

**IDENTIFIKASI PENYAKIT PADA TANAMAN TOMAT BERDASARKAN WARNA DAN
BENTUK DAUN DENGAN METODE NAIVE BAYES
CLASSIFIER BERBASIS WEB**

^[1]Fahri Alviansyah, ^[2]Ikhwan Ruslianto

^[3]Muhammad Diponegoro

^[1]^[2]^[3]Jurusan Sistem Komputer, Fakultas MIPA Universitas Tanjungpura

Jl. Prof. Dr. H. Hadari Nawawi, Pontianak

Telp./Fax.: (0561) 577963

e-mail:

^[1]fahri_alviansyah@student.untan.ac.id, ^[2]ikhwanruslianto@siskom.untan.ac.id

^[3]muhammad.diponegoro@siskom.untan.ac.id

Abstrak

Tomat merupakan salah satu tanaman hortikultura di Indonesia yang sangat rentan terserang penyakit, sehingga pengenalan penyakit pada tanaman tomat diperlukan untuk mencegah kerusakan yang lebih lanjut. Penyakit yang menyerang tanaman tomat dapat disebabkan oleh bakteri, jamur maupun virus. Sebagian besar gejala penyakit dapat dilihat pada perubahan warna dan bentuk pada daun. Kebanyakan yang dilakukan petani adalah mengenali gejala dengan mata telanjang dan langsung mengambil tindakan tanpa tahu cara penanggulangannya. Dari permasalahan tersebut, dilakukan penelitian untuk mengidentifikasi penyakit pada tanaman tomat dengan melihat warna dan bentuk daun. Penelitian ini bertujuan untuk membuat suatu aplikasi identifikasi penyakit pada tanaman tomat yang memanfaatkan pengolahan citra digital dengan proses ekstraksi fitur warna HSV dan bentuk menggunakan metode deteksi tepi operator sobel dan metode Naive Bayes Classifier untuk proses identifikasi. Berdasarkan hasil pengujian menggunakan 47 sampel citra daun, dimana masing-masing jenis daun digunakan 7 sampai 10 sampel citra, memberikan persentase keberhasilan dalam identifikasi sebesar 82,98%.

Kata kunci : penyakit tanaman tomat, citra digital, warna daun, bentuk daun, warna HSV, operator sobel, Naive Bayes Classifier.

1. PENDAHULUAN

Tomat merupakan salah satu tanaman hortikultura yang penting, yaitu terutama sebagai tanaman sayur. Walaupun demikian tanaman tomat rentan terhadap penyakit. Pengetahuan tentang penyakit dan cara penanggulangan diperlukan baik dalam proses pengendalian penyakit ataupun pada penelitian yang terkait dengannya. Pengendalian penyakit tanaman itu sendiri diperlukan untuk mencegah kerusakan pada tanaman tomat jika terserang penyakit.

Penyakit yang menyerang tanaman tomat bisa disebabkan oleh jamur atau cendawan, bakteri dan virus. Gejala yang ditimbulkan bisa dilihat dari perubahan bentuk dan warna daun. Kebanyakan yang dilakukan petani adalah mengenali gejala dengan mata

telanjang dan langsung mengambil tindakan tanpa tahu cara penanggulangannya. Oleh sebab itu diperlukan bantuan teknologi pengolahan citra yang dapat mengenali penyakit pada tanaman tomat menurut bentuk dan warna daun

2. LANDASAN TEORI

2.1. Penyakit Pada Tanaman Tomat

Tanaman dikatakan terserang penyakit bila pertumbuhannya menyimpang dari keadaan normal. Penyebabnya terdiri dari beberapa macam, diantaranya jamur atau cendawan, bakteri, dan virus. . Beberapa jenis penyakit yang terdapat pada tanaman tomat diantaranya penyakit *yellow leaf curl*, *potato virus y*, *tomato mozaik virus*, dan bercak daun. [1].

2.2. Pengolahan Citra Digital

Secara umum, pengolahan citra digital merupakan pemrosesan gambar 2 dimensi yang menggunakan komputer. Citra digital merupakan sebuah larik (array) yang berisi nilai-nilai real maupun kompleks yang direpresentasikan dengan deretan bit tertentu [2].

Suatu citra dapat didefinisikan sebagai fungsi $f(x,y)$ berukuran M baris dan N kolom, dengan x dan y adalah koordinat spasial, dan amplitudo f di titik koordinat (x,y) dinamakan intensitas atau tingkat keabuan dari citra pada titik tersebut. Apabila nilai x,y, dan nilai amplitudo f secara keseluruhan berhingga (finite) dan bernilai diskrit maka dapat dikatakan bahwa citra tersebut adalah citra digital. Sebuah citra digital dapat dituliskan dalam bentuk matriks berikut

$$f = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,M-1) \\ f(1,0) & \dots & \dots & f(1,M-1) \\ \dots & \dots & \dots & \dots \\ f(N-1,0) & f(N-1,1) & \dots & f(N-1,M-1) \end{bmatrix}$$

Gambar 1. Representasi Matriks Citra Digital
Sumber : [2].

2.3. Model Warna HSV

Model HSV (*Hue Saturation Value*) merupakan warna yang terdiri dari tiga warna utama yaitu *hue*, *saturation*, dan *value* (atau disebut juga *brighness*). *Hue* merupakan dengan sudut dari 0 sampai 360 derajat, biasanya 0 adalah merah, 60 derajat adalah kuning, 120 derajat adalah hijau, 180 derajat adalah cyan, 240 derajat adalah biru, dan 300 derajat adalah warna magenta. *Hue* menunjukkan jenis warna atau corak warna yaitu tempat warna tersebut ditemukan dalam spektrum warna. *Saturation* dari suatu warna adalah ukuran seberapa besar kemurnian dari warna tersebut. *Value* atau disebut juga *intensity* yaitu ukuran seberapa besar kecerahan dari suatu warna atau seberapa besar cahaya datang dari suatu warna. Value dapat bernilai dari 0-100% [2].

2.3.1. Konversi RGB ke HSV

RGB merupakan singkatan dari Red, Green, Blue. Warna-warna yang dibentuk oleh model warna merupakan hasil campuran dari

warna-warna primer merah, hijau, dan biru berdasarkan komposisi tertentu. Perhitungan konversi RGB menjadi HSV dapat dirumuskan sebagai berikut [2] :

$$H = \tan \left[\frac{3(G-B)}{(R-G)+(R-B)} \right] \quad (1)$$

$$S = 1 - \frac{\min(R,G,B)}{V} \quad (2)$$

$$V = \frac{R+G+B}{3} \quad (3)$$

Tapi untuk rumus di atas, jika $S = 0$ maka *hue* tidak dapat ditentukan. Untuk itu maka perlu melakukan normalisasi nilai RGB terlebih dahulu dengan rumus berikut :

$$r = \frac{R}{R+G+B} \quad (4)$$

$$g = \frac{G}{R+G+B} \quad (5)$$

$$b = \frac{B}{R+G+B} \quad (6)$$

Dengan memanfaatkan nilai r,g,dan b yang telah dinormalisasi, rumus konversi RGB ke HSV sebagai berikut :

$$V = \max(r, g, b) \quad (7)$$

$$S = \begin{cases} 0 & \text{jika } V = 0 \\ V - \frac{\min(r,g,b)}{V} & \text{jika } V > 0 \end{cases} \quad (8)$$

$$H = \begin{cases} 0 & \text{jika } S = 0 \\ \frac{60 \times (g-b)}{S \times V} & \text{jika } V = r \\ 60 \times \left[2 + \frac{(b-r)}{S \times V} \right] & \text{jika } V = g \\ 60 \times \left[4 + \frac{(r-g)}{S \times V} \right] & \text{jika } V = b \end{cases} \quad (9)$$

2.4. Deteksi Tepi Operator Sobel

Operator Sobel adalah salah satu operator yang menghindari adanya perhitungan gradient di titik interpolasi. Operator sobel memiliki kelebihan yaitu kemampuan untuk mengurangi derau atau *noise* sebelum melakukan proses perhitungan algoritma deteksi tepi. Misalkan susunan dari piksel-piksel di sekitar piksel $p_o = (x,y)$ seperti pada gambar 2.

P ₁	P ₂	P ₃
P ₈	(x,y)	P ₄
P ₇	P ₆	P ₅

Gambar 2. Susunan Piksel-Piksel Tetangga dari Piksel po = (x,y).

Berdasarkan susunan piksel-piksel tetangga tersebut, maka nilai besaran gradien yang dihitung dengan operator sobel pada persamaan (10).

$$M = \sqrt{gx^2 + gy^2} \quad (10)$$

Dimana M adalah besaran gradien yang dihitung pada titik tengah jendela dan turunan parsial pada persamaan (11) dan (12).

$$G_x = (p_3 + cp_4 + p_5) - (p_1 + cp_8 + p_7) \quad (11)$$

$$G_y = (p_1 + cp_2 + p_5) - (p_1 + cp_6 + p_5) \quad (12)$$

Dengan c suatu konstanta bernilai 2. Seperti dalam perhitungan gradien pada operator lainnya, g_x dan g_y diimplementasikan menggunakan jendela seperti pada gambar 3.

G_x:

-1	0	1
-2	0	2
-1	0	1

G_y :

1	2	1
0	0	0
-1	-2	-1

Gambar 3. Matriks Operator Sobel. Sumber : [3].

Keterangan :

M : Besar gradien.

G_x : Gradien arah vertikal.

G_y : Gradien arah horizontal.

2.5. Naive Bayes Classifier

Naive Bayes merupakan metode klasifikasi dengan metode probabilitas yang ditemukan oleh ilmuwan Inggris bernama Thomas Bayes. *Naive Bayes* didasari dari teorema Bayes (aturan *Bayes*) dengan asumsi atribut yang *independent* atau atribut yang bebas [4].

Algoritma *naive bayes* adalah sebagai berikut [5] :

1. Set data latih dengan label kelas yang telah ditentukan. masing-masing data latih diwakili oleh n-vektor atribut, $X = (X_1, X_2, X_3, \dots, X_n)$.
2. Misalkan bahwa terdapat kelas sebanyak m, $C_1, C_2, C_3, \dots, C_m$. Terdapat sebuah data X, sistem klasifikasi akan memprediksi bahwa X tergolong dalam kelas yang memiliki nilai probabilitas posterior tertinggi, sesuai dengan kondisi X. Kelas C_i maksimal dari $P(C_i|X)$ disebut *maximum posteriori hypothesis*. Berikut rumus menghitung nilai probabilitas posterior berdasarkan teorema *bayesian*.

$$P(C|X) = \frac{P(X|C_i) \times P(C_i)}{P(X)} \quad (13)$$
3. Nilai P(X) konstan atau tetap untuk semua kelas, hanya $P(X|C_i)$ dan $P(C_i)$ yang lebih diutamakan. Untuk menghitung probabilitas prior adalah dengan menghitung banyaknya data pada kelas C_i dibagi dengan banyaknya jumlah data latih.
4. Menghitung $P(X|C_i)$ dengan rumus sebagai berikut

$$P(X|C_i) = \prod_{k=1}^n P(x_k|C_i) \quad (14)$$
 Untuk tiap-tiap atribut, perlu diperhatikan lagi apakah bersifat kategori atau kontinu.
 - a. Jika A_k bersifat kategori, maka hitung jumlah banyaknya data dari kelas C_i yang memiliki nilai x_k untuk atribut ak dibagi dengan $|C_i, D|$, banyak nya jumlah data kelas C_i pada data latih
 - b. Jika A_k bersifat kontinu, maka biasanya diasumsikan dengan distribusi *gaussian* dengan nilai rata-rata (μ) dan standar deviasi (σ). Berikut rumus dari distribusi *gaussian*

$$g(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (15)$$
 Jadi,

$$P(X|C_i) = g(x, \mu, \sigma) \quad (16)$$
5. Untuk memprediksi label kelas X, $P(X|C_i) \times P(C_i)$ dihitung untuk setiap kelas. Label kelas yang diprediksi adalah

kelas yang memiliki nilai $P(X|Ci) \times P(Ci)$ maksimum.

3. METODOLOGI PENELITIAN

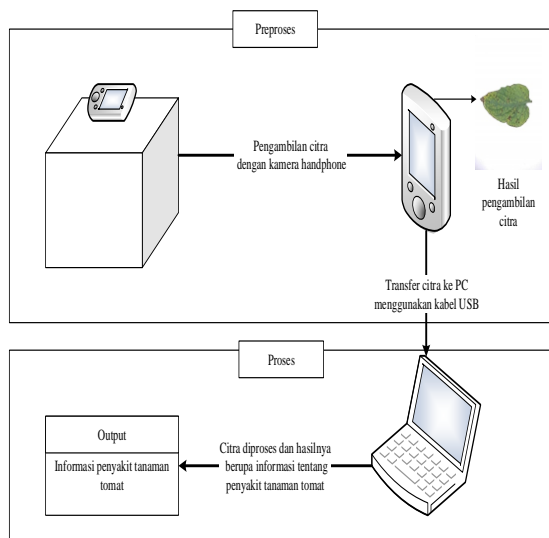
Penelitian ini menggunakan metodologi penelitian yang mencakup studi literatur, kemudian dilakukan analisa kebutuhan akan komponen-komponen yang dibutuhkan dalam pembuatan aplikasi identifikasi penyakit pada tanaman tomat berdasarkan warna dan bentuk daun, selanjutnya perancangan sistem, kemudian dilakukan tahap implementasi dan pengujian, serta analisis. Tahap terakhir adalah kesimpulan dan saran.

4. PERANCANGAN SISTEM

Perancangan sistem yang dibuat, mulai dari perancangan sistem secara umum, perancangan perangkat keras, perancangan *flowchart*, dan perancangan *interface*.

4.1. Perancangan Sistem secara umum

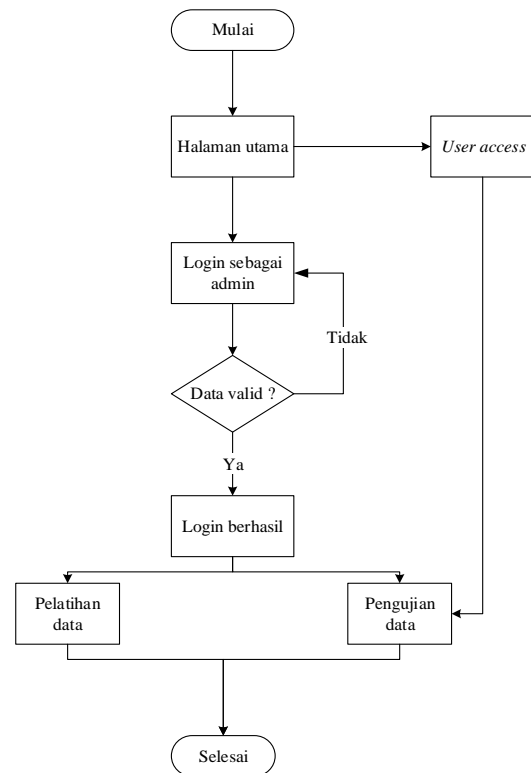
Alur sistem secara umum dapat dilihat pada Gambar 4.



Gambar 4. Alur Sistem Secara Umum.

4.2. Perancangan Flowchart

Flowchart sistem identifikasi penyakit pada tanaman tomat dapat dilihat pada Gambar 5.

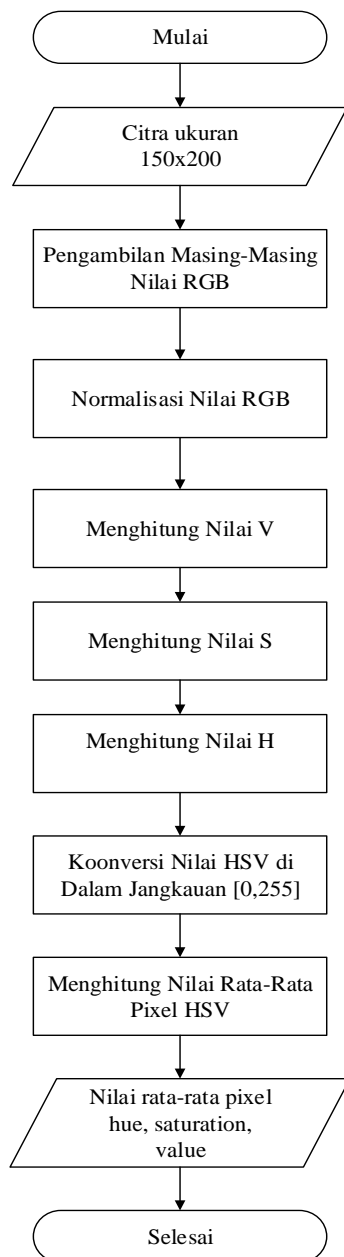


Gambar 5. *Flowchart* Sistem Identifikasi Penyakit Pada Tanaman Tomat.

Berikut merupakan penjelasan dari Gambar 5 *flowchart* perancangan sistem :

1. Masuk sebagai admin :
 - a. Dimulai dari halaman utama, jika masuk sebagai admin, maka harus *login* terlebih dahulu dan sistem akan memeriksa akun yang terdapat pada *database*.
 - b. Jika *username* dan *password* yang dimasukkan tidak *valid* maka admin akan memasukkan *username* dan *password* kembali sampai berhasil *login*.
 - c. Setelah admin *login*, admin dapat melakukan pelatihan data dan pengujian data.
 - d. Pelatihan data berfungsi sebagai tempat admin memasukkan data latih, sedangkan pengujian data berfungsi sebagai tempat menguji data baru
2. *User access* :
 - a. Jika pada halaman utama masuk sebagai *user*, maka *user* hanya bisa melakukan pengujian pada data baru.

Flowchart ekstraksi fitur warna dapat dilihat pada Gambar 6.



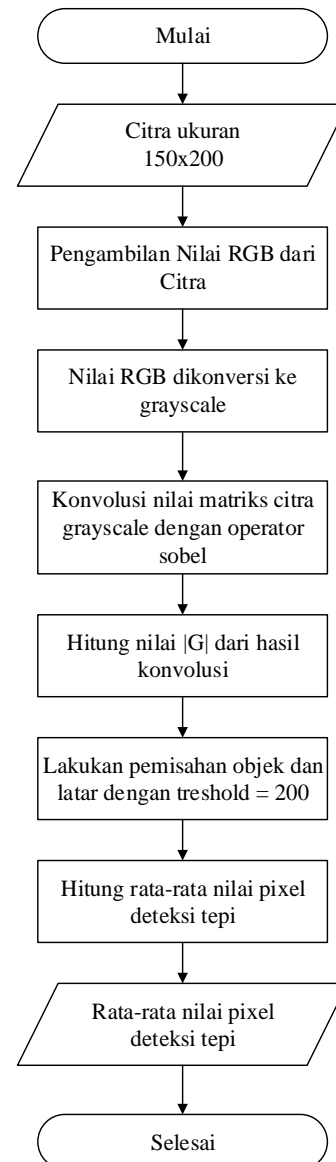
Gambar 6. Flowchart Ekstraksi Fitur Warna.

Penjelasan dari masing-masing tahapan pada Gambar 6 adalah sebagai berikut :

1. Proses dimulai
2. Pertama citra asli diambil masing-masing nilai RGB.
3. Selanjutnya normalisasi masing-masing nilai pada matriks RGB.
4. Setelah didapat nilai normalisasi, selanjutnya menghitung nilai V.
5. Dari nilai V selanjutnya menghitung nilai S.

6. Selanjutnya menghitung nilai H .
7. Setelah didapat nilai HSV, konversikan nilai HSV dalam jangkauan [0,255].
8. Selanjutnya menghitung rata-rata masing-masing nilai pada matriks HSV.
9. *Output* berupa nilai rata-rata dari matriks HSV.
10. Proses selesai.

Berikut merupakan *flowchart* ekstraksi fitur bentuk.



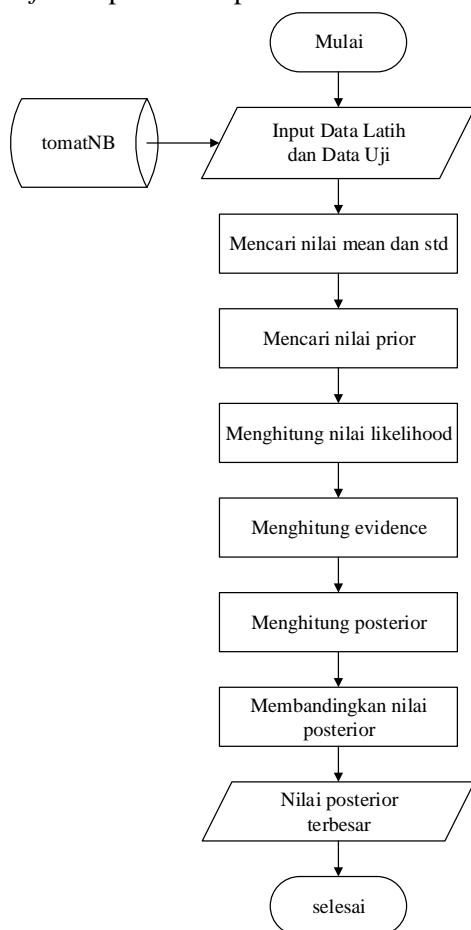
Gambar 7. Flowchart Ekstraksi Fitur Bentuk.

Penjelasan dari masing-masing tahapan pada Gambar 7 adalah sebagai berikut :

1. Proses dimulai
2. Citra asli diambil masing-masing nilai RGB.

3. Citra yang berada pada ruang warna RGB dikonversi ke warna *grayscale*.
4. Setelah citra dikonversi ke *grayscale*, selanjutnya matriks citra *grayscale* di konvolusi dengan operator sobel.
5. Hitung nilai gradien $|G|$ dari hasil konvolusi.
6. Selanjutnya adalah pemisahan latar dan objek dengan *threshol* 200. Apabila nilai $pixel \leq 200$, maka set nilai $pixel = 1$. Apabila nilai $pixel grayscale > 200$, maka set nilai $pixel = 0$.
7. Nilai $pixel$ hasil *threshol* dihitung nilai rata-ratanya.
8. Ekstraksi fitur bentuk ini akan menghasilkan nilai rata-rata dari pixel deteksi tepi.
9. Proses selesai.

Berikut merupakan *flowchart* dari identifikasi dengan menggunakan *naive bayes classifier* dapat dilihat pada Gambar 8.



Gambar 8. Flowchart Identifikasi dengan Naive Bayes Classifier.

Penjelasan rincian dari masing masing tahapan pada Gambar 8 adalah sebagai berikut :

1. Proses dimulai
2. *Input* berupa data latih yang ada pada *database* dan data uji yang berupa data baru.
3. Sistem akan menghitung nilai mean dan standar deviasi dari data latih yang berada dalam *database*.
4. Selanjutnya sistem juga menghitung nilai prior dari data latih yang berada dalam *database*.
5. Dilanjutkan dengan menghitung nilai *likelihood* dengan menggunakan nilai mean dan standar deviasi yang telah dihitung sebelumnya.
6. Setelah menghitung nilai *likelihood*, maka sistem akan menghitung nilai *evidence* terlebih dahulu.
7. Selanjutnya menghitung nilai posterior masing-masing kelas dengan menggunakan hasil perhitungan prior, *likelihood*, dan *evidence*.
8. Setelah nilai posterior masing-masing kelas selesai dihitung, maka nilai masing-masing kelas tersebut akan dibandingkan.
9. Keluaran dari algoritma *Naive Bayes* adalah nilai posterior terbesar.
10. Proses selesai.

5. HASIL DAN PEMBAHASAN

Hasil dan pembahasan meliputi proses pelatihan dan pengujian data.

Pelatihan data dilakukan dengan menentukan kelas penyakit dan melakukan ekstraksi fitur bentuk dan warna dari citra daun tomat. Fitur bentuk menggunakan deteksi tepi sobel dan fitur warna menggunakan warna HSV. Hasil ekstraksi disimpan ke dalam *database* sebagai data latih. Berikut tabel dari kelas penyakit.

Tabel 1. Kelas Penyakit

Nama Penyakit	Kelas
Daun Normal	1

<i>Yellow Leaf Curl</i>	2
<i>Potato Virus Y</i>	3
<i>Mozaik Virus</i>	4
Bercak Daun	5

Pengujian dilakukan dengan tujuan untuk mengetahui keberhasilan sistem dalam identifikasi penyakit pada daun tomat. Penelitian yang dilakukan menggunakan jumlah data gambar daun sebanyak 122 gambar. Pengujian dilakukan dengan cara memisahkan data menjadi dua bagian yaitu 75 gambar digunakan sebagai data latih 47 gambar digunakan sebagai data uji. Data latih digunakan sebagai data rujukan identifikasi yang sesuai atau yang telah ditentukan sebelumnya, sedangkan data uji digunakan untuk menguji ketepatan sistem dalam melakukan identifikasi penyakit pada daun tanaman tomat. Hasil identifikasi tersebut akan dicatat dan dibandingkan dengan identifikasi atau jenis penyakit yang sebenarnya. Dari hasil identifikasi kemudian dihitung akurasi.

5.1. Proses Pelatihan Data

Berikut merupakan contoh pelatihan data daun normal sample pertama dapat dilihat pada Gambar 9.



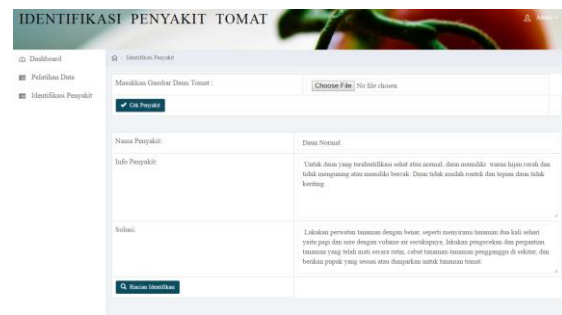
Gambar 9. Contoh Pelatihan Data.

Pada Gambar 9, setelah citra latih dimasukkan maka akan keluar nilai rata-rata hasil ekstraksi fitur bentuk dengan deteksi tepi sobel dan fitur warna HSV. Pada daun normal sampel pertama memiliki rata-rata *hue* sebesar 2360,1093, untuk rata-rata *saturation* sebesar 3371,9854, rata-rata *value* sebesar

13340,5629, dan rata-rata deteksi tepi sobel sebesar 26,3769.

5.2. Proses Pengujian Data

Pada proses pengujian sistem, tiap jenis daun memiliki 10 citra yang digunakan untuk pengujian. Berikut merupakan contoh hasil pengujian untuk daun normal sampel pertama dapat dilihat pada Gambar 10.



Gambar 10. Contoh Hasil Pengujian Daun Normal

Gambar 10 menunjukkan bahwa citra teridentifikasi kedalam kelas daun normal. Bagian info merupakan informasi mengenai gejala atau ciri dari penyakit yang teridentifikasi. Bagian solusi merupakan solusi atau penanggulangan yang dilakukan terhadap jenis penyakit yang telah teridentifikasi. Pada tombol rincian identifikasi berfungsi untuk menampilkan rincian pengujian. Berikut rincian pengujian daun normal dapat dilihat pada gambar 11.



Gambar 11. Rincian Hasil Pengujian Daun Normal

Pada gambar 11 rincian pengujian terdapat citra dalam warna HSV dan deteksi tepi sobel. Di samping citra warna HSV dan deteksi tepi sobel merupakan hasil rata-rata dari citra yang diuji yaitu memiliki nilai rata-rata *hue* sebesar 2391,5216, rata-rata

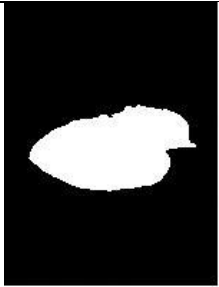
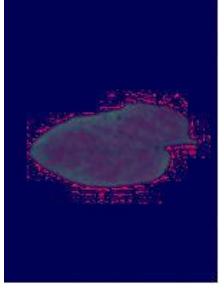
saturation sebesar 1284,9045, rata-rata *value* sebesar 12907,5886 dan rata-rata deteksi tepi sobel sebesar 23,6784.

Bagian *likelihood* merupakan bagian perhitungan dari algoritma *naive bayes classifier* yang digunakan untuk menghitung nilai posterior. Pada citra uji daun normal sampel pertama memiliki nilai *likelihood* untuk kelas 1 sebesar 1.1909327233693, untuk kelas 2 sebesar 1.1247324739423, untuk kelas 3 sebesar 1.146841477221, untuk kelas 4 sebesar 1.0611106995462, dan untuk kelas 5 sebesar 1.1132245986113.

Bagian posterior adalah hasil dari algoritma *naive bayes classifier*. Pada citra uji daun normal sampel pertama memiliki nilai posterior untuk kelas 1 sebesar 0,042255317042387, untuk kelas 2 sebesar 0,039906475270783, untuk kelas 3 sebesar 0,040690921717419, untuk kelas 4 sebesar 0,037649120010361, dan untuk kelas 5 sebesar 0,03949816595905. Penentuan kelas identifikasi berdasarkan nilai posterior terbesar yaitu 0,042255317042387 berada di kelas 1 dimana kelas 1 merupakan kelas daun normal.

Hasil rincian identifikasi yaitu berupa gambar deteksi tepi sobel dan gambar dengan warna HSV dapat dilihat pada tabel berikut.

Tabel 1. Citra Hasil Identifikasi

Jenis Citra	Hasil
Deteksi Tepi Sobel	
Warna HSV	

5.3. Hasil Pengujian

Berikut merupakan tabel hasil pengujian identifikasi penyakit pada tanaman tomat

Tabel 3. Hasil Pengujian Identifikasi Penyakit Pada Tanaman Tomat

Jenis Penyakit	Data Ke-	Hasil Identifikasi	Keberhasilan Identifikasi
Daun Normal	1	Daun Normal	Ya
	2	Daun Normal	Ya
	3	Bercak	Tidak
	4	Daun Normal	Ya
	5	Daun Normal	Ya
	6	Daun Normal	Ya
	7	Daun Normal	Ya
	8	Daun Normal	Ya
	9	Penyakit <i>Yellow Leaf Curl</i>	Tidak
	10	Penyakit <i>Yellow Leaf Curl</i>	Tidak
Yellow Leaf Curl	11	Penyakit <i>Yellow Leaf Curl</i>	Ya
	12	Penyakit <i>Yellow Leaf Curl</i>	Ya
	13	Penyakit <i>Yellow Leaf Curl</i>	Ya
	14	Penyakit <i>Yellow Leaf Curl</i>	Ya
	15	Penyakit <i>Yellow Leaf Curl</i>	Ya
	16	Penyakit <i>Yellow Leaf Curl</i>	Ya
	17	Penyakit <i>Yellow Leaf Curl</i>	Ya
Potato Virus Y	18	Penyakit Bercak	Tidak
	19	Penyakit Bercak	Tidak
	20	Penyakit <i>Potato Virus Y</i>	Ya
	21	Daun Normal	Tidak
	22	Penyakit <i>Potato Virus Y</i>	Ya
	23	Penyakit Bercak	Tidak
	24	Penyakit Bercak	Tidak
	25	Penyakit <i>Potato Virus Y</i>	Ya
	26	Penyakit <i>Potato Virus Y</i>	Ya
	27	Penyakit <i>Potato Virus Y</i>	Ya
Mozaik Virus	28	Penyakit <i>Mozaik Virus</i>	Ya
	29	Penyakit <i>Mozaik Virus</i>	Ya
	30	Penyakit <i>Mozaik Virus</i>	Ya
	31	Penyakit <i>Mozaik Virus</i>	Ya
	32	Penyakit <i>Mozaik Virus</i>	Ya
	33	Penyakit <i>Mozaik Virus</i>	Ya
	34	Penyakit <i>Mozaik Virus</i>	Ya
	35	Penyakit <i>Mozaik Virus</i>	Ya
	36	Penyakit <i>Mozaik Virus</i>	Ya
	37	Penyakit <i>Mozaik Virus</i>	Ya
Bercak	38	Penyakit Bercak	Ya
	39	Penyakit Bercak	Ya
	40	Penyakit Bercak	Ya
	41	Penyakit Bercak	Ya
	42	Penyakit Bercak	Ya
	43	Penyakit Bercak	Ya
	44	Penyakit Bercak	Ya
	45	Penyakit Bercak	Ya
	46	Penyakit Bercak	Ya
	47	Penyakit Bercak	Ya

Tabel 3 merupakan tabel hasil pengujian. Dari 47 data pengujian terdapat 39

data yang berhasil teridentifikasi dan 8 data yang tidak berhasil teridentifikasi.

Perhitungan persentase keberhasilan proses pengujian dilihat sebagai berikut :

Dari 47 kali pengujian, terdapat 39 kali pengujian yang berhasil, dan terdapat 8 kali pengujian yang tidak teridentifikasi.

$$\begin{aligned} \text{\% keberhasilan identifikasi pengujian} \\ = \frac{39}{47} \times 100\% = 82,98\% \end{aligned}$$

$$\begin{aligned} \text{\% kegagalan identifikasi pengujian} \\ = \frac{8}{47} \times 100\% = 17,02\% \end{aligned}$$

Dari hasil pengujian diperoleh akurasi keberhasilan sebesar 82,98% dan akurasi kegagalan 17,02% dikarenakan ada daun yang memiliki bentuk yang hampir sama pada kelas yang berbeda dan ada kemungkinan nilai citra memiliki kesamaan dengan citra lain pada kelas yang berbeda.

6. Kesimpulan dan Saran

6.1. Kesimpulan

Berdasarkan pengujian dan analisis terhadap Sistem Identifikasi Penyakit Pada Tanaman Tomat Berdasarkan Warna dan Bentuk Daun Menggunakan Metode *Naive Bayes Classifier* yang telah dilakukan, dapat ditarik kesimpulan sebagai berikut :

1. Nilai rata-rata pixel warna HSV didapatkan dari normalisasi nilai RGB. Selanjutnya konversi citra RGB ke HSV dan masing-masing nilai pixel HSV dihitung rata-ratanya. Nilai rata-rata deteksi tepi sobel didapatkan dari konversi citra RGB ke grayscale terlebih dahulu. Selanjutnya menghitung konvolusi menggunakan operator sobel. Hasil nilai gradien akan digunakan pada proses tresholding untuk konversi menjadi citra biner. Nilai pixel dari citra biner dihitung rata-ratanya sebagai nilai rata-rata untuk deteksi tepi sobel.
2. Dari 47 data pengujian menghasilkan 39 pengujian yang teridentifikasi dan 8 pengujian hasilnya tidak teridentifikasi. Persentase keberhasilan sistem dalam

mengidentifikasi penyakit pada tanaman tomat adalah sebesar 82,98%, dan persentase kegagalan sistem dalam mengidentifikasi penyakit pada tanaman tomat adalah sebesar 17,02%.

3. Metode *Naive Bayes Classifier* dapat digunakan untuk mengidentifikasi penyakit pada tanaman tomat dengan persentase keberhasilan sebesar 82,98%.
4. Cahaya dan *noise* dapat mempengaruhi kesalahan identifikasi, semakin tinggi cahaya dan *noise* maka objek semakin tidak jelas sehingga nilai fitur ekstraksi citra bisa berbeda jauh.

6.2. Saran

Berdasarkan pengujian dan analisis terhadap Sistem Identifikasi Penyakit Pada Tanaman Tomat Berdasarkan Warna dan Bentuk Daun Menggunakan Metode *Naive Bayes Classifier* yang telah dilakukan, terdapat beberapa saran yang bisa digunakan untuk penelitian yang lebih lanjut, antara lain :

1. Untuk ekstraksi ciri citra hanya menggunakan fitur warna HSV dan deteksi tepi sobel. Untuk penelitian selanjutnya bisa dikembangkan contohnya dengan menambahkan fitur tekstur dengan metode GLCM.
2. Identifikasi penyakit tanaman tomat ini menggunakan metode naive bayes classifier, untuk penelitian selanjutnya dapat mencoba metode yang lain seperti Probabilistic Neural Network atau Support Vector Machine.
3. Aplikasi dapat dibuat berbasis android supaya aplikasi identifikasi dapat digunakan secara mobile.

DAFTAR PUSTAKA

- [1] Penebar Swadaya, T. P. *Budi Daya Tomat Secara Komersial*. Jakarta: Penebar Swadaya, 2012.
- [2] Putra D. *Pengolahan Citra Digital*. Yogyakarta: Penerbit Andi, 2010.
- [3] Ahmad, U. *Pengolahan Citra Digital dan Teknik Pemograman*. Yogyakarta: Graha Ilmu, 2005.
- [4] Chakrabarti, S., Cox, E., & Frank, E. (2009). *Data Mining : Know It All*. United States: Morgan Kaufmann.
- [5] Han, J., Kamber, M., & Pei, J. *Data Mining Concepts and Techniques*. USA: Morgan Kaufmann, 2011.