

## **Identifikasi Tingkat Kematangan Buah Tomat Berdasarkan Ruang Warna HSV Menggunakan Algoritme K - *Nearest Neighbour***

AYUDIA ARIFAINI (G64130050), DEVIRA VRIZAL (G64130064), TRI SETIO BAKHTIAR R (G64130070), INDRI NURUL GAYATRI (G64130079), EXCLEZIA KRISTIANINGTYAS (G64130105)

Kelompok: 11, Lab: 2

### **PENDAHULUAN**

#### **Latar Belakang**

Tomat merupakan salah satu jenis buah pada sektor pertanian yang cukup populer karena memiliki banyak manfaat yang terkandung. Melihat fakta tersebut, petani berlomba-lomba meningkatkan produksi tomat dengan melakukan berbagai upaya. Salah satunya adalah dengan proses pemanenan buah yang dilakukan secara otomatis dengan robot pemanen seperti yang telah diterapkan di Jepang. Cara yang paling mudah untuk menentukan tingkat kematangan buah secara otomatis adalah dengan ekstraksi warna pada citra digital.

Pada penelitian ini, warna yang terekstraksi akan diklasifikasikan ke dalam kelas-kelas untuk menentukan tingkat kematangan buah tomat. Kelas-kelas tersebut ialah mentah, setengah matang, dan matang. Algoritme klasifikasi yang digunakan adalah K-*Nearest Neighbour* (KNN) berdasarkan nilai dari ruang warna HSV citra.

Penelitian sebelumnya yang dilakukan oleh Deswari *et al* (2013) yang berjudul “Identifikasi Kematangan Buah Tomat Menggunakan Metoda *Backpropagation*” memiliki tingkat keberhasilan identifikasi sebesar 71.76%. Menurut Deswari *et al* (2013), tingkat keberhasilan identifikasi tersebut dipengaruhi oleh faktor pencahayaan terhadap citra yang diidentifikasi. Oleh karena itu, penelitian ini dilakukan dengan cara melakukan *image enhancement* sebelum citra diklasifikasikan dengan algoritme KNN untuk mendapatkan tingkat keberhasilan identifikasi kematangan buah tomat yang lebih baik dibandingkan penelitian sebelumnya tersebut.

#### **Perumusan Masalah**

Rumusan masalah penelitian ini adalah menentukan metode yang akan diterapkan untuk mengidentifikasi tingkat kematangan buah tomat melalui pemrosesan citra digital ketika pengguna memasukkan citra tomat yang baru.

#### **Tujuan Penelitian**

Tujuan dari penelitian ini adalah untuk mengidentifikasi tingkat kematangan dari buah tomat dan memasukkan nilai tersebut ke dalam 1 dari 3 kelas yang telah ditentukan yaitu mentah, setengah matang, dan matang.

#### **Manfaat Penelitian**

Manfaat dari penelitian ini adalah membuat program yang dapat diimplementasikan pada sistem dengan bahasa pemrograman Python untuk menentukan tingkat kematangan buah tomat berbasis pemrosesan citra digital. Selain itu, penelitian ini diharapkan dapat memudahkan pengguna untuk mengetahui tingkat kematangan tomat.

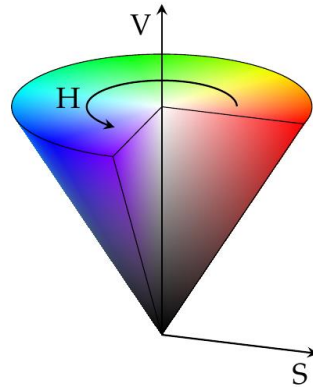
#### **Ruang Lingkup Penelitian**

Ruang lingkup penelitian ini adalah mengidentifikasi tingkat kematangan buah tomat pada citra digital yang telah dilakukan proses segmentasi.

## TINJAUAN PUSTAKA

### HSV (*Hue Saturation Value*)

HSV merupakan karakter warna berdasarkan *hue*, *saturation* dan *value*. Ruang warna HSV jika dimodelkan berbentuk prisma atau disebut *hex-cone* yang dapat dilihat pada Gambar 1.



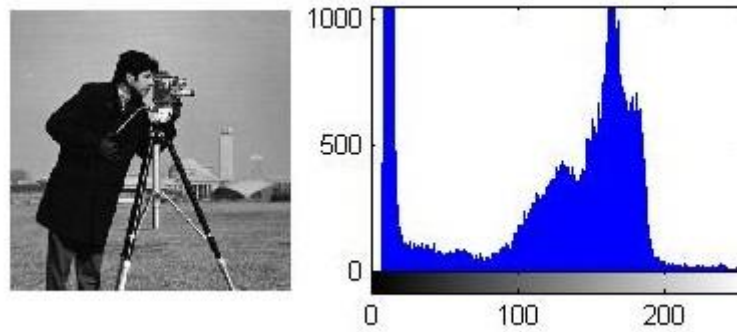
Gambar 1 Ilustrasi ruang warna HSV

Pada Gambar 1 dapat dilihat bahwa bagian tengah alas prisma berwarna putih dan bagian sisi prisma terdapat warna yang merupakan turunan dari kubus ruang warna RGB. Semakin mengerucut maka warna akan semakin gelap bahkan mendekati hitam. Menurut PM dan Chezian (2013) ruang warna HSV dapat diuraikan sebagai berikut.

- 1) *Hue* merupakan warna yang dapat dilihat oleh mata manusia yaitu merah, jingga, kuning, hijau, biru, nila, dan ungu. *Hue* dalam ruang warna HSV memiliki rentang nilai 0-360.
- 2) *Saturation* merupakan nilai intensitas atau kemurnian sebuah warna. Warna yang memiliki nilai *saturation* yang kecil maka warna semakin mendekati warna abu-abu, sedangkan warna yang memiliki nilai *saturation* yang tinggi maka warna akan semakin mendekati warna asli. Dalam ruang warna HSV, *saturation* memiliki rentang nilai 0-100%.
- 3) *Value* merupakan nilai tingkat gelap dan terang sebuah warna. Warna yang gelap memiliki *value* yang kecil, sedangkan warna yang terang memiliki *value* yang tinggi. Jika warna memiliki *value* sebesar 0, warna menjadi berwarna hitam tanpa memperhatikan nilai *hue* dan *saturation*. Dalam ruang warna HSV, *value* memiliki rentang nilai 0-100%.

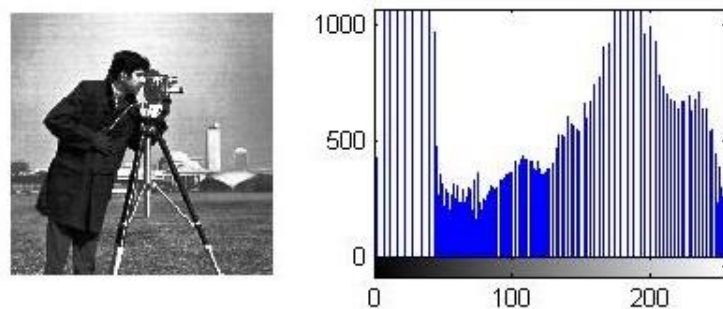
### Histogram

Histogram merupakan sebuah fungsi diskret pada citra digital dengan rentang tingkat intensitas  $[0, L-1]$ . Histogram dapat dinyatakan pada persamaan  $h(r_k) = n_k$  dengan  $r_k$  merupakan nilai dari intensitas pada tingkat  $k$  dan  $n_k$  merupakan jumlah pixel pada citra dengan intensitas  $r_k$ . Histogram dapat digunakan sebagai dasar untuk melakukan *image enhancement* seperti *histogram equalization* (Gonzalez dan Woods 2004). Contoh histogram sebuah citra dapat dilihat pada Gambar 2.



Gambar 2 Citra dan *histogram*

Histogram *equalization* merupakan teknik *image enhancement* untuk memperbaiki kontras pada citra dengan cara meratakan nilai histogram keseluruhan nilai keabuan yang digunakan sehingga citra yang terlalu gelap menjadi lebih terang, sedangkan citra yang terlalu terang menjadi lebih gelap (Maravi dan Sharma 2013). Contoh histogram setelah *histogram equalization* dapat dilihat pada Gambar 3



Gambar 3 Citra dan *histogram* setelah *histogram equalization*

### Klasifikasi

Klasifikasi merupakan proses pencocokan data berdasarkan kemiripan atribut data. Klasifikasi dibagi menjadi dua tipe yaitu *supervised* (klasifikasi terawasi) dan *unsupervised* (klasifikasi tak terawasi). Klasifikasi *unsupervised* merupakan bentuk pengklasifikasian terhadap hasil akhir dengan didasari pada analisis perangkat lunak terhadap data tanpa menyediakan contoh-contoh kelas terlebih dahulu (Darmawan). Sedangkan *supervised* merupakan metode untuk menemukan hubungan antar atribut data masukkan dengan atribut data *target*. Nilai atribut *target* telah diketahui sebelumnya. Jika hubungan antar data masukkan dan data *target* sudah ditemukan maka dapat dibuat sebuah model. Model tersebut dapat membantu dalam menjelaskan peristiwa yang tersembunyi atau untuk memprediksi nilai dari atribut target setelah diketahui nilai atribut data masukkan (Maimon dan Rokach 2005).

### K - Nearest Neighbour

K - Nearest Neighbour (KNN) merupakan salah satu contoh algoritme dengan metode klasifikasi yang memiliki ide bahwa objek-objek yang berdekatan akan memiliki karakteristik yang serupa. Sehingga, dengan mengetahui karakteristik dari suatu objek, karakteristik objek lain yang bertetangga dengannya dapat diprediksi. Hal ini berdasarkan ide bahwa sebuah objek baru dapat diklasifikasikan sesuai dengan jumlah *voting* terbanyak dari  $k$  buah objek tetangga. Jarak antara objek dihitung berdasarkan jarak *Euclidean* yang dapat dilihat pada persamaan sebagai berikut.

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

Namun, penggunaan jarak *Euclidian* mempunyai permasalahan dari menggunakan jarak *Euclidean* adalah nilai-nilai yang memiliki frekuensi besar akan menutupi nilai-nilai yang lebih kecil. Untuk mengatasi permasalahan tersebut, dilakukan normalisasi sehingga semua atribut memiliki pengaruh yang sama pada ukuran jarak antar objek (Khamis, Cheruiyot, dan Kimani 2014).

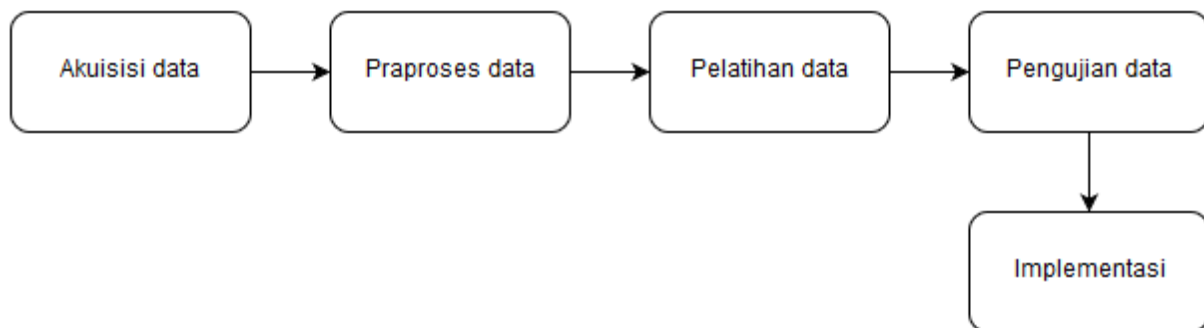
## METODE

### Data Penelitian

Dapat penelitian ini berupa citra tomat mentah, tomat setengah matang, dan tomat matang yang didapatkan dari situs <http://www.images.google.com>.

### Tahapan Penelitian

Penelitian identifikasi tingkat kematangan buah Tomat memiliki 5 tahapan yaitu akuisisi data, praproses data, pelatihan data, pengujian data, dan implementasi. Tahapan penelitian identifikasi tingkat kematangan buah tomat dapat dilihat pada Gambar 4.



Gambar 4 Tahapan penelitian identifikasi tingkat kematangan buah tomat.

### Akuisisi data

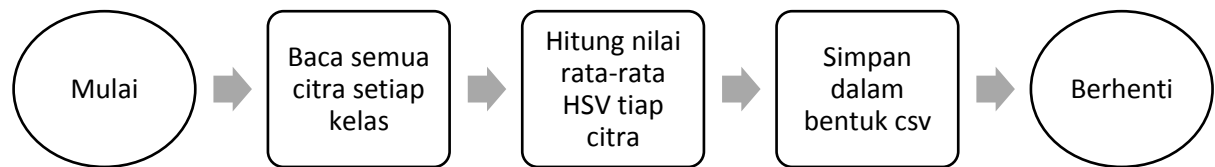
Akuisisi data dilakukan dengan cara mengunduh citra dari Google sebanyak 50 citra tomat matang, 50 citra tomat setengah matang, dan 50 citra tomat mentah.

### Praproses data

Praproses data dibagi menjadi dua tahap yaitu *image enhancement* dan pengubahan format ruang warna citra. Tahap pertama ialah *image enhancement*, pada tahap ini dilakukan proses berupa *histogram equalization* yang bertujuan untuk memperbaiki kontras citra sehingga kualitas citra meningkat sebelum dilakukan pelatihan dan pengujian. Tahap kedua ialah pengubahan *format* ruang warna citra dari RGB menjadi HSV.

### Pelatihan data

Pelatihan data dilakukan dengan cara mencari nilai rata-rata *hue*, *saturation*, dan *value* pada setiap citra data latih yang kelasnya telah ditentukan sebelumnya. Setelah itu, hasil pelatihan data disimpan dalam bentuk csv. Tahapan pelatihan data dapat dilihat pada Gambar 5.



Gambar 5 Tahapan pelatihan data.

### Pengujian data

Pengujian data dilakukan menggunakan algoritme KNN. Tahapan pertama yang dilakukan ialah mencari nilai rata-rata *hue*, *saturation*, dan *value* pada setiap citra data uji. setelah itu, dihitung jarak *Euclidian* citra data uji dengan citra data latih. Tahapan kedua, dilakukan proses pengurutan berdasarkan N data nilai jarak *Euclidian* terkecil. Setelah itu, hasil klasifikasi dihitung berdasarkan kelas terbanyak yang dipilih oleh N data nilai jarak *Euclidian* terkecil sehingga akurasi klasifikasi dapat dihitung.

### Implementasi

Implementasi menggunakan sistem dengan bahasa pemrograman Python. Sistem ini bekerja dengan cara seorang pengguna memasukkan data berupa citra tomat yang telah disegmentasi kemudian sistem akan melakukan proses klasifikasi. Setelah proses klasifikasi selesai, sistem akan menampilkan tingkat kematangan buah tomat.

### Lingkungan Pengembangan

Lingkungan pengembangan berupa perangkat keras dan perangkat lunak yang digunakan untuk penelitian ini adalah:

- 1 Laptop Asus X450CP dengan spesifikasi RAM 4GB dan prosesor Intel(R)Core(TM) i3-3217U CPU@1.80GHz
- 2 Sistem operasi Windows 8.1 pro 64-bit
- 3 Python 2.75
- 4 Numpy 1.11.2
- 5 Argparse
- 6 OpenCV 3.1.0

## HASIL PENELITIAN

### Akuisisi data

Pada tahap akuisisi data, didapatkan citra tomat yang telah tersegmentasi dari objek-objek lainnya. Citra tomat yang akan diproses hanya berisi satu objek buah tomat seperti yang dapat dilihat pada Gambar 6.



Gambar 6 Citra tomat hasil akuisisi data

### Praproses data

Pada tahap ini, akan dilakukan proses *image enhancement* pada citra tomat yang didapatkan dari proses akuisisi data. Metode *image enhancement* yang dilakukan ialah histogram *equalization* yang bertujuan untuk meratakan kontras pada citra. Hasil praproses data dapat dilihat pada Gambar 7.



Gambar 7 Citra tomat setelah dilakukan proses histogram *equalization*

Pada Gambar 7 dapat dilihat bahwa citra tomat menjadi lebih kontras. Setelah itu dilakukan proses transformasi dari ruang warna *RGB* menjadi ruang warna *HSV* dengan persamaan sebagai berikut:

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

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

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

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

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

### Pelatihan data

Pada tahap pelatihan data akan dicari nilai rata-rata *hue*, *saturation*, dan *value* pada setiap citra data latih yang kelasnya telah ditentukan sebelumnya. Berikut merupakan contoh perhitungan pencarian nilai rata-rata *hue*, *saturation*, dan *value* dari *folder test* dengan nama file 2.jpg.

$$\bar{H} = \frac{\sum H}{p \times l} = \frac{5\,287\,608}{276 \times 309} = 62$$

$$\bar{S} = \frac{\sum S}{p \times l} = \frac{12\,280\,896}{276 \times 309} = 144$$

$$\bar{V} = \frac{\sum V}{p \times l} = \frac{11\,939\,760}{276 \times 309} = 140$$

Hasil yang diperoleh dari proses pelatihan data ini kemudian dimasukkan ke dalam *list* kemudian disimpan dalam format CSV. Contoh tabel hasil pelatihan data dapat dilihat pada Tabel 1.

Tabel 1 Hasil pelatihan data

No	Hue	Saturation	Value	Kelas
1	109	185	168	mentah
2	62	144	140	mentah
.....	.....	.....	.....	.....
42	157	203	212	setengah matang
43	59	133	168	setengah matang
.....	.....	.....	.....	.....
119	36	58	191	matang
120	37	60	190	matang

### Pengujian data

Pada tahap pengujian data dilakukan proses klasifikasi dengan algoritme KNN. Tahap pertama yang dilakukan terlebih dahulu ialah mencari nilai rata-rata *hue*, *saturation*, dan *value* pada citra data uji. Setelah nilai tersebut didapatkan, dilakukan proses perhitungan jarak *Euclidean* antara citra data uji dengan citra data latih. Contoh perhitungan menggunakan citra dengan nama *file 2.jpg* yang terdapat pada *folder test* dapat dilihat pada Tabel 2.

Tabel 2 Hasil pelatihan data

No	Hue	Saturation	Value	Jarak Euclidian
1	109	185	168	122
2	62	144	140	107
.....	.....	.....	.....	.....
119	36	58	191	65
120	37	60	190	64

Setelah nilai jarak *Euclidian* didapatkan, nilai tersebut akan diurutkan sebanyak K buah data nilai terkecil. Hasil pengurutan dengan  $K = 7$  dapat dilihat pada Tabel 3.

Tabel 3 Hasil pengurutan jarak *Euclidian*

No	Jarak Euclidian	Kelas
1	10	matang

2	16	matang
3	26	matang
4	28	setengah-matang
5	30	matang
6	31	setengah-matang
7	32	setengah-matang

Selanjutnya, citra akan diklasifikasikan berdasarkan kelas yang terbanyak dipilih oleh K buah data tetangga. Pada Tabel 3 dapat dilihat bahwa kelas matang sebanyak 4 dan kelas setengah-matang sebanyak 3 sehingga citra dengan nama *file 2.jpg* diklasifikasikan sebagai tomat matang. Proses ini berlangsung hingga semua data uji telah dilakukan proses klasifikasi. Setelah proses pengujian data dilakukan, didapatkan *Confusion matrix* hasil klasifikasi yang dapat dilihat pada Tabel 4, Tabel 5, dan Tabel 6.

Tabel 4 *Confusion matrix* dengan K = 5

Kelas	Mentah	Setengah-matang	Matang
Mentah	10	0	0
Setengah-matang	1	9	0
matang	0	0	10

Tabel 5 *Confusion matrix* dengan K = 7

Kelas	Mentah	Setengah-matang	Matang
Mentah	10	0	0
Setengah-matang	1	9	0
matang	0	0	10

Tabel 6 *Confusion matrix* dengan K = 9

Kelas	Mentah	Setengah-matang	Matang
Mentah	10	0	0
Setengah-matang	1	9	0
matang	0	0	10

Dapat dilihat pada Tabel 4, Tabel 5, dan Tabel 6 bahwa hasil klasifikasi dengan nilai K yang berbeda memiliki hasil yang serupa. Hasil yang diberikan ialah tingkat akurasi sebesar 96.67%. Oleh sebab itu, penggunaan nilai K dari salah satu nilai tersebut tidak masalah. Namun, algoritme ini memiliki kelemahan jika citra tomat hasil segmentasi tidak sempurna yang dapat dilihat pada Gambar 8.



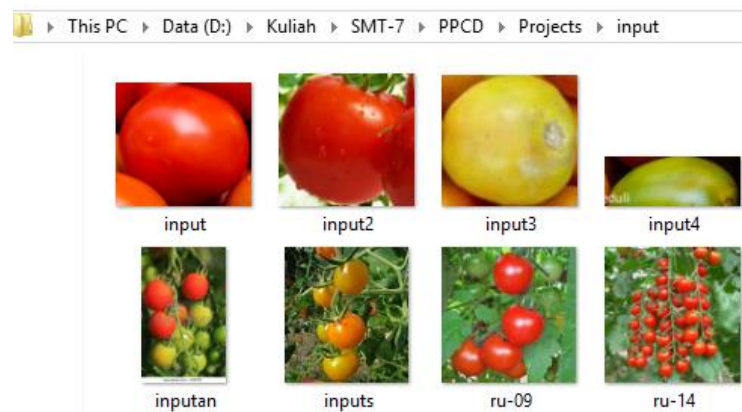
Gambar 8 Citra tomat yang tidak tersegmentasi sempurna



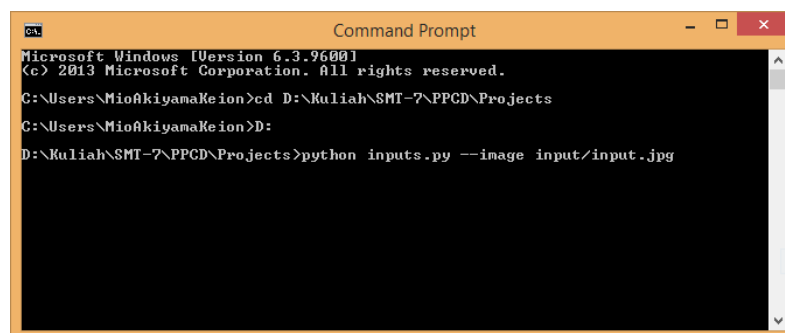
Pada Gambar 8 terlihat bahwa terdapat satu objek tomat setengah matang yang dikelilingi tomat mentah. Citra tersebut salah diklasifikasikan yang seharusnya masuk sebagai kelas setengah matang namun terdeteksi sebagai kelas mentah. Hal ini disebabkan karena warna hijau pada Gambar 8 lebih dominan sehingga nilai rata-rata menjadi lebih bergeser ke warna hijau.

## Implementasi

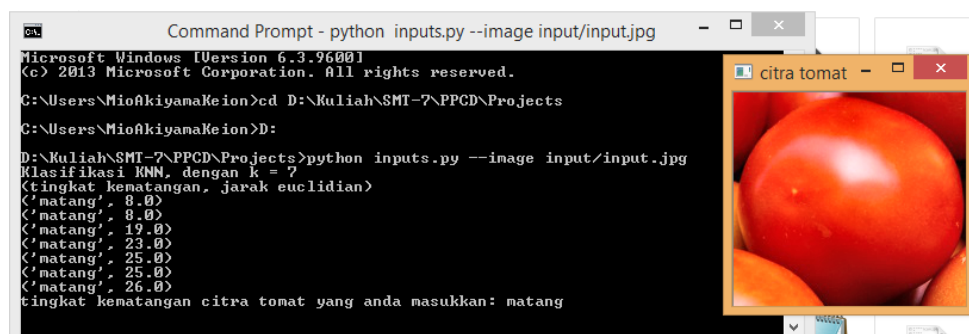
Tahap ini seorang pengguna dapat mengidentifikasi tingkat kematangan sebuah citra tomat yang telah tersegmentasi. Pengguna terlebih dahulu memasukkan gambar pada folder input. Kemudian, seorang pengguna membuka Command Prompt pada direktori tempat sistem ini. Setelah itu, seorang pengguna menjalankan dengan cara memasukkan *command* ke cmd dengan perintah `python inputs.py --image direktori_ke_file`. Setelah itu, sistem akan menampilkan tingkat kematangan buah tomat. Hasil *implementasi* dari penelitian ini yang dapat dilihat pada Gambar 9, Gambar 10, dan Gambar 11.



Gambar 9 Folder *input* tempat citra tomat



Gambar 10 Tampilan Command Prompt



Gambar 11 Tampilan hasil identifikasi tingkat kematangan

## SIMPULAN DAN SARAN

Berdasarkan hasil penelitian ini, disimpulkan bahwa penggunaan algoritme *k - nearest neighbour* untuk mengidentifikasi tingkat kematangan buah tomat sudah cukup baik dengan tingkat akurasi 96.67%. Meskipun demikian, algoritme ini mempunyai kelemahan ketika citra hasil segmentasi tomat tidak sempurna sehingga identifikasi tingkat kematangan menjadi tidak akurat. Hal itu disebabkan karena penggunaan nilai rata-rata sebagai penciri.

## DAFTAR PUSTAKA

- Gonzalez RC, Woods RE. 2008. *Digital Image Processing Third Edition*. New Jersey: Pearson Education.
- Khamis HS, Cheruiyot KW, Kimani S. 2014. *Application of k - Nearest Neighbour Classification in Medical Data Mining*. 4(4): 122.
- Maimon O, Rokach L. 2005. *Data Mining and Knowledge Discovery Handbook*. Israel: Springer International Publishing AG
- Maravi YPS, Sharma S. 2013. *A Comparative Study of Histogram Equalization Based Image Enhancement Techniques for Brightness Preservation and Contrast Enhancement*. 4(5):12-13.
- Nishad, Chezian RM. 2013. *Various Colour Spaces and Colour Space Conversion Algorithms*. 4(1): 46.