

# Pendeteksi Tomat Siap Panen Berdasarkan Warna Dengan Menggunakan Kamera

## Detection Of Tomatoes Ready To Harvest Based On Coloe Using A Camera

1<sup>st</sup> Mahesa Cipta Adila  
Fakultas Ilmu Terapan  
Universitas Telkom  
Bandung, Indonesia

mahescipta@student.telkomuniversity.ac.id.

2<sup>nd</sup> Marlindia Ike Sari  
Fakultas Ilmu Terapan  
Universitas Telkom  
Bandung, Indonesia

Marlindia@staff.telkomuniversity.ac.id.

3<sup>rd</sup> Rini Handayani  
Fakultas Ilmu Terapan  
Universitas Telkom  
Bandung Indonesia

Rinihandayani@staff.telkomuniversity.ac.id

**Abstrak** — secara tradisional memanen buah tomat dapat dilakukan petani dengan cara memetik buah tergantung bentuk ukuran serta warna dari tomat tersebut. Seiring perkembangan teknologi, untuk menentukan kematangan buah tomat tersebut tidak hanya dilakukan secara konvensional (manual) tetapi bisa juga dapat dilakukan secara *computing* (berbasis teknologi). Cara komputasinya bisa dilakukan dengan menggunakan kamera sebagai pengolah citra dari gambar yang di rekam (*image processing*). Buah tomat diidentifikasi berdasarkan input RGB akan di konversikan ke nilai HSV untuk deteksi kualitasnya. Setelah proses tersebut maka buah tomat akan diberi label sesuai tingkat kematangannya, dalam proses ini tidak hanya mendeteksi warna tomat tetapi dapat menghitung jumlah tomat yang terdeteksi sesuai warna. Jarak yang dapat mendeteksi warna tomat mulai dari 10cm – 40 cm, sedangkan > 50 cm warna tomat tidak dapat terdeteksi. Untuk cahaya yang dapat mendeteksi warna tomat di dalam ruangan yaitu sebesar 3 lux (gelap) – 89 lux (terang), sedangkan cahaya yang di luar ruangan sebesar 2 lux (gelap) – 1155 (terang) warna tomat dapat terdeteksi.

**Kata kunci**— buah tomat, image processing, computing, rgb, hsv, cahaya, lux.

### I. PENDAHULUAN.

Secara tradisional memanen buah tomat dapat dilakukan petani dengan cara memetik buah tergantung bentuk ukuran serta warna dari tomat tersebut. Tomat merupakan buah atau sayur yang suka dikonsumsi sehari-hari oleh masyarakat dan juga di manfaatkan sebagai bumbu masak maupun dikonsumsi sebagai buah langsung. Buah tomat memiliki 3 warna untuk menentukan apakah buah tomat tersebut sudah matang, setengah matang, dan mentah, yaitu warna merah ketika buah tomat sudah matang, warna kuning ketika buah tomat setengah matang, dan warna hijau ketika buah tomat mentah. Untuk menentukan kematangan buah tomat tersebut tidak hanya dilakukan secara konvensional (manual), tetapi bisa juga dapat dilakukan secara *computing* (berbasis

teknologi) [1]. Salah satunya dengan menggunakan citra digital yang akan mendapatkan nilai warna pada tomat, nilai tersebut didapatkan melalui proses deteksi RGB (Red, Green dan Blue) [2]. Dimulai pada mendeteksi warna yang diproses sehingga akan menjadi suatu keputusan matang atau tidaknya tomat yang dilihat. Sehingga tomat yang siap panen akan bu In ke pasar tradisional ataupun supermarket dengan tomat pilihan yang baik [2]. covid-19 dari pengunjung yang tidak mematuhi protokol. Dengan dibuatnya aplikasi ini diharapkan dapat.

### II. KAJIAN TEORI

Dari hasil pengamatan pakar diperoleh nilai subjektif yang menjadi acuan untuk perbandingan akurasi pada sistem dan performansi sistem. Pakar mengamati dengan dua sisi citra tomat. Jumlah citra yang diamati sebanyak 50 buah tomat dengan koresponden berjumlah 15 orang. Setelah melakukan pengamatan, 17 tomat dinyatakan mentah, 17 tomat dinyatakan setengah matang dan 16 tomat matang. Sebelumnya tomat telah dilakukan pengambilan citra dan dicari nilai *mean*, standar deviasi dan variannya. Nilai tersebut akan menjadi ciri dan *database* untuk melakukan pengujian[2].

Penelitian oleh Sri Chandrasekharendra Saraswathi Viswa Mahavidyalaya, Kanchipuram 2021 membuat sistem yang dapat mengidentifikasi warna. Dengan manfaatkan *library OpenCV* sistem akan dapat Ekstraksi warna *RGB* pada fase ini warna 3 lapis diekstraksi dari gambar input. Setiap warna primer mengambil nilai intensif 0 dan tertinggi 255 [3].

Penelitian oleh Iskandar Zulkarnain, Mukhlis Ramadhan, Badrul Anwar 2019 membuat alat pendeteksi Pendeteksi Warna Benda Menggunakan *Fuzzy Logic* dengan Sensor TCS3200 Berbasis *Arduino*. Dengan sensor TCS3200 sistem dapat mendeteksi 7 warna yaitu merah, hijau, biru, kuning, coklat, orange dan pink. Sistem deteksi warna menggunakan sensor TCS3200 bahwa tingkat akurasi sangat tergantung dari

beberapa hal seperti pencahayaan, jenis benda warna yang akan dideteksi, jarak antara sensor dengan objek warna[4].

Penelitian oleh Yanri Bili Eliezer Purba , Naikson F Saragih , Arina Prima Silalahi, Suriyanto Sitepu , Asaziduhu Gea 2022 [5] Membuat alat pendeteksi kematangan buah nanas. Dengan metode *Convolutional Neural Network (CNN)* data input tan akan menentukan citra pada *grayscale* atau hitam putih dengan juga *RGB* kemudian melakukan proses *pooling*. [5]

#### A. Tomat.

Tomat memiliki kandungan yang baik bagi tubuh manusia sehingga banyak sekali orang-orang yang mengkonsumsi tomat untuk berbagai hal terutama tomat yang matang dan siap dipakai. Buah tomat segar mengandung beberapa zat, yang penting bagi tubuh yaitu karbohidrat, terutama *glucose* dan *fructose* serta asam organik terutama asam *citric* dan *malic*[6]

#### B. Python

Untuk Bahasa program yang digunakan yaitu Python merupakan Bahasa pemrograman yang sederhana, Bahasa ini dapat mendukung berbagai gaya pemrograman termasuk struktural dan berorientasi objek Bahasa ini mudah karena bisa dijalankan tanpa harus di *compile*[7]. Pengolahan citra digital merupakan sebuah ilmu yang mempelajari hal - hal yang berkaitan dengan perbaikan kualitas citra. Dengan kata lain pengolahan citra adalah kegiatan memperbaiki kualitas citra agar mudah diterjemahkan oleh manusia atau mesin (komputer)[8].

#### C. OpenCv

*OpenCV (Open Computer Vision)* yaitu sebuah API (*Application Programming Interface*) *library* yang sudah sangat familiar pada pengolahan citra *computer vision*. *Computer vision* merupakan salah satu cabang dari bidang pengolahan citra (*image processing*) yang memungkinkan computer dapat melihat seperti manusia. *OpenCV* ini bisa menggunakan *library C/C++* dan juga bisa menggunakan *python*[9].

#### D. Kamera

*Webcam* digunakan untuk mengambil gambar dari objek yang dideteksi. Hasil *capture* dari *webcam* ini dapat ditampilkan pada PC atau dapat melalui internet. *Webcam* atau *WebCamera* adalah sebuah kamera video digital kecil yang dihubungkan ke komputer melalui port USB ataupun port COM [10].

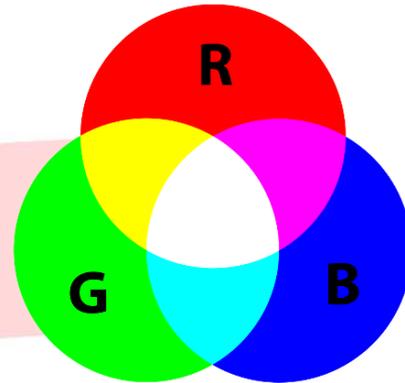
#### E. Pycharm

*PyCharm* adalah lingkungan pengembangan terintegrasi (IDE) yang digunakan dalam pemrograman komputer, khusus untuk bahasa *Python*. Aplikasi ini dikembangkan oleh sebuah perusahaan yang berasal dari Ceko bernama *JetBrains*. Aplikasi ini menyediakan beberapa kemudahan untuk menganalisis kode, *debugger grafis*, unit tester terintegrasi, dan integrasi dengan sistem kontrolversi(*VCSes*)[11].

#### F. Histogram

Histogram adalah Komposisi warna yang merupakan salah satu fitur yang dapat digunakan dalam sistem temu balik citra. Komposisi warna juga dapat direpresentasikan kedalam histogram warna yang nantinya akan merepresentasikan distribusi jumlah piksel untuk setiap intensitas warna dalam citra[12].

#### G. Deteksi Ruang RGB.



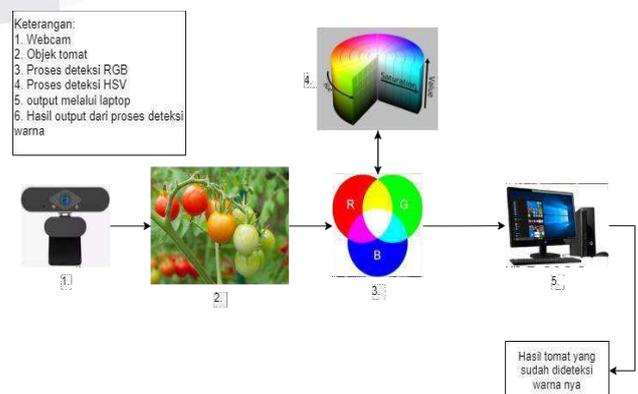
GAMBAR 1  
RUANG WARNA RGB

Ruang Warna RGB adalah sebuah ruang warna additif dimana pancaran warna *red* (merah), *green* (hijau), dan *blue* (biru) ditambahkan bersama dengan cara yang bervariasi untuk mereproduksi susunan warna yang lebar (Gonzalez, 1987). Warna additif digunakan untuk *lighting*, *dvideo*, dan *monitor*. Tujuan utama dari ruang warna *RGB* adalah untuk mempresentasikan ulang, dan menampilkan gambar dalam sistem elektronik, misalnya dalam televisi dan komputer[9].

### III. HASIL DAN PEMBAHASAN

#### A. Gambaran Sistem

Untuk perancangan sistem saat ini yaitu mendeteksi warna dan objek dari tomat. Sistem deteksi ini dimulai dari webcam yang berfungsi sebagai penangkap objek tomat, setelah objek tomat terdeteksi maka akan ada proses seleksi warna yaitu warna merah, orange, kuning dan hijau. Dari proses tersebut kemudian akan diketahui jika tomat itu matang atau tidak matang.



GAMBAR 2  
SISTEM SAAT INI

B. Identifikasi Kebutuhan Sistem

TABEL 1  
KEBUTUHAN FUNGSIONAL

No	Kebutuhan Fungsional
1.	Membuat sistem untuk mendeteksi warna buah tomat menggunakan <i>webcam</i> dengan metode RGB yang dikonversikan ke HSV
2.	Mendeteksi keakuratan cahaya dengan menggunakan lux meter
3.	Dapat menghitung tomat berdasarkan warna yang dideteksi

TABEL 2  
KEBUTUHAN NON FUNGSIONAL

No	Kebutuhan Non Fungsional
1.	<i>Webcam</i> sebagai inputan gambar untuk mendeteksi warna tomat.
2.	Python sebagai Bahasa pemrograman.
3.	OpenCV sebagai <i>library</i> Bahasa pemrograman pyhton.
4.	Comma Separated Values atau CSV merupakan format teks untuk menyimpan informasi.
5.	Laptop sebagai mesin pemrosesan.
6.	Matplotlib sebagai <i>library</i> untuk membuat histogram.
7.	Histogram merupakan diagram untuk menentukan warna dan kecerahan.

C. Metode

Metode pengembangan yang digunakan pada sistem ini adalah Metode yang digunakan dalam penelitian ini terdiri atas lima tahapan utama yakni tahap pengambilan dataset citra tomat (akuisisi citra), tahap preprocessing meliputi operasi mengubah citra RGB ke channel R, G, dan B, tahap segmentasi teknik Otsu, tahap morfologi meliputi operasi konversi channel green ke gray, kemudian dilakukan operasi closing, hole filling, area open, area objek,

1. Akurasi citra

Akuisisi citra adalah proses mengambil citra untuk dijadikan dataset dalam penelitian. Proses mengambil citra yakni bentuk citra analog diubah ke citra digital dengan memanfaatkan kamera.

2. Tahapan Preprocessing

Tahap preprocessing diawali dengan membaca citra tomat yang diinput ke sistem, kemudian citra RGB dikonversi ke channel R, G, dan B. lalu menentukan channel citra yang cocok untuk dijadikan sebagai citra untuk tahap segmentasi.

3. Segmentasi

Segmentasi adalah proses membagi citra menjadi beberapa bagian untuk mendapatkan objek yang terkandung pada citra dengan setiap objek atau daerah memiliki kemiripan atribut.

4. Operasi Morfologi.

Operasi Morfologi adalah operasi yang umum dikenakan pada citra biner (hitam-putih) untuk mengubah struktur bentuk objek yang terkandung dalam citra. Operasi morfologi yakni penggabungan beberapa piksel area background menjadi area objek atau sebaliknya.

5. Ekstraksi Fitur Warna RGB

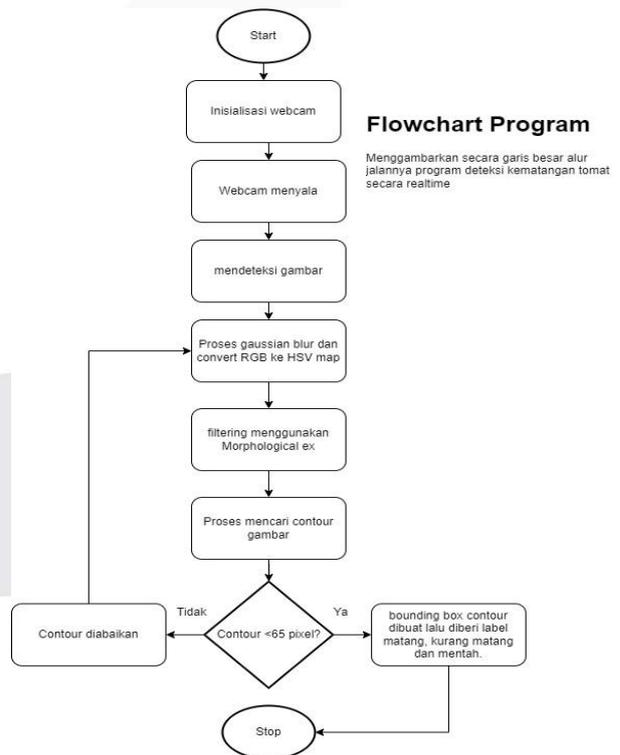
Proses ekstraksi fitur yakni fitur warna RGB, berdasarkan citra hasil operasi morfologi didapat area objek pada citra tomat. Selanjutnya, pada area objek dilanjutkan pada proses ekstraksi fitur red, green, dan blue.

6. Tahapan Klasifikasi.

Tahap klasifikasi merupakan tahap terakhir setelah proses ekstraksi fitur.

D. Flowchart

Berikut ini merupakan gambaran flowchart sebagai berikut:



GAMBAR 3  
FLOWCHART

Berdasarkan gambar maka dijelaskan cara kerja dari sistem deteksi warna pada tomat menggunakan webcam sebagai mengambil gambar untuk ditransformasikan gambar dengan proses gaussian blur dan convert ke HSV setelah dari proses maka akan diproses lagi melalui untuk menghilangkan noise pada gambar dan mencari contours pada gambar, jika

contour lebih besar dari 65-pixel maka gambar akan diabaikan, jika kurang dari 65-pixel maka akan menampilkan label warna dan perhitungan kelas.

H. Implementasi Dan Pengujian

1. Implementasi

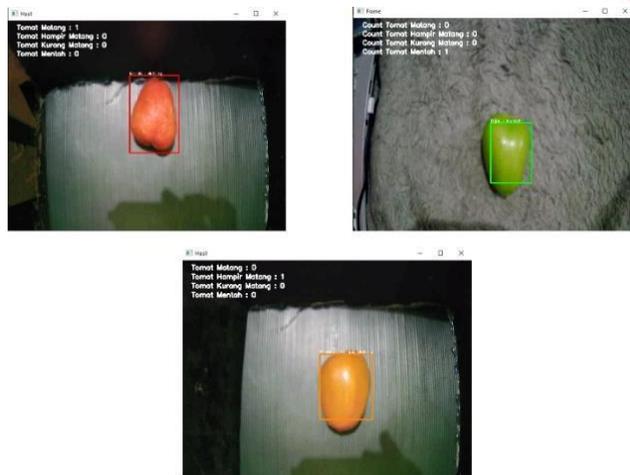
Implementasi yang dilakukan pada sistem ini adalah menggunakan perangkat lunak PyCharm sebagai code editor yang berperan sebagai pendukung sistem ini dan perangkat kerasnya menggunakan webcam sebagai penangkap objek. Untuk library nya itu memakai Cv2, numpy dan argparse dan Bahasa pemrograman nya menggunakan python.

2. Pengujian

Pengujian yang dilakukan terhadap sistem deteksi kematangan ini meliputi berbagai hal, salah satunya adalah pengujian terhadap kematangan pada tomat.

a. Pengujian Jarak

Pengujian dilakukan dengan sample 10 tomat matang, 10 tomat hampir matang, 10 tomat mentah. Objek akan ditaruh didalam kotak lalu dilakukan pengujian dengan cara menaikkan tinggi kamera terhadap objek dengan jarak 20 cm, 40 cm, 60 cm.



GAMBAR 4 HASIL PENGUJIAN

TABEL 3 KEMATANGAN TOMAT, UKURAN, JARAK

Jarak (cm)	Tomat matang ( $\leq 6$ cm)	Tomat setengah matang ( $\leq 7$ )	Tomat mentah ( $\leq 6$ )
10	Terdeteksi	Terdeteksi	Terdeteksi
15	Terdeteksi	Terdeteksi	Terdeteksi
20	Terdeteksi	Terdeteksi	Terdeteksi
25	Terdeteksi	Terdeteksi	Terdeteksi
30	Terdeteksi	Terdeteksi	Terdeteksi
35	Terdeteksi	Terdeteksi	Terdeteksi
40	Terdeteksi	Terdeteksi	Tidak Terdeteksi
45	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
50	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
55	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
60	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi

Pada table 4.1 dapat disimpulkan jika jarak tomat mulai dari 10 cm sampai 35 cm warna tomat dapat terdeteksi, sedangkan jarak 45 cm sampai 60 cm warna tomat tidak dapat terdeteksi.

b. Pengujian jumlah objek tomat.

Pengujian ini dilakukan untuk mengetahui jumlah tomat yang dapat terdeteksi secara real-time baik tomat yang matang maupun yang mentah. Dengan kamera yang ditaruh diatas objek tomat dengan tinggi 30cm-35cm. Setelah dilakukan pengujian dengan menggunakan banyak nyj, alat ini dapat mendeteksi jumlah tomat serta warna yang terdeteksi dengan warna yang berbeda.



GAMBAR 5 CONTOH PENGUJIAN JUMLAH TOMAT TERDETEKSI.

TABEL 4 TABLE PENGUJIAN JUMLAH TOMAT

Jumlah tomat	keterangan
2 tomat	Tomat yang terdeteksi yaitu 2
4 tomat	Tomat dapat terdeteksi yaitu 4
8 tomat	Tomat yang terdeteksi yaitu 8
9 tomat	9 dari 10 tomat yang tidak terdeteksi hanya 1 tomat, karena tingkat akurasi warna nya kurang.
12 tomat	Tomat yang bisa terdeteksi dengan tinggi 35cm.

TABEL 5  
TABLE HASIL DATA SCV

frame	tomat matang	tomat hampir matang	tomat kurang matang	tomat mentah
0	0	0	0	0
1	3	1	0	0
2	3	1	0	0
3	3	1	0	0
4	3	1	0	0
5	3	1	0	0
6	3	1	0	0
7	2	1	0	0
8	3	1	0	0
9	3	1	0	0
10	4	1	0	0
11	4	1	0	0
12	4	1	0	0
13	4	1	0	0
14	3	1	0	0
15	4	1	1	0

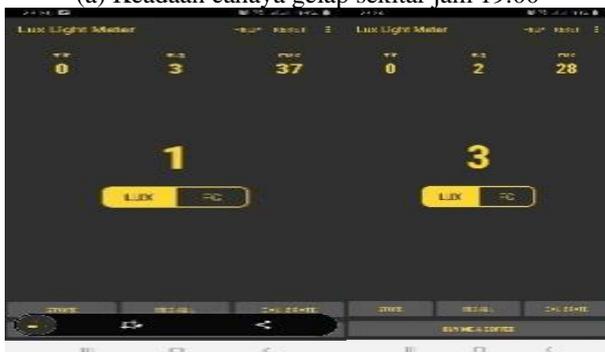
16	4	1	1	0
17	4	2	1	0
18	4	2	1	0
19	3	2	1	0
20	3	2	0	0
21	3	1	0	0
22	3	2	0	0
23	3	1	0	0
24	4	2	0	1
25	4	2	0	1
26	4	2	0	1
27	4	2	0	1
28	4	2	0	1
29	3	2	0	1
30	3	2	0	1
31	4	2	0	1
32	3	2	0	1

Setelah dilakukan pengujian dengan menggunakan banyak nya, alat ini dapat mendeteksi jumlah tomat serta warna yang terdeteksi dengan warna yang berbe.

c. Pengujian cahaya

Pengujian dilakukan di dalam ruangan dan luar ruangan. Alat ini diletakkan di dalam ruangan dan diluar ruangan untuk mengamati warna, sistem agar terdeteksi maka dilakukan pengujian sebanyak 45 kali deteksi cahaya ini menggunakan luxmeter.

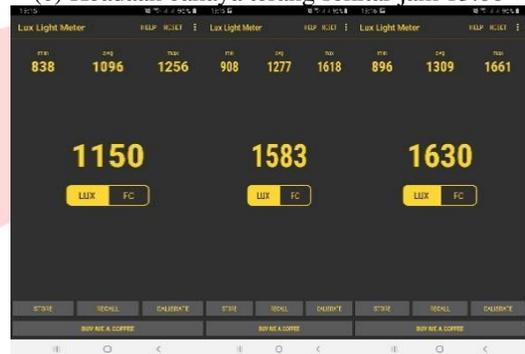
(a) Keadaan cahaya gelap sekitar jam 19.00



(b) Keadaan cahaya redup sekitar jam 17.00



(c) Keadaan cahaya terang sekitar jam 13.00



GAMBAR 6  
DETEKSI CAHAYA

TABEL 6  
PENGUJIAN CAHAYA

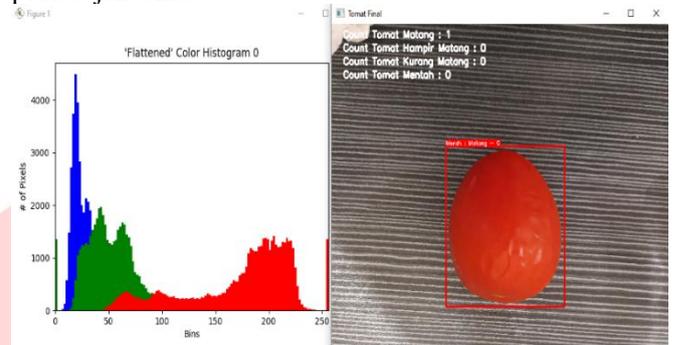
NO	Tingkat kematangan tomat	kondisi Cahaya dan Nilai lux meter dalam ruangan	keterangan	Nilai lux matter(lux) luar ruangan	keterangan
1	Matang	Gelap (3 lux)	Terdeteksi	Gelap (2 lux)	Terdeteksi
2		Gelap (3 lux)	Terdeteksi	Gelap (1 lux)	Tidak Terdeteksi
3		Gelap (3 lux)	Tidak Terdeteksi	Gelap (1 lux)	Terdeteksi
4		Gelap (3 lux)	Terdeteksi	Gelap (2 lux)	Terdeteksi
5		Gelap (3 lux)	Terdeteksi	Gelap (1 lux)	Tidak Terdeteksi
6	Matang	Redup (44 lux)	Terdeteksi	Redup (58 lux)	Terdeteksi
7		Redup (44 lux)	Terdeteksi	Redup (56 lux)	Terdeteksi
8		Redup (43 lux)	Terdeteksi	Redup (57 lux)	Terdeteksi
9		Redup (44 lux)	Terdeteksi	Redup (56 lux)	Terdeteksi
10		Redup (44 lux)	Terdeteksi	Redup (57 lux)	Terdeteksi
11	Matang	Terang (84 lux)	Terdeteksi	Terang (1155 lux)	Terdeteksi
12		Terang (84 lux)	Terdeteksi	Terang (1115 lux)	Terdeteksi
13		Terang (89 lux)	Terdeteksi	Terang (1119 lux)	Terdeteksi
14		Terang (84 lux)	Terdeteksi	Terang (1009 lux)	Terdeteksi
15		Terang (88 lux)	Terdeteksi	Terang (1134 lux)	Terdeteksi
16	Setengah matang	Gelap (2 lux)	Tidak Terdeteksi	Gelap (2 lux)	Terdeteksi
17		Gelap (3 lux)	Terdeteksi	Gelap (1 lux)	Tidak Terdeteksi
18		Gelap (2 lux)	Terdeteksi	Gelap (1 lux)	Tidak Terdeteksi
19		Gelap (3 lux)	Terdeteksi	Gelap (2 lux)	Terdeteksi
20		Gelap (3 lux)	Tidak Terdeteksi	Gelap (2 lux)	Terdeteksi
21	Setengah matang	Redup (44 lux)	Terdeteksi	Redup (56 lux)	Terdeteksi
22		Redup (44 lux)	Terdeteksi	Redup (57 lux)	Terdeteksi
23		Redup (43 lux)	Terdeteksi	Redup (56 lux)	Terdeteksi
24		Redup (44 lux)	Terdeteksi	Redup (57 lux)	Terdeteksi
25		Redup (44 lux)	Terdeteksi	Redup (56 lux)	Terdeteksi
26	Setengah matang	Terang (85 lux)	Terdeteksi	Terang (1121 lux)	Terdeteksi
27		Terang (85 lux)	Terdeteksi	Terang (1114 lux)	Terdeteksi
28		Terang (84 lux)	Terdeteksi	Terang (1120 lux)	Terdeteksi
29		Terang (82 lux)	Terdeteksi	Terang (1134 lux)	Terdeteksi
30		Terang (86 lux)	Terdeteksi	Terang (1141 lux)	Terdeteksi
31	Mentah	Gelap (2 lux)	Terdeteksi	Gelap (2 lux)	Terdeteksi
32		Gelap (3 lux)	Terdeteksi	Gelap (2 lux)	Terdeteksi
33		Gelap (3 lux)	Terdeteksi	Gelap (2 lux)	Terdeteksi
34		Gelap (3 lux)	Tidak Terdeteksi	Gelap (1 lux)	Tidak Terdeteksi
35		Gelap (3 lux)	Terdeteksi	Geap (2 lux)	Terdeteksi
36	Mentah	Redup (44 lux)	Terdeteksi	Redup (56 lux)	Terdeteksi
37		Redup (45 lux)	Terdeteksi	Redup (55 lux)	Terdeteksi
38		Redup (44 lux)	Terdeteksi	Redup (57 lux)	Terdeteksi
39		Redup (45 lux)	Terdeteksi	Redup (56 lux)	Terdeteksi
40		Redup (44 lux)	Terdeteksi	Redup (57 lux)	Terdeteksi
41	Mentah	Terang (86 lux)	Terdeteksi	Terang (1600 lux)	Tidak Terdeteksi
42		Terang (86 lux)	Terdeteksi	Terang (1690 lux)	Tidak Terdeteksi
43		Terang (84 lux)	Terdeteksi	Terang (1125 lux)	Terdeteksi
44		Terang (88 lux)	Terdeteksi	Tomat (1920 lux)	Tidak Terdeteksi
45		Terang (85 lux)	Terdeteksi	Tomat (1138 lux)	Terdeteksi

Pada table 6 Pengujian ini dilakukan pada jam 13.00 untuk mendeteksi cahaya terang di dalam ruangan dan di luar ruangan dengan mendapatkan nilai cahaya yang dapat mendeteksi warna yaitu 1009 lux sampai 1155 lux, untuk pengujian cahaya redup dilakukan didalam ruangan dan luar ruangan pada jam 17.00 dengan hasil cahaya yang didapat yaitu 55 lux sampai 58 lux masih dapat mendeteksi warna, untuk pengujian di cahaya gelap dilakukan di luar ruangan dan didalam ruangan pada jam 19.00 dengan nilai cahaya yang didapat yaitu 1 lux sampai 3 lux.

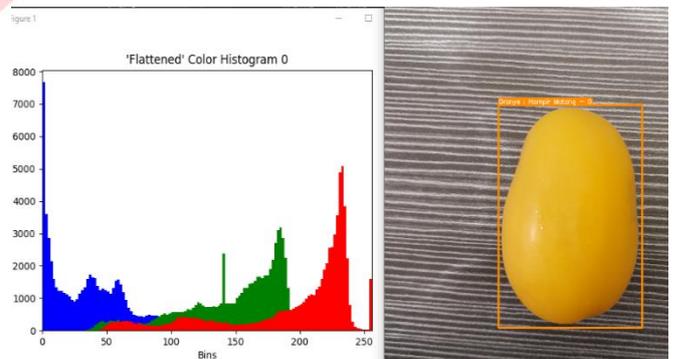
d. Histogram

Pengujian ini dilakukan dengan memasukan gambar tomat mulai dari tomat matang, hampir matang dan mentah lalu hasilnya bisa dilihat dari histogram. Dapat di lihat pada gambar 4.4 tomat matang menunjukkan grafik warna yang dimana hasil dari grafik warna nya semakin ke kanan itu

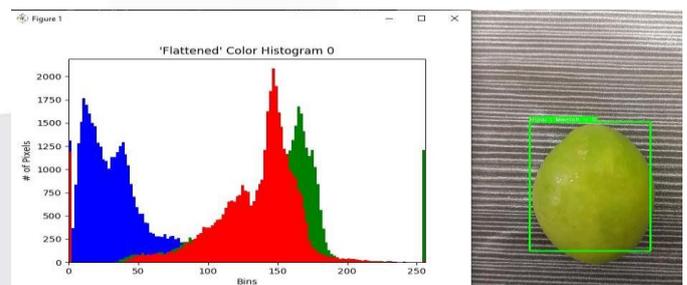
warna merah nya semakin terlihat dan juga kecerahan yang didapat cukup untuk mendeteksi warna pada tomat. untuk gambar 4.5 merupakan histogram tomat hampir matang yang menunjukkan grafik warna nya mendeteksi warna tomat tersebut dan juga kecerahan yang didapatkan cukup untuk mendeteksi warna tomatnya. Sedangkan pada gambar 4.6 yaitu histogram tomat mentah dengan grafik yang tinggi yang menunjukkan cahaya yang didapat tidak terlalu cerah dan tidak terlalu gelap dan juga warna yang didapat mampu terdeteksi pada objek tomat.



GAMBAR 7  
HISTOGRAM TOMAT MATANG



GAMBAR 8  
HISTOGRAM TOMAT HAMPIR MATANG



GAMBAR 9  
HISTOGRAM TOMAT MENTAH

IV. KESIMPULAN

Kesimpulan Berdasarkan hasil dari penelitian pada proyek akhir ini, dapat diambil kesimpulan sebagai berikut ini:

- A. Alat dan sistem yang dibuat untuk mendeteksi warna sudah dapat, dan jarak yang dapat mendeteksi warna yaitu 10 cm sampai 40 cm jika jaraknya 40 cm sampai 60 cm warna tidak terdeteksi.

- B. Sistem pendeteksi warna tomat menggunakan kamera dengan bantuan OpenCV telah berhasil dibangun dan dijalankan dengan baik
- C. Agar pendeteksi tomat berhasil maka dibutuhkan cahaya didalam ruangan yaitu 3 Lux (gelap) sampai 89 Lux(terang) sedangkan untuk cahaya yang dibutuhkan diluar ruangan.
- D. Sistem pendeteksi warna tomat menggunakan kamera dengan bantuan OpenCV telah berhasil dibangun dan dijalankan dengan baik.3 yaitu 2 Lux sampai 1155 Lux agar deteksi warna tomat berhasil.

Ada beberapa saran untuk pengembangan pada penelitian agar bisa lebih baik yaitu ditambahkannya sensor warna pada sistem deteksi agar dapat mengetahui warna lebih akurat dan konsisten. Kemudian, sistem mampu mengirimkan hasil data pada sebuah server yang terhubung pada kamera sehingga data dapat diakses secara mudah dimanapun dan kapanpun.

- [10] R. Asmara, “No 主観的健康感を中心とした在宅高齢者における健康関連指標に関する共分散構造分析Title,” *Rev. Bras. Ergon.*, vol. 3, no. 2, pp. 80–91, 2016, [Online]. Available: <https://www.infodesign.org.br/infodesign/article/view/355%0Ahttp://www.abergo.org.br/revista/index.php/ae/article/view/731%0Ahttp://www.abergo.org.br/revista/index.php/ae/article/view/269%0Ahttp://www.abergo.org.br/revista/index.php/ae/article/view/106>.
- [11] “Apa itu PyCharm? Mengenal PyCharm Python, Kekurangan, serta Kelebihannya | APPKEY.” <https://appkey.id/pembuatan-aplikasi/mobile-programming/pycharm-python/> (accessed Aug. 20, 2022).
- [12] H. Prabowo, “Deteksi Kondisi Kematangan Buah Jeruk Berdasarkan Kemiripan Warna Pada Ruang Warna RGB Berbasis Android,” *J. Elektron. Sist. Inf. dan Komput.*, vol. 3, no. 2, pp. 9–19, 2017.

#### REFERENSI

- [1] M. S. Nasution and N. Fadillah, “Deteksi Kematangan Buah Tomat Berdasarkan Warna Buah dengan Menggunakan Metode YCbCr,” *InfoTekJar (Jurnal Nas. Inform. dan Teknol. Jaringan)*, vol. 3, no. 2, pp. 147–150, 2019, doi: 10.30743/infotekjar.v3i2.1059.
- [2] N. Ihsan, “Automated System In Tomato Sorting With Image Processing Using RGB Detection Method,” 2016.
- [3] C. K. Gomathy, “Color detection using pandas and opencv,” *Int. J. Adv. Res. Comput. Commun. Eng. Vol.*, vol. 10, no. December, pp. 461–466, 2021.
- [4] I. Zulkarnain, M. Ramadhan, and B. Anwar, “Implementasi Alat Pendeteksi Warna Benda Menggunakan Fuzzy Logic dengan Sensor TCS3200 Berbasis Arduino,” *J. Teknol. Sist. Inf. dan Sist. Komput. TGD*, vol. 2, no. 2, pp. 106–117, 2019.
- [5] Y. Bili, E. Purba, N. F. Saragih, A. P. Silalahi, and S. Sitepu, “Perancangan Alat Pendeteksi Kematangan Buah Nanas Dengan Menggunakan Mikrokontroler Dengan Metode Convolutional Neural Network ( CNN ),” vol. 2, no. 1, pp. 13–21, 2022.
- [6] C. Wasonowati, “MENINGKATKAN PERTUMBUHAN TANAMAN TOMAT (*Lycopersicon esculentum*) DENGAN SISTEM BUDIDAYA HIDROPONIK,” *Agrovigor*, vol. 4, no. 1, pp. 21–28, 2011.
- [7] G. E. G., . R. Ruman M., and C. Setianingsih, “Perancangan dan Implementasi untuk Penyortiran Buah Tomat (*Lycopersicum Esculentum*) Dengan Metode Learning Vector Quantizationprototype,” vol. 4, no. 3, pp. 4177–4185, 2017.
- [8] R. D. Kusumanto and A. N. Tompunu, “PENGOLAHAN CITRA DIGITAL UNTUK MENDETEKSI OBYEK MENGGUNAKAN PENGOLAHAN WARNA MODEL NORMALISASI RGB,” 2011.
- [9] A. Gide, “濟無No Title No Title No Title,” *Angew. Chemie Int. Ed.* 6(11), 951–952., pp. 5–24, 1967.