

BAB I

PENDAHULUAN

1.1 Latar Belakang

Kelapa sawit merupakan salah satu tanaman perkebunan yang memiliki peranan penting dalam sektor pertanian dan perekonomian di Indonesia. Kelapa sawit sebagai komoditas unggulan subsektor perkebunan mempunyai peran penting dalam pembangunan ekonomi yang dipresentasikan dengan nilai PDRB. Meningkatnya nilai PDRB disebabkan oleh meningkatnya output sektor-sektor, salah satunya yaitu sub sektor perkebunan [1], Kelapa sawit (*Elaeis guinensis* Jacq) merupakan tumbuhan tropis yang berasal dari benua Afrika. Pertama kali ditemukan oleh Jacquin pada tahun 1763 di hutan belantara Afrika Barat dan tersebar luas ke seluruh dunia, terutama kawasan tropis yang sesuai dengan pertumbuhannya. Kelapa sawit pertama kali masuk ke Indonesia pada tahun 1848 yang dibawa oleh Dr. D. T. Pryce yang berasal dari Belanda [2].

Kelapa sawit termasuk tanaman yang mempunyai perakaran yang dangkal (akar serabut), sehingga mudah mengalami cekaman kekeringan. Adapun penyebab tanaman mengalami kekeringan diantaranya transpirasi tinggi dan diikuti dengan ketersediaan air tanah yang terbatas pada saat musim kemarau [3].

Dalam praktiknya, tanaman kelapa sawit tidak terlepas dari berbagai gangguan, salah satunya adalah serangan penyakit pada daun. Penyakit bercak daun adalah spora patogen yang umum menginfeksi tanaman kelapa sawit terutama pada fase pembibitan dan tanaman belum menghasilkan. Penyakit ini,

ditandai dengan munculnya bercak-bercak kuning kecoklatan di permukaan daun, yang dapat mengganggu proses fotosintesis tanaman dan mempengaruhi lamanya fase tanaman belum menghasilkan (TBM) [4].

Petani seringkali hanya mengandalkan pengalaman pribadi atau informasi dari sesama petani, yang kadang tidak akurat. Oleh karena itu, dibutuhkan solusi berbasis teknologi informasi yang dapat menjembatani kebutuhan diagnosa cepat dan tepat tanpa ketergantungan penuh pada pakar[5], Seiring dengan perkembangan teknologi, pemanfaatan Pengolahan *Citra* Digital menjadi salah satu solusi yang dapat digunakan untuk mendeteksi penyakit pada tanaman secara otomatis. Teknologi ini memungkinkan komputer untuk menganalisis *Citra* daun dan mengekstraksi informasi penting seperti warna, tekstur, dan bentuk yang dapat digunakan sebagai dasar dalam proses identifikasi penyakit. Dengan menggunakan pendekatan ini, proses deteksi dapat dilakukan secara lebih objektif dan konsisten dibandingkan dengan metode manual.

Salah satu metode yang dapat digunakan dalam proses klasifikasi *Citra* adalah *Support Vector Machine (SVM)* adalah teknik belajar yang digunakan untuk klasifikasi. *SVM* menemukan sebuah *hyperplane* yang memiliki kemungkinan fraksi poin terbesar dari kelas yang sama pada bidang yang sama. *Hyperplane* adalah fungsi yang digunakan untuk membedakan antar fitur. *SVM* mampu menangani data yang kompleks dan menghasilkan klasifikasi yang baik. *SVM* dapat digunakan dengan data klinis atau dataset yang beragam untuk meningkatkan akurasi klasifikasi penyakit pada daun kelapa sawit [6].

Berdasarkan permasalahan tersebut, penelitian ini dilakukan dengan menerapkan metode *Support Vector Machine* untuk mendeteksi dini penyakit daun kelapa sawit berbasis pengolahan *Citra* dapat menghasilkan suatu sistem yang mampu membantu proses identifikasi penyakit secara cepat dan akurat, sehingga dapat mendukung upaya peningkatan produktivitas serta kualitas hasil perkebunan kelapa sawit di Indonesia..

1.2 Rumusan Masalah

Berdasarkan uraian latar belakang diatas, maka rumusan masalah dalam penelitian ini adalah sebagai berikut:

Bagaimana merancang , membangun dan menerapkan metode *Support Vector Machine* (SVM) dalam proses klasifikasi penyakit daun kelapa sawit?

1.3 Tujuan Penelitian

Tujuan dari penelitian ini adalah sebagai berikut:

1. Untuk mengklasifikasikan penyakit pada daun kelapa sawit berdasarkan *Citra* daun secara tepat dan efisien.
2. Untuk menerapkan metode *Support Vector Machine* (SVM) dalam proses klasifikasi penyakit pada daun kelapa sawit.

1.4 Batasan Penelitian

Agar penelitian lebih terarah dan tidak melebar dari tujuan yang ingin dicapai, maka ditetapkan beberapa batasan penelitian sebagai berikut:

1. Penelitian hanya berfokus pada deteksi dini penyakit pada daun kelapa sawit, bukan pada bagian tanaman lainnya (buah, batang, atau bunga).
2. Data yang digunakan berupa *Citra* digital daun kelapa sawit dengan format gambar tertentu (misalnya JPG/PNG) yang diambil sebagai data masukan yang terdiri dari 5.000 *dataset* yang telah disiapkan atau hasil pengambilan gambar secara langsung.
3. Variabel yang digunakan dalam deteksi penyakit terbatas pada ciri visual daun, seperti warna, tekstur, dan bentuk daun yang terinfeksi penyakit.
4. Pengujian sistem dilakukan pada sejumlah data uji tertentu yang telah disiapkan, sehingga hasil penelitian hanya berlaku pada dataset yang digunakan dan tidak mewakili semua varietas kelapa sawit secara umum.
5. Penelitian ini tidak membahas aspek budidaya, pengolahan pasca panen, atau distribusi kelapa sawit, melainkan hanya pada tahap deteksi dini penyakit daun sawit berbasis *Citra*.

1.5 Manfaat Penelitian

Manfaat Teoritis

1. Memberikan kontribusi dalam pengembangan ilmu pengetahuan, khususnya di bidang kecerdasan buatan (*Artificial Intelligence*) dan pengolahan *Citra* digital.
2. Menjadi referensi penelitian terkait penerapan metode *Support Vector Machine* (SVM) dalam deteksi dini penyakit tanaman, khususnya pada kelapa sawit.

Manfaat Praktis

1. Membantu petani, pelaku usaha agribisnis, dan pihak terkait dalam mendeteksi penyakit pada daun kelapa sawit secara cepat, akurat, dan efisien.
2. Meningkatkan kualitas hasil perkebunan kelapa sawit melalui deteksi dini penyakit yang lebih akurat dan terstandar.
3. Mendukung peningkatan daya saing komoditas kelapa sawit di pasar domestik maupun internasional dengan memanfaatkan teknologi berbasis kecerdasan buatan untuk deteksi penyakit.

1.6 Sistematika Penulisan

Sistematika penulisan dari skripsi ini terdiri dari enam bagian utama, sebagai berikut:

BAB 1 PENDAHULUAN

Bab ini berisi latar belakang, rumusan masalah, tujuan penelitian, batasan masalah, manfaat penelitian dan sistematika penulisan.

BAB 2 LANDASAN TEORI

Bab ini berisi teori-teori yang digunakan pada penelitian ini. Teori-teori

BAB 3 METODOLOGI PENELITIAN

Bab ini berisi tahapan – tahapan yang dilakukan dalam penelitian ini, termasuk metode pengumpulan data, metode pengembangan aplikasi, serta alat dan teknik yang digunakan untuk implementasi dan pengujian

BAB 4 ANALISA DAN PERANCANGAN

Bab ini berisi analisa dan perancangan implementasi. Dalam bab ini, akan dijelaskan proses analisa kebutuhan, perancangan sistem, serta desain antarmuka pengguna.

BAB 5 IMPLEMENTASI DAN PENGUJIAN

Bab ini berisi implementasi dari hasil analisa dan perancangan yang telah dilakukan. Selain itu, bab ini juga akan menjelaskan proses pengujian aplikasi, termasuk metode pengujian, hasil pengujian, dan evaluasi terhadap aplikasi yang telah dikembangkan.

BAB 6 PENUTUP

Bab ini berisi kesimpulan dari hasil penelitian yang telah dilakukan dan saran-saran untuk pengembangan aplikasi atau penelitian selanjutnya. Bab ini juga akan merangkum kontribusi penelitian dan manfaat aplikasi yang telah dikembangkan.

BAB 2

LANDASAN TEORI

2.1 Kelapa Sawit

Kelapa sawit merupakan komoditas penting yang mendukung perekonomian Indonesia, namun sangat rentan terhadap berbagai penyakit seperti busuk pangkal batang dan bercak daun yang dapat menurunkan hasil panen [6], Kelapa sawit merupakan jenis tanaman yang berasal dari keluarga Arecaceae. Kelapa sawit awal dibudidayakan di Amerika Selatan. Tanaman ini merupakan salah satu komoditi utama dalam usaha petani dalam memproduksi minyak kelapa sawit. Di dalam industri perkebunan, kelapa sawit ini dikenal sebagai komoditi yang memiliki nilai tinggi sebagai pengganti kelapa yang digunakan dengan tujuan membuat minyak [7], Industri kelapa sawit merupakan sektor penting dalam perekonomian Indonesia. Produksi kelapa sawit memberikan kontribusi signifikan terhadap pendapatan nasional dan penciptaan lapangan Kerja [8].

2.2 Penyakit Daun Sawit

Salah satu penghambat utama bagi tanaman untuk menghasilkan produksi optimal sesuai dengan potensinya adalah keberadaan penyakit pada tanaman kelapa sawit, Penyakit pada daun tanaman menyebabkan penurunan produksi besar dan kerugian ekonomi negara. Di sisi lain, pekebun [9],

1. Bercak Daun

Penyakit bercak daun merupakan penyakit yang disebabkan oleh fungsi yang menghambat dan mengurangi hasil fotosintesis dan selanjutnya akan

menghambat pertumbuhan.[10]. Pada tanaman kelapa dan kelapa sawit, cendawan ini merupakan penyebab penyakit utama yang menyerang pada stadium pembibitan yang sering disebut dengan penyakit bercak daun. Penyakit bercak daun yang disebabkan oleh *Curvularia* sp. di pembibitan kelapa sawit dapat mencapai 38% seperti pada gambar di bawah ini



Gambar 2. 1 bercak daun

2. Karat Daun

Penyakit karat daun (*Cephaleuros vircens*) merupakan penyakit yang sangat merugikan pada tanaman kelapa sawit dapat mempengaruhi produktivitas dan hasil TBS. Dari beberapa struktur jamur tersebut, uredospora paling berperan dalam perkembangan penyakit karat daun. Penyakit karat daun (*cephaleuros vircens*) berwarna orange, panjang 25-35 μm dan lebar 12-28 μm , berbentuk seperti ginjal dan berduri pada bagian yang cembung [12],



Gambar 2. 2 karat daun

2.3 Dataset *Citra*

Dataset adalah sekumpulan data yang dapat digunakan sebagai bahan percobaan riset [13] dataset merupakan salah satu cara dalam penetapan metode klasifikasi yang tepat [14], Kualitas dataset merupakan faktor penting yang mempengaruhi hasil penelitian, dan peneliti harus mempertimbangkan keakuratan, kelengkapan, dan relevansi data saat melakukan pemilihan. Selain itu, aksesibilitas dan ketersediaan dataset terbuka memberikan keuntungan bagi peneliti, meskipun tantangan seperti masalah kualitas, keterbatasan lisensi, dan risiko orisinalitas tetap ada, Dataset merupakan bagian yang sangat penting dalam pembuatan suatu model deep learning [15]

2.4 Klasifikasi *Citra*

Klasifikasi *Citra* merupakan proses untuk pengelompokan sejumlah pixel atau picture element pada sebuah *Citra* menjadi kelas-kelas pada masing-masing kelas mendeskripsikan suatu entitas yang mempunyai karakter agar dapat dikenali

[16]. Klasifikasi *Citra* menjadi tahapan utama setelah segmentasi dari klasifikasi berbasis objek [17]

Segmentasi *Citra* merupakan sebuah langkah awal pada klasifikasi *Citra* dengan metode berbasis objek. Segmentasi *Citra* digunakan untuk mengelompokkan piksel yang memiliki kesamaan struktur, dengan tujuan untuk membuat setiap struktur individual menjadi region atau wilayah individual[17]

2.5 Machine Learning

Machine Learning adalah pembelajaran mesin yang sangat membantu dalam menyelesaikan masalah, membuat mudah dalam mengerjakan sesuatu. Penggunaan *Machine Learning* sudah banyak digunakan dalam dunia industri, pertanian, dunia pendidikan dan lain sebagainya [18].. *Machine Learning* (ML) dapat digunakan untuk mencari dan menemukan pola yang unik dari sekumpulan data. Algoritma *Support Vector Machine* (SVM) dipilih karena algoritma ini mampu mengklasifikasikan nilai ke dalam kelas-kelas tertentu [19]. *Machine Learning* dapat mempelajari pola data historis yang ada, sehingga dengan menggunakan metode *Machine Learning* dapat membantu dalam mengklasifikasikan ataupun mendeteksi apakah akan terjadi banjir atau tidak. Metode yang digunakan dalam penelitian ini yaitu Metode Systematic Literature Review [20].

2.6 Support Vector Machine

Support Vector Machine (SVM) adalah Teknik pembelajaran dengan banyak kualitas yang diinginkan dan menjadikan algoritma SVM sangat populer. SVM

mempunyai dasar teoritis yang kuat dan melakukan klasifikasi lebih akurat daripada kebanyakan algoritma lain di banyak aplikasi [21].

Dalam proses klasifikasi, digunakan algoritma *Support Vector Machine* (SVM) untuk mengenali pola pada data training yang telah diberi label dan kemudian menghasilkan *Machine Learning* model. Model yang dihasilkan selanjutnya digunakan untuk mengenali data tweet baru yang belum diberi label sehingga dihasilkan sebuah prediksi kelas sentimennya apakah itu positif, negatif, atau netral [22].

Terdapat empat kernel yang sering digunakan, antara lain [23]:

1. Liner Kernel

$$K = (X_i, X_j) = (X_i \cdot X_j) \dots\dots\dots (1)$$

2. Polynominal Kernel

$$K = (X_i, X_j) = (X_i \cdot X_j + 1)^p \dots\dots\dots (2)$$

3. RBF Kernel

$$K = (X_i, X_j) = e^{-\gamma(X_i - X_j)^2} \dots\dots\dots (3)$$

4. Sigmoid Kernel

$$K = (X_i, X_j) = \tanh(nX_i \cdot X_j + v) \dots\dots\dots (4)$$

2.7 Confusion Matrix

Confusion Matrix adalah tabel yang menyatakan klasifikasi jumlah data uji yang benar dan jumlah data uji yang salah[24], *Confusion Matrix* merupakan tabel yang membandingkan hasil prediksi model dengan data asli, memberikan

informasi tentang jumlah prediksi yang benar (True Positive dan True Negative) serta kesalahan prediksi (*False Positive* dan *False Negative*)[25]

Pengukuran kinerja menggunakan *Confusion matrix* yaitu dengan menggunakan 4 perhitungan berikut ini [26]:

5. *Accuracy* adalah rasio antara jumlah prediksi yang benar dengan total data yang diuji.

$$Accuracy = \frac{True\ Positive + True\ Negative}{True\ Positive + True\ Negative + False\ Positive + False\ Negative} \dots\dots\dots (5)$$

6. *Precision* adalah perbandingan antara jumlah prediksi positif yang benar dengan total prediksi positif yang dihasilkan.

$$Precision = \frac{True\ Positive}{False\ Positive + True\ Positif} \dots\dots\dots (6)$$

7. *Recall* adalah perbandingan antara jumlah prediksi positif yang benar dengan total data positif yang sebenarnya.

$$Recall = \frac{True\ Positive}{False\ Negative + True\ Positive} \dots\dots\dots (7)$$

8. Selanjutnya, nilai *F1-Score* dihitung sebagai rata-rata yang dibobotkan antara *Precision* dan *Recall*[27][28][29].

$$F1-Score = 2 * \frac{Precision * Recall}{Precision + Recall} \dots\dots\dots (8)$$

2.8 Website

Website adalah halaman informasi yang disediakan melalui jalur internet, sehingga bisa diakses dimanapun selama terkoneksi dengan jaringan internet [30]. *Website* merupakan suatu media informasi yang menawarkan berbagai kemudahan dalam menyajikan informasi kepada masyarakat luas [31].

Website kumpulan halaman dalam domain yang berisi berbagai informasi yang memungkinkan pengguna Internet untuk membaca dan melihatnya melalui mesin pencari. Informasi yang mungkin berisi situs web biasanya mencakup gambar, ilustrasi, video, dan konten tekstual untuk berbagai tujuan[32]

2.9 Google Collab

Google Collab, adalah sebuah IDE untuk pemrograman *Python* dimana pemrosesan akan dilakukan oleh server Google yang memiliki perangkat keras dengan performa yang tinggi[33], *Google Collaboratory*, juga dikenal sebagai *Google Collab*, adalah alat gratis yang berbasis cloud yang digunakan untuk tujuan pendidikan dan penelitian[34].

Google Collab memungkinkan pengguna untuk mengeksekusi kode *Python* dalam lingkungan berbasis cloud tanpa perlu menginstal atau mengatur lingkungan lokal mereka sendiri. *Google Collab* sering digunakan oleh para ilmuwan data, peneliti, dan pengembang untuk melakukan berbagai jenis pekerjaan, termasuk pemrosesan data, pengembangan model kecerdasan buatan, analisis data, dan pelatihan model mesin [35]

2.10 Matlab

Matlab (Matrix Laboratory) adalah perangkat lunak yang dikembangkan oleh *MathWorks, Inc.* sebagai program untuk analisis dan komputasi numerik berbasis matriks. *Software* ini dirancang untuk membantu dalam berbagai tugas matematika, seperti pengolahan aljabar linier, optimasi, aproksimasi, serta analisis numerik dan statistik. *Matlab* juga menyediakan berbagai fungsi *built-in* yang

mempermudah perhitungan yang kompleks, terutama dalam bidang teknik dan sains [36].

Sebagai perangkat lunak yang banyak digunakan di lingkungan akademik dan industri, *Matlab* memungkinkan pengguna untuk melakukan perhitungan matematis, analisis data, pengembangan algoritma, serta simulasi dan pemodelan. Selain itu, *software* ini mendukung visualisasi data dalam bentuk grafis, sehingga sangat membantu dalam eksplorasi dan pemahaman data [37].

Matlab menjadi alat yang umum digunakan di berbagai bidang, seperti matematika dan komputasi, pengembangan algoritma, simulasi dan pembuatan prototipe, serta analisis teknik. Dengan kemampuannya dalam menangani operasi berbasis matriks, *Matlab* sering digunakan di fakultas teknik dan bidang-bidang lain yang memerlukan komputasi tingkat lanjut [38].

2.11 Flowchart

Flowchart , atau yang sering disebut sebagai diagram alir, merupakan representasi grafis dari langkah-langkah atau prosedur dalam suatu sistem. Diagram ini digunakan untuk menggambarkan urutan instruksi yang berurutan dalam suatu proses, sehingga memudahkan analisis dan pemecahan masalah dalam pengembangan sistem [39].

Flowchart terdiri dari berbagai simbol yang mewakili proses tertentu, seperti *input*, *output*, penyimpanan data, serta pengambilan keputusan. Dalam sistem informasi, *Flowchart* dapat dibagi menjadi dua jenis utama, yaitu *Flowchart* sistem, yang menunjukkan alur proses dalam suatu sistem beserta

media *input* dan *output*nya, serta *Flowchart* program, yang lebih rinci dalam menggambarkan urutan instruksi dalam suatu program [40].

Seorang analis sistem menggunakan *Flowchart* sebagai dokumentasi logis untuk menjelaskan rancangan sistem kepada programmer. Dengan visualisasi yang jelas, *Flowchart* membantu dalam memahami alur proses, mengidentifikasi potensi masalah, serta memberikan solusi dalam membangun suatu sistem atau program [41].

2.12 Python

Python adalah bahasa pemrograman interpretatif yang dikembangkan oleh Guido van Rossum di Belanda pada tahun 1990. Nama *Python* diambil dari acara televisi favoritnya, *Monty Python's Flying Circus*. Awalnya dikembangkan sebagai proyek hobi, *Python* kini menjadi salah satu bahasa pemrograman yang banyak digunakan dalam industri dan pendidikan karena sintaksnya yang sederhana, ringkas, dan intuitif, serta didukung oleh pustaka yang luas [42]

Python mendukung berbagai paradigma pemrograman, termasuk pemrograman prosedural seperti C, pemrograman berorientasi objek seperti *Java*, dan pemrograman fungsional seperti *Lisp*. Kombinasi dari paradigma ini membuat *Python* fleksibel dan mudah digunakan dalam berbagai proyek pengembangan perangkat lunak [43]

Sebagai bahasa yang berfokus pada keterbacaan kode, *Python* memungkinkan penulisan program dengan struktur yang jelas dan mudah dipahami. *Python* juga dapat dijalankan di berbagai *platform* seperti *Windows*,

Linux, dan lainnya, menjadikannya salah satu bahasa pemrograman yang sangat populer di kalangan pengembang perangkat lunak [44].

2.13 Visual Studio Code

Visual Studio Code adalah Software yang sangat ringan, namun kuat editor kode sumbernya yang berjalan dari desktop. Muncul dengan built-in dukungan untuk JavaScript, naskah dan Node.js dan memiliki array beragam ekstensi yang tersedia untuk bahasa lain, termasuk *C++*, *C#*, *Python*, dan *PHP* [45].

VS Code hadir dengan dukungan bawaan untuk *JavaScript*, *TypeScript*, dan *Node.js*, serta memiliki ekosistem ekstensi yang kaya untuk berbagai bahasa pemrograman lainnya, termasuk *C++*, *C#*, *Java*, *Python*, *PHP*, dan *Go*. Selain itu, editor ini juga mendukung berbagai runtime seperti *.NET* dan *Unity*, menjadikannya pilihan utama bagi banyak pengembang perangkat lunak [46].

Sebagai perangkat lunak *open-source*, *Visual Studio Code* memungkinkan pengembang untuk melihat dan berkontribusi pada kode sumbernya melalui *Github*. Selain itu, editor ini menawarkan fitur tambahan seperti penyorotan sintaksis, tema warna yang disesuaikan dengan fungsi kode, serta dukungan untuk ekstensi yang memungkinkan pengguna menambahkan fitur baru sesuai kebutuhan [47].

2.14 Google Drive

Google Drive adalah layanan penyimpanan data milik perusahaan *Google*. *Google Drive* menyediakan tempat penyimpanan data gratis sebesar 15 GB untuk

setiap akun *Gmail*. Kuota tersebut dapat digunakan oleh pengguna untuk menyimpan data di dalam *Google Drive* [50]

Google Drive adalah *platform* penyimpanan data berbasis awan *online* yang diperkenalkan pada tanggal 24 April 2012. *Google Drive* tersedia di komputer desktop dan perangkat seluler, sama seperti layanan lain seperti *Dropbox* atau *OneDrive*. *Google Drive*, yang sering dikenal sebagai *GDrive*, sudah terpasang secara default sebagai aplikasi pada beberapa smartphone [51]. *Google Drive* bisa digunakan untuk menyimpan banyak file penting contohnya, yaitu file berbentuk *Word*, *Powerpoint*, *Excel*, *Xps*, *Pdf*, *Rar*, *PNG*, Foto, Video dan file lain. [52]

2.15 Black Box Testing

Black Box Testing adalah metode pengujian perangkat lunak yang berfokus pada pengujian fungsionalitas aplikasi tanpa mempertimbangkan struktur internal atau kode program yang digunakan. Dalam metode ini, penguji tidak memerlukan pengetahuan khusus tentang kode atau desain sistem, melainkan hanya mengevaluasi apakah sistem bekerja sesuai dengan yang diharapkan berdasarkan *input* dan *output* yang dihasilkan [53].

Metode ini digunakan untuk memastikan bahwa setiap modul perangkat lunak berfungsi sebagaimana mestinya tanpa perlu menguji desain atau kode program secara langsung. Meskipun bukan pengganti pengujian *white box*, *Black Box Testing* berperan sebagai metode pelengkap untuk menguji aspek yang tidak dapat dijangkau oleh pengujian *white box*, sehingga meningkatkan keandalan sistem secara keseluruhan [53]

Proses *Black Box Testing* dilakukan dengan mencoba program yang telah dikembangkan melalui penginputan data ke dalam setiap formulir yang tersedia. Hal ini bertujuan untuk memastikan bahwa sistem dapat berjalan sesuai dengan kebutuhan pengguna atau perusahaan serta mampu memberikan hasil yang diharapkan dalam berbagai skenario pengujian [53]

2.16 Penelitian Terkait

Sejumlah penelitian sebelumnya telah menunjukkan efektivitas *SVM* dalam melakukan klasifikasi, penelitian terkait dapat dilihat pada tabel di bawah ini.

Tabel 2. 1 Penelitian Terkait

N o	Penulis dan Tahun	Judul	Metode	Hasil
1	Hendry Cipta Husada1 dan Adi Suryaputra Paramita(20 21)	Analisis Sentimen Pada Maskapai Penerbangan di PlatformTwitter Menggunakan <i>Algoritma Support Vector Machine (SVM)</i>	<i>Algoritma Support Vector Machine (SVM)</i>	(<i>SVM</i>) terbukti mampu diterapkan dengan baik untukmelakukan analisis 26sentiment dengan didukung olehbeberapa metode preprocessing, pembobotan termmenggunakan TF-IDF, dan parameter tuning algoritma.Selain itu, pada proses pengujian algoritma Support VectorMachine (parameter tuning), nilai accuracy terbaikdihasilkan oleh kernel RBF dengan nilai parameter complexity= 10 dan gamma = 1, yaitu sebesar 84,37%. Sedangkan padapengujian menggunakan 10-fold cross validation, diperolehrata-rata nilai accuracy sebesar 80,41%.

				Nilai precision, recall, dan f-score terbaik juga dihasilkan oleh kernel RBF dengan nilai parameter complexity = 10 dan gamma = 1, yang masing-masing sebesar 84,33%, 84,67%, dan 84,50%.
2	Irma Surya Kumala Idris, dkk(2023)	Analisis Sentimen Terhadap Penggunaan Aplikasi Shopee Menggunakan Algoritma Support Vector Machine (SVM)	Algoritma Support Vector Machine (SVM)	Model <i>algoritma Support Vector Machine (SVM)</i> dapat disimpulkan bahwa, SVM terbukti sebagai algoritma klasifikasi yang cukup baik untuk menganalisis sentimen pada pengguna aplikasi Shopee dengan perolehan akurasi sebesar 98% dan f1-score 0.98 atau sebesar 98%.
3	Thalita Meisya Permata Aulia, Nur Arifin Dan Rini Mayasari(2021)	Perbandingan Kernel <i>Support Vector Machine (SVM)</i> Dalam Penerapan Analisis Sentimen Vaksinasi Covid-19	<i>SUPPORT VECTOR MACHINE (SVM)</i>	Algoritma SVM bekerja dengan baik pada klasifikasi teks. Dari percobaan terhadap 4 kernel SVM kernel sigmoid dan linear mendapatkan nilai akurasi lebih tinggi yaitu 0.87, dibandingkan dengan RBF dan polynomial yang hanya 0.86.
4	Yusuf Amrozi, Dian Yulianti, Agung Susilo, Nur Novianto, Rizka Ramadhan(2022)	Klasifikasi Jenis Buah Pisang Berdasarkan Citra Warna dengan Metode SVM <i>Support Vector Machine (SVM)</i> Kombinasi warna, fitur tekstur, dan	Metode SVM <i>Support Vector Machine (SVM)</i>	Akurasi klasifikasi yang signifikan sebesar 89,86%. Semua fitur yang diekstraksi direduksi dan disimpan dalam vektor fitur PCA (Principal Component Analysis). Untuk penelitian selanjutnya disarankan untuk

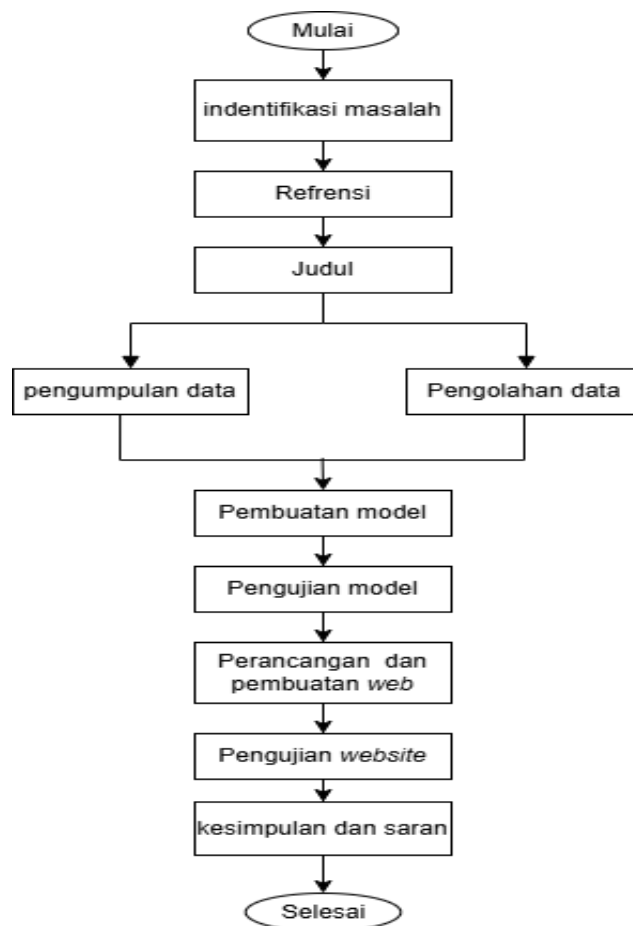
		pengukuran bentuk menggunakan SVM menghasilkan hasil yang akurat.		menggunakan metode dan model warna lain sebagai perbandingan hasil keakuratan sistem
5	Nugraheni, M.(2024)	Identifikasi Jamur Yang Dapat Dikonsumsi Dan Beracun	Klasifikasi Jenis Sampah Organik dan Anorganik Menggunakan Pengolahan Citra dan SVM Ekstraksi Warna + Bentuk + SVM	Sistem mampu membedakan jenis sampah dengan akurasi 86%, membantu sistem pendaur ulangan otomatis.
6	Hidayat, R.(2023)	Klasifikasi Citra Batik Menggunakan SVM dan Ekstraksi Ciri Histogram Warna.	SVM dan Ekstraksi Ciri Histogram Warna.	efektif dalam membedakan motif batik, dengan akurasi mencapai 91%.
7	Lestari, S.(2022)	Deteksi Dini Penyakit Daun Padi Berdasarkan Citra Digital dengan SVM.	Color Histogram + SVM	Berhasil mengklasifikasikan penyakit daun padi dengan akurasi 88%.
8	Firmansyah, A.(2022)	Pengenalan Karakter Tulisan Tangan Bahasa Indonesia Berbasis SVMPCA + SVM	SVMPCA + SVM	SVM menunjukkan performa akurasi yang tinggi dalam klasifikasi karakter tulisan tangan, mencapai 90%.
9	Rahayu, D.(2021)	Klasifikasi Citra Daun Tanaman Menggunakan Ekstraksi Ciri dan SVM Ekstraksi Ciri (GLCM)	SVM Ekstraksi Ciri (GLCM)	SVM SVM mampu mengklasifikasikan jenis daun dengan akurasi sebesar 87% berdasarkan tekstur dan bentuk.
10	Riszki Wijayatun	Analisis Sentimen Pada	Metode Support Vector	hasil analisis dan pengujian terhadap

	Pratiwi, Sharfina Febbi , Dairoh, Dwi Intan Af'idah, Qirani Rifa dan Amaliyah Gian F(2021)	Review SkincareFemale Daily Menggunakan Metode <i>Support Vector Machine</i> (SVM)	<i>Machine</i> (SVM)	analisis sentiment pada review produk kecantikan menggunakan metode <i>Support Vector Machine</i> maka didapatkan hasil kesimpulan bahwa Metode <i>Support Vector Machine</i> dapat melakukan prediksi kelas sentimen pada review produk kecantikan sesuai analisa yang disiapkan. Kemudian analisis sentimen ini belum sepenuhnya relevan untuk memprediksi kelas sentimen yang sesuai terhadap pemberian kelas berdasarkan rating atau bintang pada review tersebut, dan analisis sentiment pada produk kecantikan menggunakan SVM menghasilkan nilai akurasi menggunakan dataset 80% datatraining dan 20% data testing sehingga mendapatkan akurasi sebesar 87% dengan recall sebesar 90%, precisionsebesar 84,90%, dan f1 score sebesar 87,37%.
--	--	---	-------------------------	---

BAB 3

METODOLOGI PENELITIAN

Tahapan penelitian yang akan dilakukan dan penyelesaian masalah terhadap Implementasi algoritma SVM (*Support Vector Machine*) untuk klasifikasi penyakit daun kelapa sawit. Adapun tahapan metodologi yang dilakukan selama penelitian dapat dilihat pada Gambar 3.1 yang mana merupakan proses yang dimulai dari studi literatur hingga diperoleh kesimpulan.



Gambar 3. 1 Tahapan Metodologi Penelitian

3.1 Identifikasi Masalah

Masalah utama yang diangkat dalam penelitian ini adalah kesulitan dalam mendeteksi penyakit daun sawit secara dini hanya berdasarkan pengamatan visual oleh masyarakat umum maupun petani. Gejala penyakit pada daun sawit seringkali memiliki kemiripan satu sama lain, seperti perubahan warna, munculnya bercak, atau kerusakan tekstur permukaan daun, sehingga berpotensi menimbulkan kesalahan dalam identifikasi. Kondisi ini dapat menyebabkan keterlambatan penanganan penyakit yang berdampak pada menurunnya pertumbuhan dan produktivitas tanaman kelapa sawit. Oleh karena itu, dibutuhkan suatu sistem deteksi berbasis komputer yang mampu memproses *Citra* daun sawit dan mengklasifikasikan kondisi daun secara otomatis dan akurat.

3.2 Referensi

Penelitian ini didukung oleh studi literatur terhadap berbagai sumber seperti jurnal, artikel ilmiah, buku, dan dokumentasi terkait penyakit daun kelapa sawit, pengolahan *Citra* digital, serta penerapan algoritma *Machine Learning*. Fokus utama kajian pustaka adalah pemahaman mengenai karakteristik penyakit daun sawit, ciri-ciri visual daun yang sehat dan terserang penyakit, tahapan pengolahan *Citra*, serta efektivitas algoritma *Support Vector Machine (SVM)* dalam klasifikasi berbasis *Citra*. Referensi juga mencakup penelitian terdahulu yang menggunakan pendekatan serupa pada objek tanaman atau *Citra* biologis lainnya.

3.3 Judul

Berdasarkan permasalahan yang ditemukan dan hasil studi pustaka, maka ditetapkan judul penelitian yaitu “Penerapan Metode *SVM* untuk Deteksi Dini Penyakit Daun Sawit Berbasis Pengolahan *Citra*”. Judul ini mencerminkan fokus penelitian pada penggunaan metode *Support Vector Machine* (*SVM*) untuk membangun model klasifikasi berbasis *Citra* daun sawit guna membantu proses deteksi penyakit secara lebih efisien dan akurat.

3.1.1 Pengumpulan Data

Sumber data diperoleh dari dataset *Citra* daun sawit yang tersedia secara daring maupun dari hasil dokumentasi langsung di lapangan. Data yang dikumpulkan terdiri atas gambar-gambar daun sawit yang telah diberi label sesuai dengan kondisi daun, yaitu daun sehat dan daun yang terindikasi terserang penyakit. Pengumpulan data dilakukan dengan mempertimbangkan variasi kondisi pencahayaan, sudut pengambilan gambar, tingkat keparahan gejala penyakit, serta keseimbangan jumlah data pada setiap kelas agar model klasifikasi dapat bekerja secara optimal. Secara rinci, data jamur yang digunakan dalam penelitian ini adalah sebagai berikut:

1. Daun Sawit Sehat, yaitu *Citra* daun sawit yang tidak menunjukkan gejala penyakit, seperti bercak, perubahan warna yang abnormal, atau kerusakan permukaan daun.

2. Daun Sawit Berpenyakit, yaitu *Citra* daun sawit yang menunjukkan gejala serangan penyakit, seperti bercak kuning kecoklatan, perubahan tekstur, atau kerusakan lain pada permukaan daun.

Seluruh data dikumpulkan dalam format gambar seperti *.jpg* atau *.png*, kemudian dilakukan seleksi awal untuk memastikan kualitas gambar cukup baik, tidak buram, dan tidak mengandung noise berlebihan. Data yang telah lolos seleksi selanjutnya digunakan dalam proses pengolahan dan pelatihan model.

3.1.2 Pengolahan Data

Pengolahan data merupakan tahap penting yang dilakukan sebelum data digunakan dalam proses pelatihan dan pengujian model klasifikasi. Langkah-langkah pengolahan data pada penelitian ini meliputi::

3. Seluruh gambar daun sawit yang telah dikumpulkan diubah ukurannya menjadi 224×224 piksel agar memiliki ukuran yang seragam dan sesuai untuk proses ekstraksi fitur.
4. Dataset yang telah diproses kemudian dilakukan pembagian data kedalam *Set train* dan *validation* 80:20.
5. *Citra* yang telah melalui tahap preprocessing selanjutnya diekstraksi fitur-fitur pentingnya untuk menghasilkan representasi numerik dari setiap gambar.
6. Vektor fitur hasil ekstraksi kemudian diklasifikasikan menggunakan algoritma *Support Vector Machine (SVM)* dengan kernel yang sesuai, misalnya *Radial*

Basis Function (RBF), untuk mengukur performa model terhadap data yang belum dikenali sebelumnya.

3.4 Pembuatan Model

Dataset yang digunakan dalam penelitian ini terdiri dari gambar daun sawit yang diklasifikasikan ke dalam dua kategori utama, yaitu daun sehat dan daun yang terserang penyakit. Data diperoleh dari sumber dataset publik maupun hasil pengambilan gambar secara langsung. Setiap kelas terdiri atas sejumlah gambar yang telah melalui tahap seleksi dan pelabelan. Setelah data siap, dilakukan proses pembuatan model klasifikasi yang bertujuan untuk mendeteksi kondisi daun sawit berdasarkan *Citra digital*.



Gambar 3. 2 Flowchart Pembuatan Model

3.4.1 Proses *Input* Gambar

Gambar daun sawit yang akan diklasifikasikan diproses terlebih dahulu. Proses preprocessing ini meliputi resize gambar ke ukuran 224×224 piksel dengan 3 channel warna (RGB). Ukuran ini digunakan agar seluruh *Citra* memiliki format input yang seragam dan mempermudah proses pengolahan pada tahapan selanjutnya.

3.4.2 Eksrtaksi Fitur

Gambar yang telah melalui tahap preprocessing kemudian diproses untuk mendapatkan fitur-fitur penting dari *Citra*. Fitur yang diekstraksi dapat berupa karakteristik warna, tekstur, dan pola permukaan daun yang berkaitan dengan gejala penyakit. Hasil ekstraksi fitur ini merupakan representasi numerik dari *Citra* daun sawit yang akan digunakan sebagai masukan dalam proses klasifikasi.

3.4.3 Vektor Fitur Yang Diekstraksi

Fitur hasil ekstraksi dari setiap *Citra* daun sawit akan dikumpulkan dalam bentuk vektor fitur. Vektor inilah yang menjadi input utama bagi algoritma klasifikasi *SVM*. Dengan menggunakan vektor fitur tersebut, model dapat mempelajari pola perbedaan antara daun sawit sehat dan daun sawit yang berpenyakit.

3.4.4 SVM Klasifikasi

Algoritma Support Vector Machine (SVM) digunakan sebagai model klasifikasi. *SVM* akan mempelajari pola dari data fitur hasil ekstraksi *Citra* daun sawit dan memisahkan antar kelas berdasarkan kondisi daun, yaitu daun sehat dan daun berpenyakit, dengan menggunakan hyperplane terbaik.

Dalam metode SVM, proses klasifikasi dilakukan menggunakan fungsi keputusan sebagai berikut:

beriku

$$f(x) = \sum_{i=1}^n \alpha_i \cdot y_i \cdot k(x_i, x) + b \dots\dots\dots(9)$$

- a. x_i : vektor fitur dari data latih ke-i
- b. x : vektor fitur dari data uji
- c. α_i : Lagrange multiplier dari hasil pelatihan
- d. y_i : label kelas dari data latih
- e. $K(x_i, x)$: kernel function antara data latih dan data uji
- f. b : bias atau intersep dari model

3.4.5 Keluaran Label

Setelah proses klasifikasi selesai, sistem akan memberikan output berupa label kondisi daun, yaitu Sehat, Berpenyakit. Output ini ditampilkan kepada pengguna sebagai hasil dari proses deteksi dini penyakit daun sawit berdasarkan *Citra* yang dimasukkan.

3.5 Pengujian Model

Pada tahapan pengujian model, model yang telah dilatih sebelumnya diuji untuk mengevaluasi kinerjanya. Pengujian ini mencakup:

1. Evaluasi akurasi pada set data validasi.

2. Penggunaan metrik tambahan seperti *Accuracy*, *Precision*, *Recall* dan *F1-Score* bertujuan untuk memberikan gambaran yang lebih lengkap tentang performa model secara keseluruhan.
3. Analisis hasil pengujian untuk mengidentifikasi bagian yang perlu diperbaiki atau dioptimalkan.

Pengukuran kinerja menggunakan *Confusion Matrix* yaitu dengan menggunakan 4 perhitungan berikut ini:

1. *Accuracy* adalah rasio antara jumlah prediksi yang benar dengan total data yang diuji.

$$Accuracy = \frac{true\ positive + true\ negative}{true\ positive + true\ negative + false\ positive + false\ negatif} \dots\dots (10)$$

2. *Precision* adalah perbandingan antara jumlah prediksi positif yang benar dengan total prediksi positif yang dihasilkan.

$$Precision = \frac{true\ positive}{false\ positive + true\ positive} \dots\dots\dots (11)$$

3. *Recall* adalah perbandingan antara jumlah prediksi positif yang benar dengan total data positif yang sebenarnya.

$$Recall = \frac{true\ positive}{false\ negative + true\ positive} \dots\dots\dots (12)$$

4. Selanjutnya, nilai *F1-Score* dihitung sebagai rata-rata yang dibobotkan antara *Precision* dan *Recall*.

$$F1 - Score = 2 * \frac{precision * recall}{precision + recall} \dots\dots\dots (13)$$

Hasil dari tahapan pengujian model ini berupa laporan performa model

berdasarkan metrik evaluasi yang nantinya ketika performa model sudah mencapai akurasi tertinggi, maka model siap diproses lebih lanjut.

3.6 Perancangan dan Pembuatan Website

Pada tahap ini, sistem deteksi dini penyakit daun sawit mulai dirancang dan dikembangkan. Proses ini mencakup berbagai aspek, mulai dari pengumpulan dan preprocessing data hingga integrasi model klasifikasi menggunakan algoritma *Support Vector Machine (SVM)*.

Sistem ini diimplementasikan dalam bentuk *Website* berbasis antarmuka grafis atau *Graphical User Interface (GUI)* yang memungkinkan pengguna mengunggah gambar daun sawit dan memperoleh hasil deteksi secara otomatis. Desain antarmuka dibuat agar mudah digunakan dan informatif, dengan menampilkan hasil klasifikasi kondisi daun yang dimasukkan oleh pengguna.

3.6.1 Perancangan Website

Perancangan *Website* dilakukan dengan mempertimbangkan aspek fungsionalitas, pengalaman pengguna (*user experience*), serta efisiensi sistem dalam mendukung proses deteksi *Citra* daun sawit. *Website* ini bertujuan sebagai antarmuka pengguna yang dapat digunakan untuk mengunggah gambar daun, melakukan proses deteksi, dan menampilkan hasil secara cepat dan informatif. Berikut adalah langkah-langkah utama dalam proses perancangan *Website*:

1. Analisis Kebutuhan

- a) Menentukan fitur utama yang akan tersedia pada *Website*, antara lain: fitur unggah gambar daun sawit, tombol proses deteksi, hasil klasifikasi berupa label sehat atau berpenyakit.
- b) Memastikan *Website* dapat memberikan pengalaman penggunaan yang dapat diakses dengan mudah oleh berbagai jenis perangkat, baik *desktop* maupun *mobile*

2. Perancangan UI/UX

- a) Mendesain tampilan antarmuka menggunakan *wireframe* dan *mockup* untuk memastikan navigasi yang mudah, tata letak yang rapi, dan tampilan hasil klasifikasi yang jelas.
- b) Memilih skema warna yang sesuai dan elemen desain yang sederhana agar pengguna merasa nyaman dalam mengoperasikan sistem. Desain dibuat dengan gaya minimalis namun informatif.

3. Perancangan Arsitektur Sistem

- a) *Frontend*: Antarmuka pengguna dibangun menggunakan Streamlit, yaitu *framework* berbasis *Python* yang memungkinkan pengembangan antarmuka *web* secara cepat dan interaktif. *Streamlit* dipilih karena mendukung pemrosesan gambar secara langsung dan memudahkan integrasi dengan model klasifikasi berbasis *Python*. Penggunaan *Streamlit* juga mempermudah visualisasi hasil klasifikasi dan interaksi pengguna

melalui tombol unggah gambar, tampilan prediksi, serta antarmuka yang responsif dan minimalis.

- b) *Backend*: Menggunakan bahasa pemrograman *Python* dengan dukungan pustaka seperti *TensorFlow/Keras* dan *Scikit-learn*, serta *framework Streamlit* sebagai *backend* interaktif berbasis *web*. *Backend* juga mengelola alur integrasi model klasifikasi serta mengembalikan hasil klasifikasi secara langsung ke antarmuka pengguna melalui *platform Streamlit*

3.6.2 Hasil Pembuatan Website

Setelah tahap perancangan dan pembuatan *Website* selesai, dihasilkan sebuah sistem deteksi penyakit daun sawit berbasis web dengan fitur utama sebagai berikut:

1. Antarmuka pengguna yang intuitif untuk mengunggah gambar daun sawit dan menerima hasil klasifikasi berupa label sehat atau berpenyakit.
2. *Backend* yang mampu memproses input gambar secara otomatis, menghasilkan klasifikasi gambar secara *real-time*.
3. *Website* dapat diakses melalui berbagai perangkat (*desktop* maupun *mobile*) setelah di-*deploy* ke server atau menggunakan *Streamlit Community Cloud*.

Dengan adanya *Website* ini, pengguna dapat melakukan deteksi dini penyakit daun sawit secara praktis, cepat, dan akurat tanpa memerlukan pemahaman teknis yang mendalam terhadap metode *Machine Learning* yang digunakan.

3.7 Pengujian Website

Pengujian sistem ini dilakukan dengan cara menggunakan *Black Box*. Dalam Pengujian *Black Box* ini berfokus pada perangkat lunak untuk mendapatkan hasil yang baik, apabila terjadi *error* atau tidak sesuai dengan tujuan yang diharapkan maka dilakukan penganalisa sistem kembali hingga tidak ditemukan *error*.

3.7.1 Black Box Testing

Black Box Testing merupakan pengujian kualitas perangkat lunak yang berfokus pada fungsionalitas perangkat lunak, *pengujian Black Box Testing* bertujuan untuk menemukan fungsi yang tidak benar kesalahan antarmuka kesalahan pada struktur data, kesalahan performansi, kesalahan inisialisasi dan terminasi, dalam pengujian *Black Box Testing* digunakan alat untuk pengumpulan data yang disebut dengan *user acceptance test*, dokumen ini terdiri deskripsi indikator dari prosedur-prosedur pengujian fungsionalitas dari perangkat lunak.

3.8 Kesimpulan dan Saran

Tahapan terakhir adalah menarik kesimpulan dari hasil penelitian yang diperoleh dalam penerapan metode *Support Vector Machine (SVM)* untuk deteksi dini penyakit daun sawit berbasis pengolahan *Citra*. Pada tahapan ini juga disampaikan saran dari peneliti bagi pembaca atau peneliti selanjutnya untuk melakukan pengembangan lebih lanjut, seperti penambahan jumlah dataset, pengujian dengan metode lain sebagai pembandingan, serta pengembangan sistem agar dapat mendeteksi lebih banyak jenis penyakit daun sawit.