

# Klasifikasi Penyakit Daun Tomat Menggunakan Pengolahan Citra Dan Algoritma Machine Learning

Romi Antoni<sup>1</sup>, Susiana Khosasih<sup>2</sup>, Ricky Irnanda<sup>3</sup>, Iswanto<sup>4</sup>, Farhan Sardy Abdillah<sup>5</sup>, Yiska Dayanti Zagoto<sup>6</sup>, Rika Rosnelly<sup>7</sup>

Email: <sup>1</sup>romi04antoni@gmail.com, <sup>2</sup>susianakhosasih21@gmail.com, <sup>3</sup>rickyirmandal7@gmail.com, <sup>4</sup>iswantoo1982@gmail.com, <sup>5</sup>fsabdillah19@gmail.com, <sup>6</sup>Zagotoyiska@gmail.com, <sup>7</sup>rikarosnelly@gmail.com  
(\* Email Corresponding Author: susianakhosasih21@gmail.com)

Received: January 7, 2026. | Revision: January 10, 2026 | Accepted: January 10, 2026

## Abstrak

Klasifikasi penyakit daun tomat merupakan langkah penting untuk meningkatkan produktivitas pertanian dan meminimalkan kerugian akibat patogen. Penelitian ini bertujuan membandingkan dan mengevaluasi performa algoritma Naive Bayes dan Support Vector Machine (SVM) dalam klasifikasi penyakit daun tomat berbasis pengolahan citra digital. Pipeline penelitian mencakup segmentasi citra berbasis HSV, ekstraksi fitur warna, bentuk, dan tekstur menggunakan metode Gray Level Co-occurrence Matrix (GLCM) dan Local Binary Pattern (LBP), serta proses klasifikasi. Sistem diimplementasikan dalam bentuk Graphical User Interface (GUI) berbasis MATLAB untuk memudahkan manajemen data latih, pelatihan model, klasifikasi, dan evaluasi performa. Hasil pengujian menunjukkan bahwa SVM mencapai akurasi 92,36%, lebih tinggi dibandingkan Naive Bayes sebesar 79,41%. Kontribusi penelitian ini meliputi analisis komparatif Naive Bayes dan SVM dalam klasifikasi penyakit daun tomat, integrasi fitur warna, bentuk, dan tekstur dalam satu pipeline, dan pengembangan GUI interaktif untuk klasifikasi. Penelitian ini diharapkan dapat mendukung pertanian presisi melalui deteksi penyakit daun tomat yang lebih cepat, akurat, dan efisien.

**Kata Kunci:** Klasifikasi Penyakit, Daun Tomat, Naive Bayes, Support Vector Machine, Machine Learning

## Abstract

*Classification of tomato leaf diseases is an important step to increase agricultural productivity and minimize losses due to pathogens. This study aims to compare and evaluate the performance of the Naive Bayes and Support Vector Machine (SVM) algorithms in tomato leaf disease classification based on digital image processing. The research pipeline includes HSV-based image segmentation, extraction of color, shape, and texture features using the Gray Level Co-occurrence Matrix (GLCM) and Local Binary Pattern (LBP) methods, and the classification process. The system is implemented in a MATLAB-based Graphical User Interface (GUI) to facilitate training data management, model training, classification, and performance evaluation. The test results show that SVM achieves an accuracy of 92.36%, higher than Naive Bayes at 79.41%. The contributions of this study include a comparative analysis of Naive Bayes and SVM in tomato leaf disease classification, integration of color, shape, and texture features in one pipeline, and development of an interactive GUI for classification. This research is expected to support precision agriculture through faster, more accurate, and more efficient detection of tomato leaf diseases*

**Keywords:** Disease Classification, Tomato Leaves, Naive Bayes, Support Vector Machine, Machine Learning

## 1. PENDAHULUAN

Indonesia merupakan negara agraris dan mata pencaharian utama masyarakatnya adalah bertani. Letak geografis Indonesia yang berada di daerah tropis, sehingga kondisi iklim, tanah dan sumber daya lainnya di berbagai wilayah Indonesia memiliki potensi yang besar untuk pengembangan pertanian. Pemanfaatan sumber daya pertanian merupakan kunci untuk meningkatkan produktivitas pertanian sehingga sumber daya yang terbatas harus dialokasikan seefisien mungkin [1]. Produktivitas tanaman tomat (*Solanum lycopersicum*) sangat dipengaruhi oleh kesehatan daun yang menjadi indikator utama serangan penyakit. Daun merupakan organ vital yang berperan dalam proses *photosynthesis*, sehingga kerusakan akibat patogen dapat menurunkan kemampuan tanaman dalam menghasilkan energi dan nutrisi. Penyakit daun seperti *early blight*, *late blight*, *leaf mold*, dan *bacterial spot* telah lama dikenal sebagai ancaman serius bagi budidaya tomat karena dapat menurunkan hasil panen secara signifikan. Apabila tidak terdeteksi sejak dini, kerugian yang ditimbulkan mencakup penurunan hasil panen, degradasi kualitas buah, hingga kerugian ekonomi yang besar bagi petani. Oleh karena itu, upaya deteksi dini terhadap penyakit tanaman menjadi hal yang krusial dalam meningkatkan efisiensi pengendalian dan pencegahan penyakit tanaman [2]. Permasalahan utama yang dihadapi petani dalam mendeteksi penyakit daun tomat adalah keterbatasan metode tradisional yang masih bergantung pada pengamatan visual manual. Identifikasi konvensional membutuhkan pengalaman dan keahlian khusus, serta sering kali bersifat subjektif. Akibatnya, diagnosis menjadi tidak konsisten dan berpotensi menimbulkan kesalahan. Selain itu, keterlambatan dalam mengenali gejala penyakit dapat menyebabkan penyebaran patogen semakin luas dan sulit dikendalikan. Kondisi ini menimbulkan kebutuhan mendesak akan sistem klasifikasi otomatis berbasis citra digital yang

mampu mendeteksi penyakit daun tomat secara cepat, akurat, dan konsisten. Sebagai solusi, penelitian ini menawarkan pendekatan berbasis *machine learning* dengan algoritma *Naive Bayes* dan *Support Vector Machine (SVM)*. *Machine learning* memiliki beberapa model algoritma yang dikategorikan ke dalam *Supervised learning* dan *Unsupervised learning* [3], [4]. *Naive Bayes* dipilih karena kesederhanaannya dalam mengelola probabilitas antar fitur serta efisiensinya pada dataset berukuran sedang. Algoritma ini bekerja dengan prinsip *Bayes theorem* yang menghitung peluang suatu kelas berdasarkan distribusi fitur, sehingga cocok digunakan pada kasus klasifikasi dengan data yang memiliki ketidakpastian. Sementara itu, *SVM* dikenal memiliki kemampuan *margin-based classification* yang kuat dalam memisahkan kelas dengan akurasi tinggi. Dengan memanfaatkan fungsi *kernel*, *SVM* dapat mengatasi data yang tidak terpisahkan secara linear, sehingga lebih fleksibel dalam menangani variasi citra daun tomat.

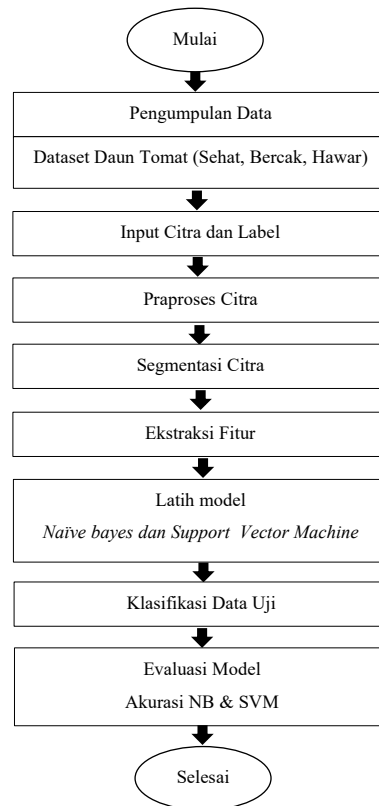
Berbagai penelitian sebelumnya telah menunjukkan efektivitas metode *machine learning* dalam klasifikasi penyakit daun tomat. Azli dkk. Menggunakan ekstraksi fitur HSV dan GLCM dengan *SVM* dan mencapai akurasi uji sebesar 93,75% [5]. Agusta dkk. Menegaskan bahwa kombinasi fitur warna dan tekstur (RGB + GLCM) memberikan hasil lebih baik dibandingkan fitur tunggal, dengan akurasi mencapai 90,6% menggunakan K-Nearest Neighbor (KNN) [6]. Zheng dan Du membandingkan beberapa arsitektur *Convolutional Neural Network (CNN)* pada dataset PlantVillage dan menemukan bahwa *MobileNet* mampu mencapai akurasi sekitar 91% dengan efisiensi komputasi tinggi [7]. Khultsum dan Subekti menunjukkan bahwa *Random Forest* dengan kombinasi fitur warna, Hu-Moments, dan Haralick mampu mencapai akurasi hingga 96% [8]. Namun, penelitian tersebut tidak memanfaatkan fitur bentuk sehingga sensitif terhadap noise. Das dkk. Melalui tinjauan literatur menegaskan bahwa pendekatan *deep learning* mendominasi klasifikasi penyakit tanaman, namun tahapan *preprocessing* dan segmentasi citra tetap menjadi faktor penentu akurasi [9]. Widyastuti dkk. Mengklasifikasikan varietas tomat Red Zebra, Green Zebra, dan Kumato menggunakan ekstraksi fitur RGB dan algoritma *Naive Bayes*, dengan akurasi tertinggi 85,71% pada kombinasi atribut RG [10]. Rizal dkk. membandingkan algoritme *SVM* dan *CNN* pada citra daun tomat, kentang, dan jagung. Astiningrum dkk. Mengidentifikasi penyakit daun tomat dengan fitur warna (RGB) dan tekstur (GLCM) menggunakan KNN, menghasilkan akurasi 92,89% [11]. Sementara itu, Attallah. Mengusulkan pendekatan berbasis *CNN* ringan dengan transfer learning dan feature selection, yang mampu mencapai akurasi 99,9% pada dataset PlantVillage [12]. Walaupun hasilnya sangat tinggi, metode ini membutuhkan dataset besar dan komputasi tinggi, sehingga kurang efisien untuk implementasi praktis di lapangan. Nyasulu dkk. Melakukan studi komparatif klasifikasi penyakit jamur daun tomat berbasis fitur tekstur GLCM dengan berbagai algoritma (ANN, KNN, RF, SVM), dan melaporkan akurasi rata-rata 94% [13]. Meskipun penelitian-penelitian tersebut telah menghasilkan akurasi tinggi, terdapat GAP Analysis yang perlu diperhatikan. Pertama, sebagian besar penelitian berfokus pada algoritma *deep learning* yang membutuhkan dataset besar dan komputasi tinggi, sehingga kurang efisien untuk implementasi di lapangan. Kedua, penelitian berbasis algoritma klasik seperti KNN dan *Random Forest* masih terbatas dalam mengeksplorasi variasi algoritma yang mampu mengelola ketidakpastian seperti *Naive Bayes*, padahal algoritma ini memiliki potensi dalam mengelola ketidakpastian data. Ketiga, meskipun *SVM* terbukti efektif, integrasi sistematis antara *SVM* dan *Naive Bayes* dengan fitur warna, bentuk, dan tekstur belum banyak diteliti. Oleh karena itu, penelitian ini berupaya mengisi celah tersebut dengan mengembangkan sistem klasifikasi penyakit daun tomat berbasis *Naive Bayes* dan *SVM* yang mengintegrasikan fitur warna, bentuk, dan tekstur dalam satu *pipeline*.

Penelitian ini bertujuan untuk membandingkan dan mengevaluasi performa algoritma *Naive Bayes* dan *SVM*, sehingga dapat diketahui keunggulan dan keterbatasan masing-masing algoritma. Sistem diimplementasikan dalam bentuk *Graphical User Interface (GUI)* berbasis MATLAB untuk memudahkan manajemen data latih, pelatihan model, klasifikasi, dan evaluasi performa secara interaktif. Harapannya, penelitian ini dapat memberikan kontribusi nyata dalam mendukung pertanian presisi melalui deteksi penyakit tomat yang lebih cepat, akurat, dan efisien, sekaligus memperkaya literatur akademik di bidang *computer vision* dan *machine learning* untuk aplikasi pertanian..

## 2. METODOLOGI PENELITIAN

### 2.1 Tahapan Penelitian

Penelitian ini dilakukan dengan pendekatan *processing pipeline* berbasis GUI MATLAB menggunakan algoritma *Naive Bayes* dan *SVM*. Seluruh tahapan penelitian dilakukan otomatis, sehingga tidak diperlukan pemrosesan manual per citra. Flowchart penelitian dapat dilihat pada Gambar 1.



**Gambar 1.** Tahapan Penelitian

## 2.2 Pengumpulan Data

Dataset yang digunakan dalam penelitian ini adalah data sekunder berupa citra daun tomat yang di ambil dari Kaggle ([www.kaggle.com/datasets/emmarex/plantdisease](http://www.kaggle.com/datasets/emmarex/plantdisease)). Jumlah citra yang digunakan sebanyak 4.055 gambar daun tomat dengan tiga kategori kelas, yaitu sehat, bercak, dan hawar. Seluruh citra kemudian dibagi menjadi dua kelompok, yaitu data latih dan data uji, dengan proporsi 80% untuk pelatihan dan 20% untuk pengujian. Data latih terdiri dari 3.244 citra daun tomat yang masing-masing diberi nama file sesuai kelasnya, yaitu sehat, bercak, dan hawar. Data ini disimpan dalam folder *DataLatih* dan digunakan untuk proses ekstraksi fitur serta pelatihan model klasifikasi. Sementara itu, data uji terdiri dari 811 citra daun tomat dengan penamaan file yang sama, disimpan dalam folder *CitraUji*, dan digunakan untuk proses klasifikasi serta evaluasi performa model.

## 2.3 Processing Pipeline

### a. Input Citra dan Pemberian Label

Tahap awal dimulai dengan pemilihan citra daun tomat melalui GUI. Setelah citra dipilih, pengguna memasukkan label kelas secara manual melalui komponen input teks. Label ini melekat pada seluruh citra yang diproses. Pada tahap klasifikasi, sistem juga mengekstrak label asli dari nama file citra (substring sebelum underscore atau ekstensi) untuk keperluan evaluasi dan menyimpannya dalam kolom LabelAsli.

### b. Segmentasi

Setiap citra yang dipilih dibaca menggunakan `imread`. Proses segmentasi dilakukan dengan fungsi `segmentasi(img)` yang menghasilkan:

1. Mask : citra biner yang memisahkan objek dari latar belakang.
2. SegInfo : informasi tambahan hasil segmentasi (misalnya area objek).

Segmentasi ini bertujuan agar fitur hanya dihitung dari area objek yang relevan. Hasil segmentasi dapat ditampilkan di GUI untuk memberikan umpan balik visual kepada pengguna.

### c. Ekstraksi Fitur

Setelah segmentasi, sistem mengekstraksi fitur menggunakan fungsi `extractFeatures(img, mask, segInfo)`. Fitur yang diekstrak adalah:

1. Fitur warna berupa rata-rata Hue, Saturation, dan Value.
2. Fitur bentuk berupa Area dan Eccentricity.
3. Fitur tekstur berupa Contrast, Correlation, Energy, dan Homogeneity dari GLCM.

Selain itu, pola tekstur lokal dihitung menggunakan Local Binary Pattern (LBP) dan dirata-ratakan. Seluruh fitur digabungkan menjadi satu vektor fitur, dan disimpan dalam dataset dataLatih.mat.  $F=[Hue, Saturation, Value, Area, Eccentricity, Contrast, Correlation, Energy, Homogeneity, LBP]$ . Dataset berlabel ini digunakan untuk melatih dua algoritma klasifikasi.

d. Klasifikasi

Proses klasifikasi berjalan dengan cara menguji data uji baru menggunakan model yang sudah dilatih sebelumnya. Setiap gambar yang dipilih pengguna pertama-tama diproses melalui segmentasi untuk memisahkan objek daun dari latar belakang, lalu dilakukan ekstraksi fitur sehingga citra berubah menjadi vektor numerik yang mewakili ciri warna, bentuk, dan tekstur. Vektor fitur ini kemudian dimasukkan ke dalam dua model klasifikasi, yaitu Naïve Bayes dan SVM.

1. *Naïve Bayes* (NB)

Naive Bayes adalah algoritma klasifikasi yang menggunakan teorema Bayes dengan asumsi bahwa semua variabel penjelas adalah independen satu sama lain. Pendekatan ini dapat dianggap sebagai ada atau tidaknya kejadian pada kelompok tertentu yang tidak bergantung pada lingkungan atau tidak adanya kejadian lainnya [14]. Naive Bayes bekerja berdasarkan Teorema Bayes dengan asumsi independensi antar fitur. Berikut rumus teorema bayes :

$$\frac{P(C | X) \cdot P(C)}{P(X)} \quad (a)$$

dimana :

$X$  : data dengan kelas yang belum diketahui

$C$  : hipotesis kelas tertentu

$P(C | X)$  : probabilitas posterior

$P(X | C)$  : probabilitas likelihood

$P(C)$  : probabilitas prior

$P(X)$  : probabilitas evidence

Asumsi independensi fitur :

Jika fitur  $X = (x_1, x_2, \dots, x_n)$ , maka:

$$P(X | C) = \prod_{i=1}^n P(x_i | C) \quad (b)$$

Kelas dipilih berdasarkan probabilitas maksimum:

$$\hat{C} = \arg \max_C P(C) \cdot \prod_{i=1}^n P(x_i | C) \quad (c)$$

2. *Support Vector Machine* (SVM)

Support Vector Machine (SVM) adalah pengklasifikasi diskriminatif yang menghasilkan hyperplane pemisah. Toleransi kesalahan disertakan untuk membuat hyperplane pemisah menjadi kuat jika terjadi data kelas yang tidak dapat dipisahkan [15]. SVM bekerja dengan mencari hyperplane terbaik yang memisahkan kelas. Berikut fungsi keputusan linear :

$$Lf(x) = \text{sign}(w \cdot x + b) \quad (d)$$

Dimana :

$w$  = vektor bobot.

$x$  = vektor fitur.

$b$  = bias.

sign menentukan apakah data berada di sisi positif atau negatif hyperplane.

Jika menggunakan kernel Radial Basis Function (RBF). Fungsi kernel RBF dituliskan sebagai:

$$K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2) \quad (e)$$

Sedangkan fungsi keputusan SVM adalah :

$$f(x) = \text{sign}\left(\sum_{i=1}^n \alpha_i y_i K(x_i, x) + b\right) \quad (f)$$

Dimana :

- $x_i, x_j$  : data training  
 $K(x_i, x)$  : fungsi kernel RBF antara data training  $x_i$  dan data uji  $x$   
 $\gamma$  : parameter kernel RBF yang mengontrol pengaruh jarak antar data  
 $f(x)$  : fungsi keputusan, hasilnya berupa tanda (+1 atau -1) untuk menentukan kelas  
 $\alpha_i$  : bobot hasil training  
 $y_i$  : label kelas  
 $b$  : bias

Model hasil pelatihan disimpan dalam modelNB.mat dan modelSVM.mat. NB menghitung probabilitas tiap kelas dengan Teorema Bayes, lalu memilih kelas dengan probabilitas tertinggi. Sedangkan SVM menentukan kelas berdasarkan posisi data terhadap hyperplane, dengan bantuan kernel jika data tidak linear.

e. Evaluasi

Evaluasi adalah tahap setelah klasifikasi yang bertujuan untuk mengukur seberapa baik model bekerja dengan data uji. Dari evaluasi ini, kita bisa tahu apakah NB lebih akurat dibanding SVM, kelas mana yang paling sulit dikenali, dan apakah sistem sudah cukup baik atau perlu perbaikan. Evaluasi dilakukan dengan *confusion matrix*, *akurasi*, *precision*. Rumus evaluasi yang digunakan adalah sebagai berikut:

1. Accuracy : Seberapa banyak prediksi yang benar dari seluruh data uji.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (f)$$

2. Precision : ketepatan prediksi positif, yaitu seberapa banyak prediksi positif yang benar dibandingkan semua prediksi positif.

$$Precision = \frac{TP}{TP+FP} \quad (g)$$

3. Recall (Sensitivity) : kemampuan model menemukan semua data positif yang benar.

$$Recall = \frac{TP}{TP+FN} \quad (h)$$

dimana :

*Accuracy* : menunjukkan proporsi prediksi yang benar dibandingkan seluruh data

*Precision* : ketepatan prediksi positif

*Recall* : cakupan prediksi positif

*TP (True Positive)* : jumlah data positif yang diprediksi benar

*TN (True Negative)* : jumlah data negatif yang diprediksi benar

*FP (False Positive)* : jumlah data negatif yang salah diprediksi positif

*FN (False Negative)* : jumlah data positif yang salah diprediksi negative

4. Confusion Matrix : tabel yang menunjukkan distribusi prediksi benar dan salah untuk setiap kelas, sehingga kita bisa melihat kelas mana yang paling sering salah dikenali.

Dari evaluasi ini, kita bisa tahu apakah NB lebih akurat dibanding SVM, kelas mana yang paling sulit dikenali, dan apakah sistem sudah cukup baik atau perlu perbaikan.

### 3. HASIL DAN PEMBAHASAN

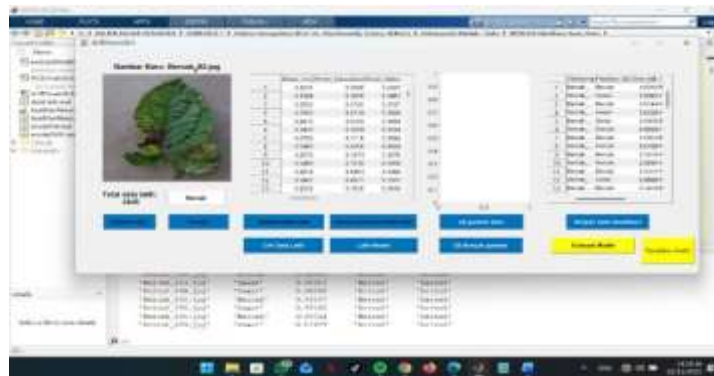
#### 3.1 Input Citra Dan Label

Proses input citra dilakukan bertahap pada setiap kelas dan diberikan nama label sesuai jenis penyakit citra daun yang di input. Input citra dapat dilakukan dalam jumlah besar tanpa harus dilakukan secara manual. Kemudian dilanjutkan processing citra. Dapat dilihat pada gambar berikut :

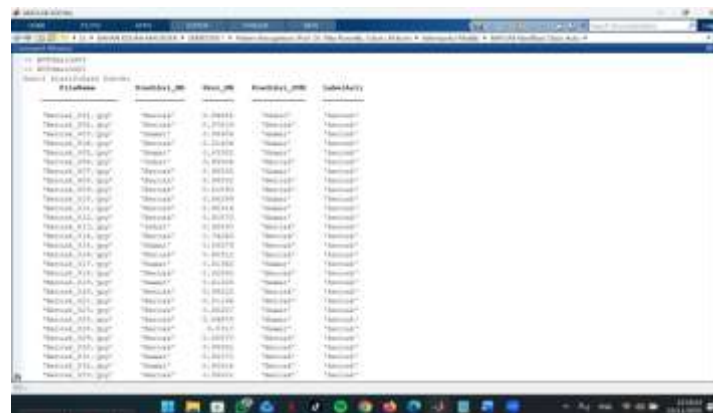


### 3.3 Klasifikasi

Proses klasifikasi dilakukan dengan dua algoritma, yaitu Naïve Bayes dan Support Vector Machine (SVM) dengan kernel RBF. Hasil uji klasifikasi yang diperoleh sebelum tahap evaluasi model berupa keluaran prediksi label untuk setiap citra uji. Pada tahap ini, sistem menampilkan daftar citra daun tomat beserta label asli dan label hasil prediksi yang dihasilkan oleh algoritma klasifikasi, baik Naïve Bayes maupun SVM. Data tersebut ditampilkan secara langsung pada *Command Window* dan juga disimpan dalam file CSV sebagai dokumentasi, sehingga memudahkan verifikasi awal terhadap kesesuaian prediksi dengan label sebenarnya. Informasi ini menjadi dasar untuk membentuk confusion matrix, karena setiap pasangan label asli dan label prediksi akan dihitung distribusinya. Dengan demikian, hasil uji klasifikasi sebelum evaluasi model berfungsi sebagai *output* mentah yang menunjukkan performa awal algoritma dalam mengenali kelas bercak, hawar, dan sehat, sebelum dilakukan pengukuran akurasi, precision, recall, dan F1-score secara lebih komprehensif. Dapat dilihat pada gambar berikut:



Gambar 5. Proses klasifikasi



Gambar 6. Hasil klasifikasi

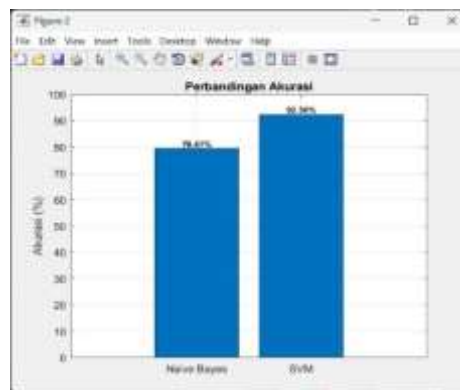
### 3.4 Evaluasi Model

Evaluasi model dilakukan menggunakan data uji sebesar 20% dari total dataset (*hold-out validation*) untuk mengukur performa model pada subset data yang tidak digunakan dalam proses pelatihan. Pendekatan ini bertujuan untuk mengetahui kemampuan generalisasi algoritma terhadap data baru yang belum pernah dilihat sebelumnya. Metrik yang digunakan dalam pengujian meliputi akurasi, precision, recall, dan F1-score, sehingga hasil evaluasi dapat memberikan gambaran komprehensif mengenai ketepatan, sensitivitas, dan keseimbangan prediksi model. Hasil evaluasi menggunakan data uji dapat dilihat pada gambar berikut berikut.

Gambar 7. Hasil Evaluasi Model

### 3.5 Perbandingan Evaluasi Model

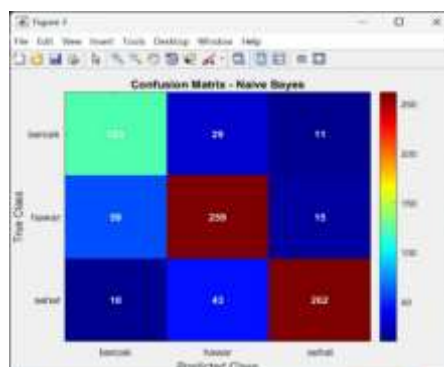
Menggunakan data uji sebesar 20% dari total dataset (*hold-out validation*) untuk mengukur performa model pada subset data yang tidak digunakan dalam pelatihan *Naive Bayes* menghasilkan akurasi 79.41%, sementara SVM mencapai 92.36%. *Naive Bayes* cukup baik untuk kelas sehat, tetapi sering keliru membedakan bercak dan hawar. SVM jauh lebih akurat dan konsisten, terutama dalam mengenali hawar dan sehat, dengan precision dan recall di atas 0.95. Kesimpulannya, SVM lebih unggul dan direkomendasikan untuk klasifikasi penyakit daun berbasis citra karena mampu menangani pola kompleks dengan lebih baik. Perbandingan dapat dilihat dengan jelas pada gambar grafik berikut :



Gambar 8. Hasil Perbandingan Model Naïve Bayes dan SVM

### 3.6 Confusion Matrix Model Naive Bayes

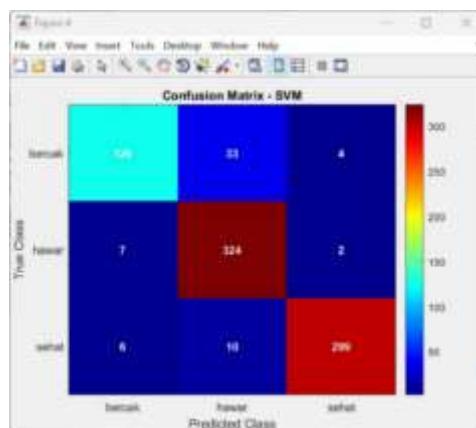
Hasil evaluasi menggunakan *confusion matrix* menunjukkan bahwa algoritma *Naive Bayes* mampu mengklasifikasikan 644 data dengan benar dari total 811 data uji, sehingga menghasilkan akurasi sebesar 79.41%. Pada kelas bercak, model berhasil mengenali 123 data dengan benar, namun masih terjadi kesalahan klasifikasi ke kelas hawar sebanyak 29 data dan ke kelas sehat sebanyak 11 data. Kelas hawar diklasifikasikan dengan benar sebanyak 259 data, sementara 59 data salah dikenali sebagai bercak dan 15 data sebagai sehat. Untuk kelas sehat, model berhasil mengklasifikasikan 262 data dengan benar, namun terdapat 10 data yang salah dikenali sebagai bercak dan 43 data sebagai hawar. Dapat dilihat pada gambar 9 :



Gambar 9. Hasil Confusion Matrix Model Naive Bayes

### 3.7 Confusion Matrix Model SVM

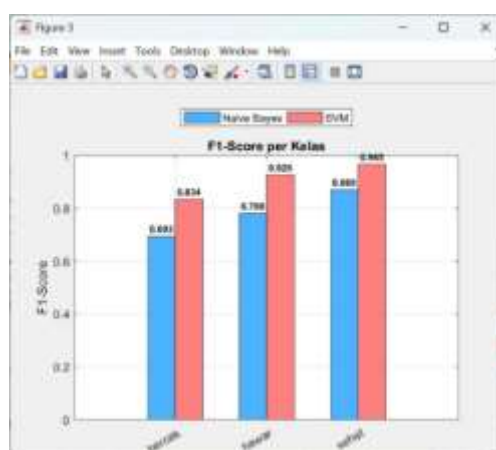
Hasil *confusion matrix* pada algoritma SVM menunjukkan performa yang lebih tinggi dengan 749 data diklasifikasikan benar dari total 811 data uji, menghasilkan akurasi sebesar 92.36%. Pada kelas bercak, SVM berhasil mengenali 126 data dengan benar, meskipun masih terdapat 33 data yang salah diklasifikasikan sebagai hawar dan 4 data sebagai sehat. Kelas hawar menunjukkan hasil yang sangat baik dengan 324 data diklasifikasikan benar, hanya 7 data salah dikenali sebagai bercak dan 2 data sebagai sehat. Sementara itu, kelas sehat berhasil diklasifikasikan dengan benar sebanyak 299 data, dengan kesalahan relatif kecil yaitu 6 data salah dikenali sebagai bercak dan 10 data sebagai hawar. Dapat dilihat pada gambar 10 :



Gambar 10. Hasil Confusion Matrix Model SVM

### 3.8 Grafik F1-Score Per-Kelas

Evaluasi dua model klasifikasi citra daun menunjukkan bahwa SVM jauh lebih unggul dibandingkan *Naive Bayes*. Untuk F1-score per kelas, SVM mencetak nilai lebih tinggi di semua kategori: Bercak: 0.834 (vs 0.693), Hawar: 0.926 (vs 0.780), Sehat: 0.965 (vs 0.869). Dapat dilihat pada gambar berikut :



Gambar 11. Grafik F1-Score Perkelas

## 4. KESIMPULAN

Kesimpulan dari penelitian ini menunjukkan bahwa permasalahan utama dalam mendeteksi penyakit daun tomat yang selama ini masih bergantung pada pengamatan visual manual, yang bersifat subjektif dan tidak konsisten, dapat diatasi dengan sistem klasifikasi berbasis pengolahan citra digital dan algoritma Machine Learning. Melalui pipeline terpadu yang mencakup segmentasi citra berbasis HSV, ekstraksi fitur warna, bentuk, dan tekstur menggunakan GLCM serta LBP, kemudian diimplementasikan dalam bentuk GUI MATLAB, penelitian ini berhasil menghadirkan solusi yang efisien, konsisten, dan mudah digunakan. Hasil pengujian memperlihatkan bahwa algoritma Support Vector Machine (SVM) memberikan performa terbaik dengan akurasi sebesar 92,36%, sebaliknya, algoritma Naive Bayes hanya

mencapai akurasi 79,41%. Dengan demikian, penelitian ini tidak hanya menjawab permasalahan keterbatasan metode tradisional, tetapi juga menutup celah penelitian sebelumnya dengan menghadirkan sistem klasifikasi yang terintegrasi, komparatif, dan robust. Secara keseluruhan, SVM direkomendasikan sebagai algoritma utama untuk klasifikasi penyakit daun tomat karena mampu menangani pola kompleks dengan akurasi dan konsistensi yang lebih tinggi, sehingga dapat mendukung pertanian presisi melalui deteksi penyakit yang cepat, akurat, dan efisien.

## REFERENCES

- [1] J. Kusuma, R. Rosnelly, and B. H. Hayadi, "JOURNAL OF APPLIED COMPUTER SCIENCE AND TECHNOLOGY ( JACOST ) Klasifikasi Penyakit Daun Pada Tanaman Jagung Menggunakan Algoritma Support Vector Machine , K-Nearest Neighbors dan Multilayer Perceptron," vol. 4, no. 1, pp. 1–6, 2022.
- [2] E. S. Rahayu, O. Anugrah, A. Purnama, H. Zakaria, and P. Rosyani, "BULLETIN OF COMPUTER SCIENCE RESEARCH Klasifikasi Penyakit Jamur Pada Tanaman Tomat dengan Algoritma SVM," vol. 5, no. 4, pp. 756–762, 2025, doi: 10.47065/bulletinsr.v5i4.515.
- [3] A. Saleh, R. Eka Sari, R. Ramadani, F. Fujiati, and R. Lestari, "Analisis Komparatif Algoritma Klasifikasi Machine Learning untuk Memprediksi Diabetes," *Algoritm. J. Ilmu Komput. dan Inform.*, vol. 9, no. 1, p. 26, 2025, doi: 10.30829/algoritma.v9i1.23794.
- [4] R. Dewi, R. S. Hayati, A. Saleh, D. Y. H. Tanjung, and A. Jinan, "Enhancing Machine Learning Algorithm Performance for PCOS Diagnosis Using SMOTENC on Imbalanced Data," *JITK (Jurnal Ilmu Pengetah. dan Teknol. Komputer)*, vol. 11, no. 1, pp. 55–63, 2025, doi: 10.33480/jitk.v11i1.6676.ENHANCING.
- [5] P. A. Azli, M. Minarni, A. Syahrani, G. Y. Swara, and A. Anisya, "Segmentasi Citra Daun Tomat untuk Klasifikasi Penyakit Tanaman Menggunakan Support Vector Machine (SVM)," *J. Inform. J. Pengemb. IT*, vol. 10, no. 4, pp. 997–1008, 2025, doi: 10.30591/jpit.v10i4.9404.
- [6] S. W. Agusta and W. Kaswidjanti, "The Implementation of Color Feature Extraction and Gray Level Co-occurrence Matrix Combination in K-Nearest Neighbor Classification Method for Tomato Leaf Disease Identification," *Telematika*, vol. 20, no. 2, p. 250, 2023, doi: 10.31315/telematika.v20i2.10009.
- [7] J. Zheng and M. Du, "Study on Tomato Disease Classification based on Leaf Image Recognition based on Deep Learning Technology," *Int. J. Adv. Comput. Sci. Appl.*, vol. 14, no. 4, pp. 575–583, 2023, doi: 10.14569/IJACSA.2023.0140463.
- [8] U. Khultsum and A. Subekti, "Penerapan Algoritma Random Forest dengan Kombinasi Ekstraksi Fitur Untuk Klasifikasi Penyakit Daun Tomat," *J. Media Inform. Budidarma*, vol. 5, no. 1, p. 186, 2021, doi: 10.30865/mib.v5i1.2624.
- [9] A. Das, F. Pathan, J. R. Jim, M. M. Kabir, and M. F. Mridha, "Deep learning-based classification, detection, and segmentation of tomato leaf diseases: A state-of-the-art review," *Artif. Intell. Agric.*, vol. 15, no. 2, pp. 192–220, 2025, doi: 10.1016/j.aiaa.2025.02.006.
- [10] E. Widyastuti, A. Hermawan, and D. Avianto, "Klasifikasi Tomat Berdasarkan Varietas Dengan Ekstraksi Fitur Rgb Dan Algoritma Naïve Bayes," *Indones. J. Inf. Syst.*, vol. 8, no. 1, pp. 127–137, 2025, [Online]. Available: <http://jom.fti.budiluhur.ac.id/index.php/IDEALIS/indexEviWidyastuti%7Chttp://jom.fti.budiluhur.ac.id/index.php/IDEALIS/index%7C>
- [11] R. A. Saputra, "KLASIFIKASI PENYAKIT PADI MELALUI CITRA DAUN," vol. 12, no. 2, 2024.
- [12] O. Attallah, "Tomato Leaf Disease Classification via Compact Convolutional Neural Networks with Transfer Learning and Feature Selection," *Horticulturae*, vol. 9, no. 2, 2023, doi: 10.3390/horticulturae9020149.
- [13] C. Nyasulu *et al.*, "A comparative study of Machine Learning-based classification of Tomato fungal diseases: Application of GLCM texture features," *Heliyon*, vol. 9, no. 11, p. e21697, 2023, doi: 10.1016/j.heliyon.2023.e21697.
- [14] J. Khatib and S. Dalam, "Indonesian Journal of Computer Science," vol. 12, no. 1, pp. 1231–1240, 2023.
- [15] S. K. Parinduri, R. Rosnelly, A. Purnama, A. Sihotang, M. Chintya, and U. P. Utama, "PERBANDINGAN ALGORITMA SUPPORT VECTOR MACHINE DAN K-," vol. 13, no. 2, pp. 147–155, 2023.