Teknologi Informasi DINAMIK*. 16(2), 118-123.

