

# Segmentasi Penyakit pada Citra Daun Tomat dengan Metode *K-Means Cluster* melalui Ruang Warna HSV

Abdi Negara Guci<sup>a</sup>, Ahmad Hanafi Prasetyo<sup>a</sup>, Alexander Victorius Budianto<sup>a</sup>,

Haidar Ahmad.Fajri<sup>a\*</sup>, dan Safrizal Ardana Ardiyansa<sup>a</sup>

<sup>a</sup>Departemen Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam,

Universitas Brawijaya, Malang, Indonesia

\*Email: [haidaraf6@student.ub.ac.id](mailto:haidaraf6@student.ub.ac.id)

---

## Abstrak

Tomat merupakan salah satu jenis sayuran yang banyak dikonsumsi oleh manusia di seluruh dunia. Daun menjadi indikasi kesehatan pada tanaman tomat. Daun yang sehat adalah daun yang berwarna hijau terang. Pengecekan kesehatan setiap tanaman tomat secara manual akan menjadi sulit, membutuhkan waktu yang lama, dan tenaga pekerja yang banyak. Metode *k-means* merupakan salah metode segmentasi citra pada pengolahan citra digital yang cepat dan otomatis. Metode *k-means* menghasilkan segmentasi daerah yang sehat dan daerah yang sakit. Penelitian ini menggunakan analisis kualitatif dengan pengukuran validasi *overlap*, yaitu mengukur rasio perbandingan dari dua hasil segmentasi citra baik secara manual, metode *k-means*, dan kemiripannya. Komponen warna *hue* pada citra hasil *contrast stretching* pada ruang warna HSV dapat digunakan untuk melakukan segmentasi terhadap daerah sehat dan daerah sakit pada citra. Nilai piksel *hue* pada rentang 80 hingga 175 yang disertai dengan metode *k-means* dapat mendeteksi ketiga bagian citra dengan tingkat akurasi yang tinggi. Tingkat akurasi yang dihasilkan berada di atas 90%. Pada tingkat presisi, sensitivitas atau *f1 score* juga menghasilkan nilai diatas 75%.

**Kata kunci:** tomat, *k-means*, sakit, sehat, akurasi, presisi

---

## 1. Pendahuluan

Makanan adalah salah satu hal terpenting yang dibutuhkan oleh setiap manusia. Makanan dibutuhkan sebagai sumber energi yang digunakan untuk melakukan berbagai hal seperti sebagai sumber energi, menjaga metabolisme tubuh, dan menjaga kekebalan tubuh. Masalah kelaparan merupakan salah satu akibat dari tidak adanya

makanan yang memiliki cukup energi untuk tetap menjalankan metabolisme tubuh, sehingga diperlukan persediaan makanan yang cukup untuk mencegah terjadinya kelaparan.

Kesehatan tanaman tomat adalah sesuatu yang sangat penting dan perlu diperhatikan agar produksi tomat di seluruh dunia dapat terjaga. Produksi tanaman tomat dipengaruhi oleh adanya patogen yang menginfeksi tanaman tomat. Patogen yang menginfeksi pada tanaman diantaranya seperti *Ralstonia solanacearum*, *Alternaria linariae*, *Phytophthora infestans*, *Septoria lycopersici*, dan lain-lain. Patogen tersebut menjadi perhatian khusus karena akan mempengaruhi kondisi tanaman tomat.

Kondisi keadaan daun pada tanaman dapat menjadi indikasi kesehatan pada suatu tanaman. Menurut (Gardening Know How, 2020), daun yang sehat memiliki warna hijau cerah dan merata, kecuali pada daun yang memiliki dua warna atau lebih. Kondisi daun juga dapat merepresentasikan apakah tanaman apakah diserang hama atau tidak. Pengecekan kondisi daun tomat dapat dilihat dari warna daunnya, adanya lubang atau tidak, dan kering atau tidaknya daun tersebut.

Pengecekan daun tomat tidak menjadi masalah apabila tanaman tomat yang tersedia jumlahnya sedikit, tetapi akan menimbulkan masalah jika jumlahnya sangat banyak. Pengecekan kesehatan setiap tanaman tomat secara manual akan menjadi sulit, membutuhkan waktu yang lama, dan tenaga pekerja yang banyak. Pengolahan citra digital merupakan solusi dari permasalahan tersebut. Program pada pengolahan citra dapat mendeteksi kondisi daun tomat secara otomatis dan cepat. Program tersebut bekerja dengan cara menginputkan citra daun tomat, kemudian dideteksi apakah daun tomat sehat atau memiliki penyakit.

Metode segmentasi citra *k-means* yang diterapkan pada ruang warna HSV, merupakan salah satu metode yang dapat membedakan kondisi tanaman daun yang sehat atau mencari bagian daun yang sakit. *K-means* merupakan metode segmentasi yang cepat dan mudah diimplementasikan. Hasil dari segmentasi menggunakan *k-means* dapat digunakan dalam analisa pengecekan daerah pada daun tomat sehingga daerah terbagi menjadi tiga bagian yaitu *background* citra, bagian daun yang sehat, dan

bagian daun yang sakit. Citra daun yang memiliki bagian sakit, menjadi indikasi bahwa tanaman tomat tersebut sedang sakit, sehingga dapat melakukan penanganan pada tanaman tersebut.

## **2. Metodologi**

### **2.1 Sumber Data**

Sumber data yang digunakan dalam penelitian ini adalah citra daun tomat berjumlah enam dan citra hasil segmentasi dari keenam citra tersebut melalui Photoshop yang dibagi menjadi tiga bagian, yaitu *background* citra, bagian daun sehat, dan bagian daun sakit. Sumber data tersebut akan digunakan sebagai *dataset* pada program pengolahan citra. Keenam *dataset* terbagi menjadi tiga jenis penyakit yang berbeda. Suatu penyakit diwakili oleh dua buah citra.

Citra yang digunakan diasumsikan memiliki warna *background* yang sama dan memiliki format jpg dengan ukuran citra yang berbeda-beda. Selain itu diasumsikan bahwa daun citra merupakan daun yang sudah dewasa dan seluruh bagian daun tomat termuat dalam citra, sehingga pada tepian citra sebagian besar merupakan *background*. Keenam citra tersebut memiliki nama file yang bersesuaian dengan jenis penyakit pada citra. Ketiga jenis penyakit citra tersebut adalah *tomato septoria leaf spot*, *tomato leaf mosaic virus*, dan *tomato leaf bacterial spot*. Citra tersebut diperoleh dari *fact sheet* yang telah disimpan pada google drive dalam link [s.ub.ac.id/DataSetAwal](https://s.ub.ac.id/DataSetAwal).

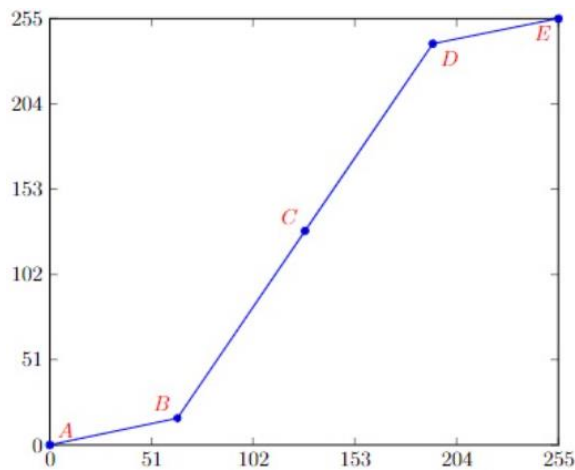
### **2.2 Preprocessing Citra**

#### **2.2.1 Contrast Stretching**

Persebaran warna pada citra mempengaruhi kejelasan atau kualitas gambar pada citra. Citra dengan kualitas tinggi memiliki distribusi terang dan gelap yang merata. Distribusi warna tersebut biasanya disebut dengan *contrast*. *Contrast* dapat mempengaruhi kualitas pada citra, sehingga menjadi salah satu komponen yang penting dalam citra. Kualitas citra yang baik akan memudahkan program pengolahan citra mengenali bagian-bagian pada citra. Semakin merata persebaran *contrast* dan intensitas warna, maka akan semakin mudah bagian-bagian pada citra dikenali oleh program.

*Contrast stretching* merupakan salah satu teknik yang berguna untuk memperbaiki kualitas citra dengan cara meregangkan rentang nilai intensitas citra. Keempat citra daun tomat yang menjadi *dataset* masih memiliki distribusi warna yang tidak merata. Sehingga citra tersebut harus dilakukan proses *contrast stretching*.

Peregangan nilai intensitas citra dinyatakan sebagai fungsi garis lurus dalam interval 0 sampai 255 yang sepotong-potong dan melewati berhingga titik. Nilai intensitas awal dinyatakan dalam sumbu-x dan nilai intensitas hasil *contrast stretching* dinyatakan pada sumbu-y. Pada penelitian ini, himpunan titik-titik yang dipilih agar dapat menghasilkan citra dengan distribusi warna yang baik adalah  $\{(0, 0); (64, 16); (128, 128); (192, 240); (255, 255)\}$ . Berikut merupakan grafik fungsi linear yang melewati titik-titik tersebut.



Gambar 1. Fungsi linear pada proses *contrast stretching*

### 2.2.1 Normalisasi

Normalisasi citra adalah proses membagi semua nilai intensitas pada setiap piksel pada citra dengan 255. Normalisasi citra bertujuan agar setiap nilai pada piksel berada pada interval 0 sampai 1. Nilai piksel dalam rentang tersebut, memudahkan kita untuk membuat diagram sebagai visualisasi penyebaran warna pada citra. Diagram yang digunakan sebagai visualisasi berjenis diagram titik atau yang biasanya disebut dengan *scatter plot*. Agar menghasilkan bentuk visualisasi yang baik, penyebaran warna citra dinyatakan dalam dua *scatter plot*. Pada *scatter plot* pertama, sumbu-x dan sumbu-y

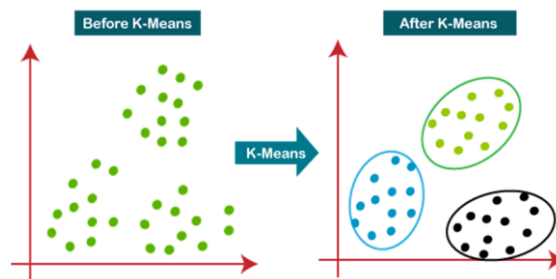
berturut-turut menyatakan nilai intensitas warna merah dan hijau, sedangkan pada *scatter plot* kedua, berturut-turut menyatakan nilai intensitas warna merah dan biru. Kedua diagram tersebut diperlukan karena citra yang diolah merupakan citra dalam ruang warna RGB yang berdimensi tiga.

### 2.3 Pemrosesan Citra

Citra dilakukan proses segmentasi dengan metode *k-means* terlebih dahulu agar didapatkan nilai piksel yang tidak bervariasi. Setelah itu, dilakukan proses penghapusan *background*. Citra tanpa *background* selanjutnya diubah ke dalam ruang warna HSV. Pada ruang warna *hue*, terlihat bahwa bagian daun pada citra yang sakit memiliki nilai intensitas *hue* yang tinggi atau nilai intensitas *hue* yang rendah. Setelah itu, didapatkan bagian daun yang berpenyakit. Selanjutnya, citra sehat didapatkan dari hasil pengurangan antara citra setelah dilakukan pra proses dengan citra bagian daun yang sakit.

#### 2.3.1 Segmentasi *K-means*

Segmentasi *k-means* adalah salah satu metode sederhana yang dapat digunakan untuk melakukan pengelompokan data. Setelah dilakukan proses segmentasi *k-means*, maka pada *scatter plot* akan terlihat sekumpulan warna yang dikelompokkan dalam *cluster* yang sama seperti pada gambar di bawah.



Gambar 2. Visualisasi persebaran warna setelah proses segmentasi

Tujuan dari segmentasi atau *clustering* dalam pengolahan citra ini adalah menyamakan nilai piksel pada masing-masing komponen warna RGB pada bagian *background*. Apabila nilai intensitasnya telah sama, maka nilai pada intensitas tersebut dapat diubah menjadi warna biru. Pengubahan warna *background* menjadi warna biru dilakukan karena warna biru merupakan warna yang tidak dijumpai pada citra asal. Selain itu, pengubahan warna ini juga bertujuan untuk menggambarkan citra tanpa *background*.

### 2.3.2 Penghapusan *Background*

Mengingat *dataset* yang digunakan merupakan citra daun yang tidak terpotong, maka keseluruhan bagian daun akan termuat dalam citra. Sehingga nilai modus dapat dimanfaatkan dalam melakukan penghapusan *background* dalam pengolahan citra ini. Nilai modus yang dimaksud merupakan nilai piksel pada tepian citra (bukan tepian daun) yang paling sering muncul.

Hal yang pertama dilakukan adalah membuat tiga *list* kosong, kemudian ketiga *list* tersebut akan diisi dengan nilai intensitas pada ruang warna R, G, dan B. Anggota pada ketiga *list* tersebut diperoleh dari mengoleksi nilai-nilai intensitas pada tiga bagian baris teratas, tiga bagian baris terbawah, tiga kolom paling kiri, dan tiga kolom paling kanan citra. Selanjutnya dari ketiga *list* tersebut akan dicari masing-masing modulusnya. Setelah didapatkan modulusnya, maka semua nilai piksel pada citra hasil segmentasi dengan *k-means* yang mempunyai selisih tidak jauh dari modulusnya akan diubah menjadi warna biru, sehingga akan didapatkan citra baru tanpa *background* atau citra baru dengan *background* berwarna biru.

### 2.3.3 Pemilihan Rentang Komponen *Hue*

Citra tanpa *background* kemudian diubah ke dalam ruang warna HSV, sehingga didapatkan empat buah citra yaitu citra dalam ruang HSV, citra dalam komponen *hue*, citra dalam komponen *saturation*, dan citra dalam komponen *value*. Selanjutnya, keempat citra tersebut diamati secara manual dan dipilih salah satu dari keempat citra yang paling menonjol dalam menunjukkan perbedaan daerah berpenyakit, daerah yang sehat, dan daerah *background* citra. Berdasarkan pengamatan, didapatkan citra yang paling menonjol adalah citra komponen *hue*.

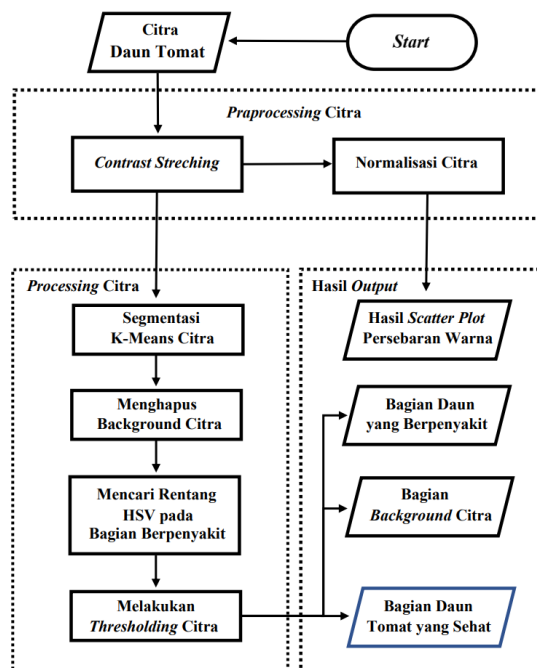
Pada ruang warna *hue*, terlihat bahwa pada *background* citra memiliki nilai intensitas *hue* yang rendah. Kemudian pada bagian daun yang sehat atau berwarna hijau, memiliki nilai intensitas *hue* yang tinggi. Akibatnya dapat diduga bahwa bagian daun yang tidak sehat memiliki nilai intensitas *hue* yang tidak terlalu tinggi (tidak mendekati 255) atau tidak terlalu rendah (tidak mendekati nol), dengan kata lain nilainya dapat dinyatakan dalam suatu rentang atau interval.

Metode *trial and error* adalah salah satu metode untuk menemukan rentang nilai *hue* yang mampu mendeteksi bagian yang sakit. Setelah dilakukan coba-coba, akan didapatkan interval terbaik yaitu [80, 175] yang merupakan rentang nilai intensitas *hue* pada bagian daun yang berpenyakit. Akibatnya bagian daun yang sehat dan *background*, memiliki nilai *hue* kurang dari 80 dan lebih dari 175.

### 2.3.4 Melakukan *Thresholding* Citra

Setelah program mampu membedakan bagian daun yang sehat, bagian daun yang sakit, dan bagian *background*, maka langkah selanjutnya adalah melakukan *thresholding* pada citra. *Thresholding* ini dilakukan agar seluruh citra hanya memiliki tiga jenis warna yang mewakili setiap bagian. Ketiga jenis warna yang digunakan adalah warna biru yang menyatakan bagian *background*, warna merah yang menyatakan bagian daun berpenyakit, dan warna hijau yang menyatakan bagian daun yang sehat.

Tujuan dari melakukan *thresholding* citra adalah agar citra hasil *output* pada program dapat dibandingkan dengan citra hasil pendeteksian bagian daun berpenyakit dengan cara manual, sehingga dapat diperoleh tingkat akurasi dan tingkat kesalahan dari program pendeteksian bagian berpenyakit pada daun tomat.



Gambar 3. Diagram alir proses pengolahan citra

## 2.4 Teknik Analisis Data

Teknik analisis data yang digunakan adalah analisis kualitatif dengan pengukuran validasi *overlap*, yaitu mengukur rasio perbandingan dari dua hasil segmentasi citra baik secara manual dan secara metode *k-means* dan kemiripannya. Variabel yang dilibatkan untuk mengukur analisis kualitatif dalam penelitian ini adalah tingkat akurasi, presisi, *recall*, dan *f1 score* yang diperoleh dari *confusion matrix*.

### 2.4.1 Confusion Matrix

*Confusion matrix* adalah matriks yang digunakan untuk mengukur kualitas dalam pengklasifikasian program. Program yang telah dibuat mengklasifikasikan citra menjadi tiga bagian, sehingga *confusion matrix* yang akan didefinisikan berukuran  $3 \times 3$ . Jadi, total banyaknya variabel yang akan didefinisikan pada *confusion matrix* adalah 9.

		Nilai Prediksi ( Segmentasi Program )		
		<i>Back</i>	<i>Healthy</i>	<i>Sick</i>
Nilai Sebenarnya ( Segmentasi Manual )	<i>Back</i>	<i>True Back</i>	<i>False Healthy I</i>	<i>False Sick I</i>
	<i>Healthy</i>	<i>False Back I</i>	<i>True Healthy</i>	<i>False Sick II</i>
	<i>Sick</i>	<i>False Back II</i>	<i>False Healthy II</i>	<i>True Sick</i>

Gambar 4. *Confusion matrix*

*True background* (TB), *true healthy* (TH), dan *true sick* (TS) berturut-turut menyatakan banyaknya piksel yang terdeteksi *background*, bagian sehat, dan bagian berpenyakit pada citra hasil program dan citra segmentasi manual pada titik  $(x, y)$  yang sama. Sedangkan pada variabel yang berawalan *false* merupakan banyaknya piksel yang salah terdeteksi oleh program.

### 2.4.2 Tingkat Akurasi

Tingkat akurasi (TA) diformulasikan sebagai banyaknya piksel yang benar dideteksi oleh program, kemudian dibagi dengan total keseluruhan banyaknya piksel. Tujuan dari tingkat akurasi adalah untuk mengukur seberapa akurat program mengklasifikasikan dengan benar. Formulasi dari tingkat akurasi dituliskan sebagai berikut

$$TA = \frac{TB+TS+TH}{TB+TS+TH+FB1+FB2+FS1+FS2+FH1+FH2}$$



### 2.4.3 Tingkat Presisi

Tingkat presisi masing-masing bagian (*background*, sehat, dan sakit) diformulasikan dengan banyaknya piksel bagian tersebut yang benar dideteksi oleh program, lalu dibagi dengan banyaknya piksel yang diprediksi program ke dalam bagian tersebut. Tingkat presisi (TP) diformulasikan sebagai rata-rata tingkat presisi pada masing-masing bagian. Tingkat presisi menggambarkan akurasi antara data yang diminta dengan hasil prediksi yang diberikan oleh program. *Formulasi* dari tingkat presisi dituliskan sebagai berikut

$$TP = \frac{1}{3} \left( \frac{TB}{TB+FB1+FB2} + \frac{TH}{TH+FH1+FH2} + \frac{TS}{TS+FS1+FS2} \right)$$

### 2.4.4 Recall atau Sensitivitas

Tingkat *recall* atau sensitivitas masing-masing bagian (*background*, sehat, dan sakit) diformulasikan dengan banyaknya piksel bagian tersebut yang benar dideteksi oleh program, lalu dibagi dengan banyaknya piksel yang merupakan nilai sebenarnya dalam bagian tersebut. Sensitivitas (S) diformulasikan sebagai rata-rata sensitivitas pada masing-masing bagian. Sensitivitas menggambarkan keberhasilan program dalam menemukan kembali informasi. *Formulasi* dari tingkat sensitivitas dituliskan sebagai berikut

$$S = \frac{1}{3} \left( \frac{TB}{TB+FH1+FS1} + \frac{TH}{TH+FB1+FS2} + \frac{TS}{TS+FB2+FH2} \right)$$

### 2.4.5 Nilai F1 Score

Nilai *f1 score* (F1) diformulasikan sebagai dua kali sensitivitas kali tingkat presisi kemudian dibagi dengan jumlah sensitivitas dengan tingkat presisi. Nilai *f1 score* menggambarkan perbandingan rata-rata antara nilai presisi dan sensitivitas yang dibobotkan. *Formulasi* dari tingkat presisi dijabarkan sebagai berikut :

$$F1 = \frac{2 \times S \times TP}{S + TP}$$

## 3. Hasil dan Pembahasan

### 3.1 Hasil Contrast Stretching Citra

Pada tahapan ini, citra yang telah dipanggil dari *dataset* akan diatur persebaran gelap terang intensitas warnanya. Teknik *contrast stretching* berguna untuk memperbaiki citra yang terlalu gelap atau terlalu terang. Keenam *dataset* yang

digunakan memiliki warna yang sedikit gelap sehingga harus dilakukan proses *contrast stretching*.



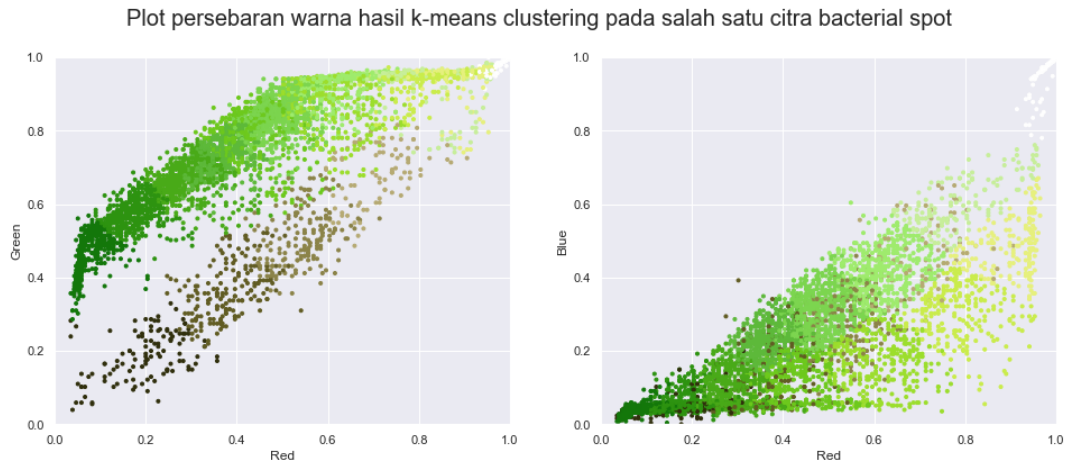
Gambar 5. Citra *bacterial spot* original (kiri) dan hasil *contrast stretching* (kanan)

Gambar sebelah kiri merupakan salah satu citra *original* yang diambil dari *dataset* yang memiliki warna yang sedikit gelap. Sedangkan pada gambar sebelah kanan, merupakan hasil citra yang telah dilakukan proses *contrast stretching* yang memiliki kualitas lebih baik.

### 3.2 Hasil Visualisasi Persebaran Warna Hasil *K-means Clustering*

Pada tahapan ini, persebaran warna hasil *k-means clustering* akan divisualisasikan. Karena visualisasi telah dilakukan, maka proses pengamatan persebaran warna pada hasil *clustering* citra dapat dilakukan. Proses ini bertujuan untuk mengidentifikasi penyebaran warna penyakit pada daun berdasarkan hasil visualisasi melalui *scatter plot*.

*K-means clustering* akan melakukan pengelompokan piksel warna. Nilai piksel warna yang mirip atau memiliki lokasi yang berdekatan akan diubah menjadi warna yang sama. Banyaknya warna pada hasil *clustering* bergantung kepada banyaknya jumlah *cluster* yang ditentukan. Jumlah cluster terbaik dalam proses segmentasi ini adalah sebanyak 16 yang didapatkan melalui metode *trial and error*. Akibatnya, total variasi warna pada *scatter plot* ada sebanyak 16 seperti yang terlihat pada gambar di bawah.

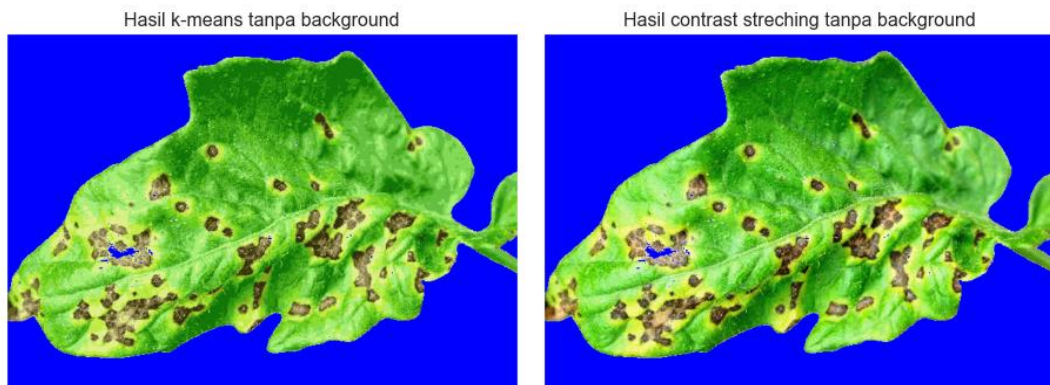


Gambar 6. Plot persebaran warna hasil *k-means clustering*

Pada gambar sebelah kiri adalah plot persebaran warna dengan sumbu-x menyatakan nilai intensitas ruang warna merah dan sumbu-y menyatakan nilai intensitas ruang warna hijau. Sedangkan pada gambar sebelah kanan adalah plot persebaran warna dengan sumbu sumbu-x menyatakan nilai intensitas ruang warna merah dan sumbu-y menyatakan nilai intensitas ruang warna biru.

### 3.3 Hasil *K-Means* dan *Contrast Stretching* Tanpa *Background*

Pada tahapan ini penghapusan *background* pada citra akan dilakukan. Proses penghapusan *background* ini bertujuan untuk memisahkan bagian daun dengan *background*. Meskipun *dataset* yang digunakan dalam penelitian ini telah memiliki warna yang polos, tetapi proses penghapusan *background* ini tetap harus dilakukan agar semua citra yang diolah memiliki *background* yang sama, yaitu berwarna biru seperti pada gambar di bawah.

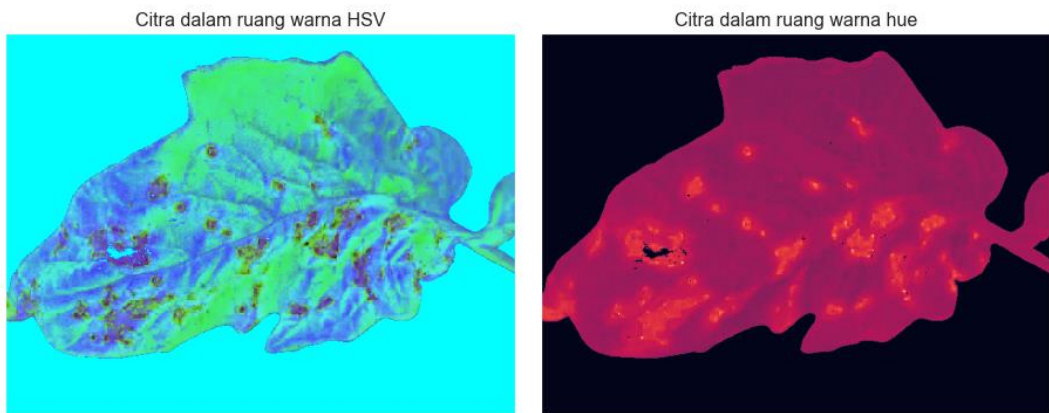


Gambar 7. Hasil *k-means* (kiri) dan *contrast stretching* (kanan) tanpa *background*

Gambar di sebelah kiri merupakan citra hasil segmentasi dengan *k-means* yang telah dihapus *backgroundnya*. Sedangkan pada gambar di sebelah kanan merupakan citra hasil *contrast stretching* yang telah dihapus *backgroundnya*.

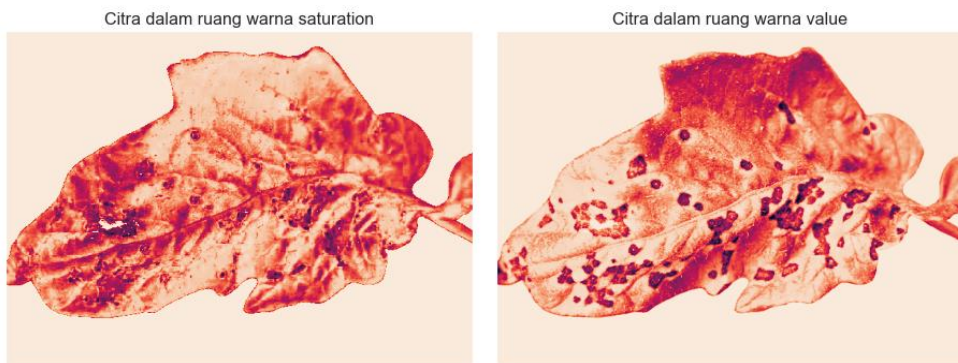
### 3.4 Hasil Citra dalam Ruang Warna HSV

Pada tahapan ini citra diubah menjadi ruang warna HSV. Setiap komponen ruang warna HSV pada citra diamati dengan melihat perbedaan daerah sakit dan daerah sehat yang paling terlihat. Setelah dilakukan pengamatan, komponen *hue* merupakan komponen terbaik yang memperlihatkan perbedaan daerah *background*, daerah sakit, dan daerah sehat daun yang terbaik.



Gambar 8. Citra dalam ruang warna HSV (kiri) dan dalam komponen *hue* (kanan)

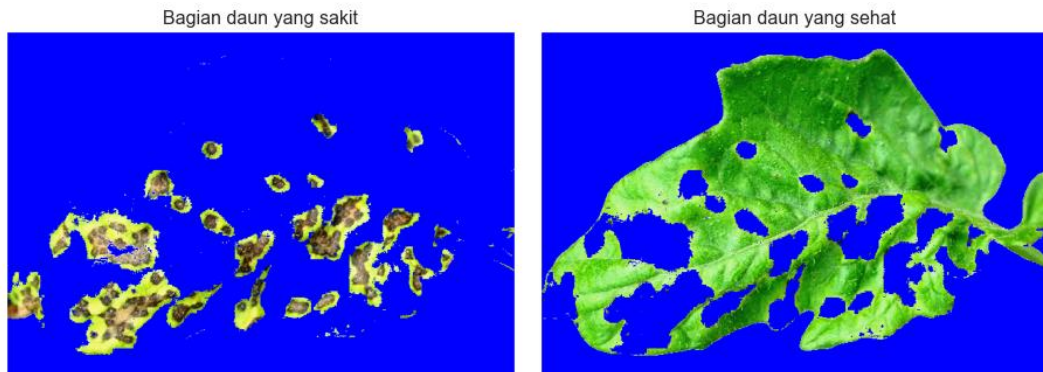
Gambar di sebelah kiri atas merupakan citra hasil *contrast stretching* yang pada awalnya dalam ruang warna BGR yang diubah menjadi HSV, lalu pada kanan atas adalah citra dalam komponen *hue*. Selanjutnya gambar pada kiri bawah adalah citra dalam komponen *saturation*, sedangkan pada kanan bawah adalah citra dalam komponen *value*.



Gambar 9. Citra dalam ruang warna *saturation* (kiri) dan dalam komponen *value* (kanan)

### 3.5 Hasil Pemisahan Bagian Berpenyakit dengan Bagian Sehat

Pada tahapan ini citra hasil *contrast stretching* tanpa *background* akan dipisahkan menjadi dua bagian yaitu bagian daun yang berpenyakit dan bagian yang sehat. Pemisahan tersebut dilakukan dengan bantuan pemilihan rentang nilai *hue* pada citra. Salah satu contoh hasil pemisahan ini adalah seperti pada gambar di bawah.



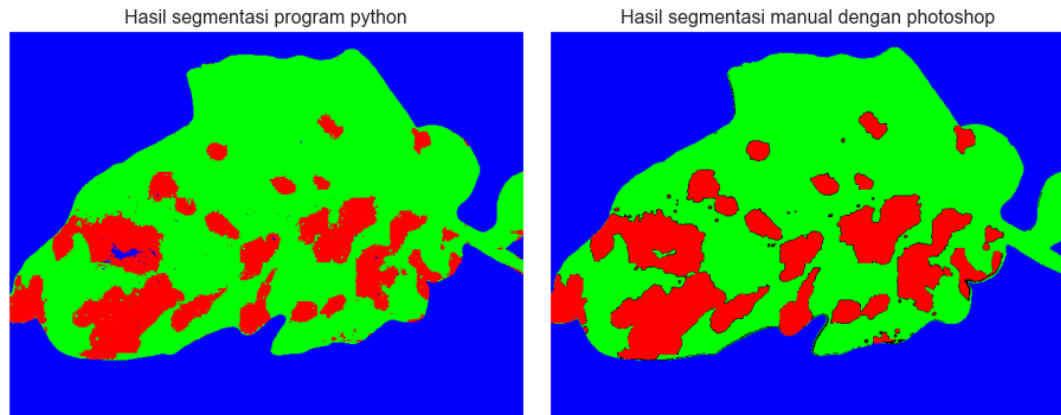
Gambar 10. Bagian daun yang sakit (kiri) dan yang sehat (kanan)

Gambar di sebelah kiri adalah citra hasil *contrast stretching* tanpa *background* yang berpenyakit, sedangkan gambar di sebelah kanan adalah bagian yang sehat. Pada kedua gambar di atas, terlihat bahwa pemilihan rentang nilai *hue* yang tepat dapat memisahkan bagian daun sakit yang menguning atau kecoklatan serta bagian daun sehat yang berwarna hijau.

### 3.6 Hasil *Thresholding* Secara Manual dan Melalui Program

Pada tahapan ini citra yang telah dibagi menjadi tiga bagian akan dilakukan *thresholding*. Tujuan dari tahapan ini adalah agar citra hanya memiliki tiga jenis warna, yaitu warna biru yang merepresentasikan bagian *background*, warna hijau yang merepresentasikan bagian daun sehat, dan warna merah yang merepresentasikan bagian daun berpenyakit seperti pada gambar di bawah.





Gambar 11. Hasil segmentasi program python (kiri) dan manual dengan Photoshop (kanan)

Gambar di sebelah kiri adalah hasil segmentasi yang dihasilkan oleh program, sedangkan gambar di sebelah kanan adalah hasil segmentasi manual dengan Photoshop. Hasil segmentasi melalui Photoshop diasumsikan merupakan citra yang tersegmentasi menjadi bagian *background*, bagian sehat dan bagian berpenyakit yang sebenarnya atau memiliki keakuratan 100%. Sehingga kedua gambar di atas dapat dibandingkan untuk mencari tingkat akurasi dari program yang telah dibuat.

### 3.7 Hasil Tingkat Akurasi, Presisi, *Recall*, dan *F1 Score* Program

Pada tahapan ini akan dilakukan perhitungan tingkat akurasi, presisi, sensitivitas, dan *f1 score*. Tabel di bawah menunjukkan tingkat akurasi dan tingkat presisi terhadap keenam citra yang digunakan sebagai *dataset*.

**Tabel 1.** Tingkat akurasi dan presisi program terhadap setiap citra

Jenis Penyakit	Nama File	Akurasi	Presisi	<i>Recall</i>	<i>F1 Score</i>
Septoria Spot	tomato-septoria-leaf-spot.jpg	92.7784%	88.0643%	86.0610%	87.0511%
	tomato-septoria-leaf-spot2.jpg	97.1968%	90.2937%	83.8705%	86.9636%
Mosaic Virus	tomato-mosaic-virus.jpg	92.4909%	89.8637%	78.4509%	83.7703%
	tomato-mosaic-virus2.jpg	93.1646%	88.2257%	86.6007%	87.4057%
Bacterial Spot	bacterial-leaf-spot.jpg	92.2413%	78.0173%	81.8757%	79.9000%
	bacterial-leaf-spot2.jpg	96.8934%	97.0539%	94.1665%	95.5884%

Pada tabel di atas dapat dilihat bahwa tingkat akurasi pada model dalam membedakan daun berpenyakit sudah baik dengan semua akurasi citra berada di atas 90%. Sedangkan pada tingkat presisi, *f1 score* dan sensitivitas atau *recall* juga

menunjukkan hasil yang cukup baik dengan nilai diatas 75%. Hal ini menginterpretasikan bahwa program dapat mengklasifikasikan bagian daun yang berpenyakit dengan baik.

#### 4. Kesimpulan

Metode *k-means* yang dikombinasikan dengan nilai modus pada *list* yang berisi piksel tepian citra dapat digunakan untuk memisahkan bagian daun pada citra dengan backgroundnya. Komponen warna *hue* pada citra hasil *contrast stretching* pada ruang warna HSV dapat digunakan untuk melakukan segmentasi terhadap daerah sehat dan daerah sakit pada citra. Nilai piksel *hue* pada rentang 80 hingga 175 yang disertai dengan metode *k-means* dapat mendeteksi ketiga bagian citra dengan tingkat akurasi yang tinggi. Tingkat akurasi yang dihasilkan program semuanya berada di atas 90%. Sedangkan pada tingkat presisi, sensitivitas atau *f1 score* juga menunjukkan hasil yang cukup baik dengan nilai diatas 75%. Hal ini menunjukkan bahwa metode *k-means* dan pemilihan rentang *hue* yang tepat pada program pengolahan citra dapat mengklasifikasikan ketiga bagian daun tomat dengan baik.

#### 5. Saran

Citra yang digunakan dalam penelitian ini masih diasumsikan memiliki *background* polos, seluruh bagian daun termuat dalam citra, dan merupakan daun tomat dewasa. Kemudian pada proses evaluasi model, hasil segmentasi program akan dibandingkan dengan hasil segmentasi manual melalui photoshop yang diasumsikan memiliki keakuratan 100%. Artikel ini diharapkan dapat dikembangkan, agar proses pendeteksian penyakit pada daun tomat tidak memerlukan asumsi yang terlalu banyak. Selain itu pada perkembangan selanjutnya, proses segmentasi secara manual sebaiknya dilakukan oleh ahli agar program pengolahan citra ini dapat memberikan hasil akurasi yang lebih tinggi dan terpercaya.

#### Daftar Pustaka

- [1] "Gardening Know How," 2020. [Online]. Available: <https://www.gardeningknowhow.com/>. [Diakses 15 Desember 2022].

- [2] F.-X. Branthôme, "Consumption: 2021 in the wake of 2020," 25 Maret 2022. [Online]. Available: [https://www.tomatonews.com/en/consumption-2021-in-the-wake-of-2020\\_2\\_1618.html](https://www.tomatonews.com/en/consumption-2021-in-the-wake-of-2020_2_1618.html). [Diakses 15 Desember 2022].
- [3] K. A. Hudaya, B. Sunarko and A. Nugroho, "Kinerja Algoritma Pengelompokan *Fuzzy C-Means* pada Segmentasi Citra Leukosit," *Jurnal Nasional Teknik Elektro dan Teknologi Informasi*, vol. 11, no. 1, pp. 41-46, 2022.
- [4] K. Usman, N. K. C. Pratiwi, N. Ibrahim and V. P. Rahadi, " Evaluasi Optimizer pada Residual Network untuk Klasifikasi Klon Teh Seri GMB Berbasis Citra Daun," *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika* , vol. 9, no. 4, pp. 841-855, 2021.
- [5] M. Kluepfel, J. H. Blake dan A. P. Keinath, "Tomato Diseases & Disorders," 26 Mei 2021. [Online]. Available: <https://hgic.clemson.edu/factsheet/tomato-diseases-disorders/>. [Diakses 15 Desember 2022].
- [6] Meiriyama, S. Devella and S. M. Adelfi, "Klasifikasi Daun Herbal Berdasarkan Fitur Bentuk Dan Tekstur Menggunakan KNN," *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 9, no. 3, pp. 2573-2584, 2022.
- [7] R. A. Saputra, Suharyanto, S. Wasiyanti, D. F. Saefudin, A. Supriyatna, and A. Wibowo, "Rice Leaf Disease Image Classifications Using KNN Based on GLCM Feature Extraction," *J. Phys. Conf. Ser.*, Vol. 1641, No. 1, 2020, doi: 10.1088/1742-6596/1641/1/012080.
- [8] Trivusi, "K-means Clustering" 2020. [Online]. Available : <https://www.trivusi.web.id/2022/06/algoritma-kmeans-clustering.html/>. [Diakses 15 Desember 2022].