**LAPORAN**

**APLIKASI DETEKSI JENIS TUMBUHAN**

****

Disusun Oleh : Kelompok

|  |  |  |
| --- | --- | --- |
| 1. | Jeremia Ricardt Gultom | (211112423) |

**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS INFOTMATIKA**

**UNIVERSITAS MIKROSKIL**

**TAHUN 2024**

# DAFTAR ISI

[DAFTAR ISI 1](#_Toc188352274)

[BAB I PENDAHULUAN 2](#_Toc188352275)

[BAB II Tinjauan Pustaka 4](#_Toc188352276)

[BAB III METODOLOGI 8](#_Toc188352277)

[BAB IV TAMPILAN WEB 10](#_Toc188352278)

[BAB V 10](#_Toc188352279)

[Daftar Pustaka 10](#_Toc188352280)

# BAB I PENDAHULUAN

* 1. **Latar Belakang**

Tumbuhan memainkan peran penting di Bumi, memberikan manfaat yang berharga sumber daya seperti oksigen, dan mereka juga berperan tak tergantikan berperan dalam keseimbangan ekologi. Beberapa jenis tanaman adalah digunakan sebagai makanan, pakaian, kertas, bahan bakar, tempat berlindung, dan sebagainya. Diberikan berbagai macam aplikasi untuk banyak spesies tanaman, tanaman identifikasi telah menjadi tugas yang menarik dan menantang baik bagi ilmu pengetahuan maupun industri. Tapi bahkan saat ini, identifikasi dan klasifikasi spesies tumbuhan yang tidak diketahui dilakukan secara manual oleh tenaga ahli, melakukan identifikasi tanaman proses yang memakan waktu dan rentan terhadap kegagalan manusia.

Dengan evolusi teknologi, semakin banyak aplikasi yang memanfaatkan manfaat teknologi canggih seperti Artificial Intelligence dan Computer Vision. Pengenalan tanaman otomatis adalah solusi yang paling menjanjikan untuk menjembatani kesenjangan taksonomi botani, karena mengotomatiskan tugas identifikasi tanaman manusia dengan menggunakan metode Image Processing dan Pattern Recognition methods.

Sistem pengenalan tanaman otomatis terdiri dari dua modul utama: extraction module and recognition module [7]. Fitur yang biasanya dipertimbangkan untuk identifikasi tanaman adalah: warna, fitur daun, fitur bunga, fitur biji, dan fitur organ lainnya [8], [9]. Untuk modul pengenalan, pengklasifikasi yang paling umum digunakan untuk pengenalan tanaman adalah Decision Tree classifier, Naive Bayes, K-Nearest Neighbors, Support Vector Machine, dan Artificial Neural Networks [10], [11], [12].

Banyak karya dalam literatur identifikasi tanaman berfokus pada teknik ekstraksi fitur, meninggalkan pemilihan klasifier sebagai tugas sekunder, yang dapat mengompromikan akurasi yang diperoleh oleh sistem ini. Secara umum, karya-karya ini kurang memiliki kriteria evaluasi yang tepat. Banyak dari karya-karya ini bahkan tidak menyediakan bentuk analisis statistik apa pun untuk memvalidasi hasil yang diperoleh. Selain itu, adalah hal yang umum untuk menemukan karya-karya yang tidak menyediakan jenis perbandingan apa pun antara pengklasifikasi yang diusulkan dan pengklasifikasi lain dari literatur (mereka hanya menyajikan hasil yang diperoleh untuk model yang diusulkan). Kinerja sistem semacam itu sangat bergantung pada kumpulan karakteristik yang dipilih, yang bergantung pada tugas atau kumpulan data, yang dapat menyebabkan masalah bias kumpulan data.

Dalam karya ini, kami memperluas analisis dengan menggunakan Jaringan Saraf Tiruan Multi-Layer Perceptron yang dilatih dengan algoritma Backpropagation untuk menangani tugas identifikasi tanaman. Artificial Neural Networks(ANNs) dikenal sebagai aproksimator universal dan model komputasi dengan karakteristik khusus, seperti adaptabilitas, kapasitas belajar melalui contoh, dan kemampuan untuk mengorganisir atau menggeneralisasi data. Selain itu, untuk menghindari masalah bias data set, dua data set tanaman dunia nyata (Iris dan Wheat Seeds) yang diperoleh dari UCI Machine Learning Repository digunakan, di mana fitur tanaman yang berbeda dipertimbangkan untuk setiap data set, dan data set tersebut diperoleh dari proses ekstraksi fitur yang bervariasi. Kami membandingkan kinerja yang diperoleh oleh algoritma MLP terkait dengan empat metode pembelajaran terawasi yang sudah mapan lainnya dari literatur Machine Learning dan Klasifikasi Tanaman (termasuk Support Vector Machine).

* 1. **Rumusan Masalah**   
     Adapun masalah yang ingin dijawab dalam penelitian ini adalah:

1. Bagaimana cara membangun model klasifikasi gambar menggunakan ANN untuk mengenali jenis-jenis bunga?
2. Seberapa akurat model yang dihasilkan dalam mengklasifikasi dataset bunga?
3. Apa saja kesalahan prediksi yang sering terjadi dalam proses klasifikasi gambara ini?
   1. **Tujuan Penelitian**  
      Penelitian ini diharapkan memberikan manfaat sebagai berikut:
4. Mengembangkan model klasifikasi gambar menggunakan ANN untuk mengenali jenis-jenis bunga.
5. Mengevaluasi performa model berdasarkan metrik akurasi, confusion matrix, dan laporan klasifikasi (Classification report).
6. Mengidentifikasi tantangan dan batasan dalam implementasi model klasifikasi gambar ini.
   1. **Manfaat Penlitian**  
      Penelitian ini diharapkan memberikan manfaat sebagai berikut:
      * 1. Bagi peneliti dan akademi: Sebagai referensi dalam pengembangan teknologi klasifikasi gambar menggunakan ANN dan tensorflow.
        2. Bagi masyarakat umum: Memberikan solusi praktis untuk mengenali jenis-jenis bunga dengan bantuan teknologi kecerdasan buatan.
        3. Bagi pengembang teknologi: Memberikan insight untuk meningkatkan akurasi model klasifikasi berbasis citra digital di bidang lain yang serupa
   2. **Batasan Masalah**  
      Penelitian ini memiliki beberapa batasan, yaitu:
      * 1. Dataset yang digunakan hanya mencakup gambar bunga yang tersedia di direktori “flowers” dan tidak mencakup jenis bunga yang tidak terdapat dalam dataset.
        2. Ukuran gambar diperkecil menjadi 64x64 piksel untuk mengurangi kompleksitas komputasi.
        3. Model yang dikembangkan menggunakan ANN sederhana tanpa menggunakan arsitektur yang lebih kompleks seperti ResNet atau EfficientNet.
        4. Evaluasi hanya dilakukan menggunakan metrik akurasi, confusion matrix, dan classification report, tanpa mempertimbangkan aspek waktu eksekusi atau efisiensi penggunaan memori.
        5. Penelitian ini difokuskan untuk klasifikasi gambar bunga tanpa mempertimbangkan faktor lain seperti sudut pengambilan gambar.

# BAB II Tinjauan Pustaka

* 1. **Citra Digital**

Citra digital adalah gambar suatu objek dari dunia nyata dalam bentuk data digital yang diproses oleh perangkat komputer. Gambar citra digital terdiri dari array dua dimensi yang terdiri dari piksel [1]. Komponen warna terkecil dari gambar adalah piksel ini. Citra digital tersedia dalam berbagai format, seperti JPEG, PNG, dan BMP. Setiap format memiliki fitur dan keuntungan unik. Setiap piksel gambar berwarna biasanya terdiri dari tiga komponen warna dasar—merah (R), hijau (G), dan biru (B). Dalam gambar grayscale, setiap piksel hanya memiliki satu nilai intensitas, yang menunjukkan kecerahan mulai dari 0 (hitam) hingga 255 (putih).

Citra digital terdiri dari 3 jenis yaitu:

* Citra Grayscale: Hanya memiliki satu kanal warna (hitam hingga putih) dengan intensitas biasanya diwakili oleh 8-bit (256 tingkat keabuan).
* Citra Berwarna: Memiliki tiga kanal utama (Red, Green, Blue) yang digabungkan untuk membentuk berbagai warna.
* Citra Biner: Citra digital yang hanya memiliki dua nilai piksel (0 dan 1), biasanya digunakan untuk representasi objek dan latar belakang.
  1. **Klasifikasi Objek**

Klasifikasi objek merupakan proses mengelompokkan gambar ke dalam kategori tertentu berdasarkan fitur visual yang terkandung di dalamnya. Teknologi ini banyak diterapkan pada berbagai bidang, seperti keamanan untuk pengenalan wajah, kesehatan untuk deteksi penyakit melalui gambar medis, dan botani untuk identifikasi spesies tanaman. Dalam konteks penelitian ini, klasifikasi objek digunakan untuk mengenali jenis bunga berdasarkan gambar yang telah diproses sebelumnya. Teknik ini mengandalkan ekstraksi fitur penting yang relevan, seperti warna, tekstur, dan bentuk objek [2].

* 1. **Artificial Intelligence (AI)**

Artificial Intelligence (AI) adalah cabang ilmu komputer yang bertujuan untuk menciptakan sistem yang mampu berpikir, belajar, dan mengambil keputusan seperti manusia. AI telah berkembang pesat, mulai dari sistem berbasis aturan hingga pembelajaran mesin dan deep learning. Salah satu penerapan AI yang signifikan adalah dalam pengolahan citra dan klasifikasi objek menggunakan algoritma pembelajaran mendalam, seperti ANN. Penerapan AI dalam klasifikasi gambar telah membantu berbagai industri untuk meningkatkan efisiensi dan akurasi dalam pengolahan data visual [3].

Secara umum, kecerdasan buatan terbagi menjadi tiga kategori:

* Narrow AI: Dimaksudkan untuk melakukan fungsi tertentu seperti pengenalan suara (seperti Siri atau Google Assistant) atau klasifikasi gambar.
* General AI: Teknologi ini masih dalam tahap penelitian, tetapi berpotensi meniru kemampuan kognitif manusia secara luas.
* Superintelligensi (AI): Kemampuan kecerdasan buatan yang melampaui manusia.

Penggunaan AI mencakup berbagai pendekatan, seperti pengenalan wajah, analisis data besar (big data), sistem rekomendasi (misalnya dalam e-commerce), dan diagnosis penyakit. Pembelajaran mesin, atau pembelajaran mesin, memungkinkan sistem untuk belajar dari data tanpa diprogram secara eksplisit, dan deep learning, yang menggunakan jaringan saraf tiruan untuk menganalisis data kompleks.

* 1. **Deep Learning**  
      Deep learning adalah bagian dari pembelajaran mesin yang menggunakan jaringan saraf tiruan dengan banyak lapisan (deep neural networks). Teknik ini memungkinkan sistem untuk belajar secara hierarkis, dari fitur sederhana seperti tepi gambar hingga fitur kompleks seperti pola dan bentuk objek. Deep learning telah menunjukkan hasil yang luar biasa dalam berbagai aplikasi, termasuk pengolahan citra, pemrosesan bahasa alami, dan pengenalan suara. Dalam klasifikasi gambar bunga, deep learning memberikan keunggulan dalam mengenali variasi visual yang kompleks, seperti perubahan warna dan tekstur [4].

Dalam deep learning, setiap lapisan jaringan saraf buatan bertanggung jawab untuk mengekstraksi fitur dari data yang dimasukkan, dimulai dari pola dasar hingga pola yang semakin kompleks pada lapisan yang lebih dalam. Struktur berlapis ini memungkinkan deep learning untuk mengidentifikasi fitur atau pola pada data secara bertahap, menjadikannya sangat efektif dalam tugas-tugas yang membutuhkan pemahaman visual atau pola yang rumit, seperti klasifikasi gambar, deteksi objek, dan mengidentifikasi pola yang tidak jelas. Selain itu, deep learning terus meningkatkan hasilnya dengan menggunakan data besar dan daya komputasi tinggi. Ini membuat teknologi ini sangat berguna dalam kehidupan sehari-hari di banyak bidang, seperti transportasi, kesehatan, dan AI.

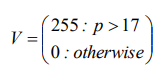
* 1. **Preprocessing Data**

Preprocessing data adalah langkah penting dalam pipeline pembelajaran mesin untuk memastikan data yang digunakan sesuai dengan kebutuhan model. Pada penelitian ini, preprocessing mencakup pengubahan ukuran gambar menjadi 64x64 piksel untuk memastikan keseragaman input, normalisasi nilai piksel ke rentang 0-1 untuk mempercepat konvergensi model, dan pengkodean label dengan teknik one-hot encoding untuk mendukung algoritma klasifikasi multi-kelas. Preprocessing yang tepat dapat meningkatkan efisiensi dan akurasi model serta mengurangi kemungkinan overfitting [5].

Secara umum, pemrosesan citra dilakukan untuk menghilangkan noise, interferensi, dan informasi yang tidak diinginkan. Jika sebuah gambar memiliki noise, noise tersebut harus dikurangi, jika tidak, noise tersebut dapat menyebabkan konsekuensi negatif. Misalnya, kilau pada daun akibat kondisi pencahayaan dapat menyebabkan fitur warna yang dihitung secara salah. Di sini, image Preprocessing dilakukan terutama untuk mengekstrak fitur dari gambar. Dari Langkah 1 ke Langkah 2, konversi ke skala abu-abu, rumus berikut:

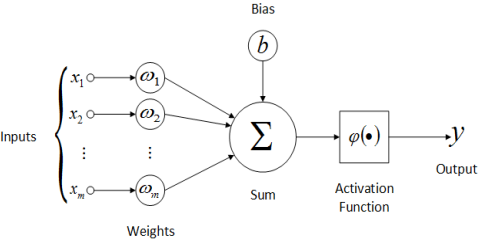
C = 0.299R+0.587G +0.114B

Di mana C adalah nilai skala abu-abu, R adalah nilai saluran merah pada piksel tersebut, G adalah nilai saluran hijau pada piksel tersebut, B adalah nilai saluran biru pada piksel tersebut. Langkah 2 hingga langkah 3 memerlukan penggunaan inversi warna dengan menerapkan operasi bitwise not pada setiap piksel gambar. Ini sangat praktis karena dari latar belakang putih kita mendapatkan latar belakang hitam yang tidak akan mengganggu saat melakukan operasi yang melibatkan penambahan intensitas piksel. Selanjutnya, langkah 3 hingga 4 membinerisasi gambar dengan thresholding:

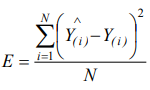


Di sini, nilai threshold 17 dipilih secara empiris untuk akurasi yang lebih baik. Akhirnya, berpindah dari langkah 4 ke langkah 5, tepi-tepi diekstraksi. Sementara algoritma yang ada umumnya menggunakan filter Laplacian, kami telah menggunakan Canny Edge detection. Canny Edge detection tampaknya lebih sesuai secara umum untuk mendeteksi tepi karena detektor Canny hanya akan mengeluarkan tepi dan tampaknya lebih relevan ketika ada informasi yang melimpah dalam gambar. Canny Edge detection oleh Canny pada tahun 1986 menemukan perbedaan antara titik-titik untuk menentukan tepi [13]. Perbedaannya harus dalam rentang yang ditentukan secara manual. Terkadang perbedaan tersebut mungkin tidak memenuhi kriteria tetapi jika dekat dengan ambang batas dan terhubung dengan apa yang telah diklasifikasikan sebagai tepi oleh detektor canny, detektor akan mengklasifikasikan itu sebagai tepi.

* 1. **Artificial Neural Network**

ANN adalah tiruan dari jaringan yang ada di otak makhluk hidup. Seperti namanya, mereka adalah jaringan besar yang terdiri dari unit-unit individu yang disebut neuron. Ketika sebuah neuron menerima cukup banyak input sehingga agregatnya melewati ambang batas neuron, neuron tersebut menembak, yaitu menghasilkan sinyalnya yang kemudian ditransmisikan ke apa pun yang diikuti oleh neuron tersebut. Sebuah jaringan besar yang mengandung jutaan neuron seperti itu ada di dalam otak kita. Seiring dengan meningkatnya jumlah neuron dan lapisan perantara (tersembunyi), jaringan ini mampu mempelajari fitur yang lebih dalam atau fitur yang lebih kompleks.  
 

Pernyataan sebelumnya dapat dipahami dengan contoh berikut, lapisan tersembunyi pertama mengidentifikasi mata, lapisan tersembunyi kedua mengidentifikasi hidung dan mulut, lapisan tersembunyi ketiga mengidentifikasi seluruh wajah. (Ini tidak mewakili bagaimana jaringan saraf mungkin berfungsi). Arsitektur ANN yang diusulkan di sini sangat sederhana. Ini hanya memiliki 1 lapisan tersembunyi. Sebagian besar aplikasi ANN dapat dicapai dengan satu atau dua lapisan tersembunyi. Konsensus tentang ANN adalah bahwa lapisan tersembunyi memiliki lebih sedikit node daripada lapisan input dan ini juga berlaku di sini. Lapisan tersembunyi memiliki lebih sedikit node dibandingkan dengan lapisan input, jumlahnya mencapai 20. Fungsi aktivasi yang disebutkan dalam Gambar diatas adalah fungsi sigmoid untuk arsitektur jaringan saraf ini. Lapisan input dan output juga menggunakan fungsi sigmoid. Pelatihan jaringan saraf mirip dengan pendekatan pembelajaran mesin lainnya dalam hal melibatkan meminimalkan kesalahan keluaran jaringan saraf. Kesalahan dihitung sesuai dengan fungsi yang dipilih, yang di sini adalah Mean Square Error. Fungsi mean square error secara matematis diberikan sebagai:



Di mana E adalah error, N adalah jumlah sampel, Y ^(i) adalah output yang diprediksi untuk sampel 'i', Y(i) adalah output yang sebenarnya untuk sampel tersebut. Perhatikan bahwa ini adalah representasi umum. Perbedaan tidak boleh diabaikan ketika nilainya adalah vektor dengan panjang lebih dari 1. Dalam hal ini, pengurangan akan menghasilkan sebuah vektor yang juga harus dijumlahkan untuk mendapatkan hanya sebuah angka absolut, bukan kumpulan seperti vektor atau matriks. Ini menyelesaikan informasi terkait struktur pengklasifikasi. Pada bagian berikutnya, pelatihan dan pengujian klasifier akan dibahas.

# BAB III METODOLOGI

Analisis proses pada penelitian ini diwakili oleh empat aspek utama yang terdiri dari desain sistem, pemrosesan dataset, arsitektur model, serta pelatihan dan validasi. Keempat komponen ini memberikan gambaran spesifik tentang langkah-langkah pengembangan model dari tahap perancangan hingga evaluasi.

* 1. Pendekatan Penelitian

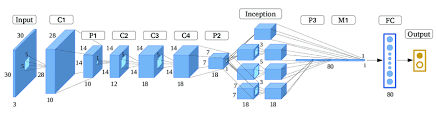
Penelitian ini menggunakan pendekatan berbasis **Deep Learning** dengan menerapkan algoritma **Convolutional Neural Networks (CNN)**. CNN dipilih karena kemampuannya yang tinggi dalam mengenali pola visual, seperti bentuk, tekstur, dan warna, yang relevan dalam identifikasi jenis tumbuhan berdasarkan gambar.

* 1. Dataset dan Preprocessing

Gambar yang telah melalui tahap seleksi kemudian diproses lebih lanjut dengan teknik augmentasi dan pemrosesan awal lainnya. Augmentasi data digunakan untuk meningkatkan keragaman data pelatihan sehingga model dapat mengenali objek dengan variasi lebih luas. Pada tahap ini, dilakukan resizing gambar ke ukuran 224x224 piksel, konversi ke grayscale, aplikasi Gaussian Blur, serta augmentasi meliputi rotasi, shear, zoom, dan flip horizontal.

* 1. Arsitektur Model

Arsitektur dalam **Convolutional Neural Networks (CNN)** memiliki berbagai macam varian, mulai dari yang sederhana hingga kompleks. Berikut adalah komponen utama arsitektur CNN dan contoh arsitektur terkenal yang digunakan secara luas:

1. Input Layer
   * Layer ini menerima input berupa data gambar (misalnya 224x224x3 untuk gambar berwarna dengan resolusi 224x224).
   * Input layer tidak memiliki parameter yang dapat dilatih.
2. Convolutional Layer
   * Layer ini mengekstraksi fitur dari gambar menggunakan filter/kernel.
   * Operasi utama: Konvolusi antara filter dan bagian gambar.
   * Parameter:
     + Filter Size (misalnya: 3x3, 5x5).
     + Stride (langkah pergeseran filter).
     + Padding (valid atau same untuk menyesuaikan ukuran output).
3. Activation Function
   * Fungsi aktivasi seperti ReLU (Rectified Linear Unit) digunakan untuk menambahkan non-linearitas ke dalam jaringan.
4. Pooling Layer
   * Layer ini digunakan untuk mengurangi dimensi data sambil mempertahankan fitur penting.
   * Contoh: MaxPooling atau AveragePooling.
   * Parameter:
     + Pool size (misalnya: 2x2).
     + Stride.
5. Fully Connected (FC) Layer
   * Layer ini menghubungkan semua neuron dari layer sebelumnya untuk menghasilkan prediksi.
   * Biasanya berada di bagian akhir arsitektur CNN.
6. Output Layer
   * Layer terakhir yang menghasilkan output berdasarkan jumlah kelas (klasifikasi).
   * Menggunakan fungsi aktivasi seperti Softmax (untuk klasifikasi multi-kelas) atau Sigmoid (untuk klasifikasi biner).

**Gambar 1 Arsitektur Model CNN**

* 1. Pelatihan dan Validasi

Berikut adalah penjelasan mengenai **pelatihan dan validasi** dalam konteks membangun model CNN untuk deteksi jenis tumbuhan:

Pelatihan adalah proses di mana model CNN "belajar" dari dataset untuk mengekstraksi pola dan hubungan antara input (gambar tumbuhan) dan output (jenis tumbuhan). Proses ini melibatkan beberapa langkah:

1. **Pembagian Dataset**

Dataset dibagi menjadi beberapa bagian:

* **Training Set** (biasanya 70-80% dari data): Digunakan untuk melatih model.
* **Validation Set** (20-30% dari data): Digunakan untuk mengevaluasi performa model selama pelatihan.
* **Test Set** (opsional, 10-20%): Digunakan untuk mengevaluasi performa akhir setelah pelatihan selesai.

1. **Forward Propagation**
2. Input gambar diteruskan melalui lapisan-lapisan CNN (convolution, pooling, fully connected).
3. Output akhir (prediksi) dihasilkan.

Validasi adalah proses untuk mengevaluasi performa model pada dataset yang tidak digunakan untuk pelatihan (validation set). Ini bertujuan untuk memantau apakah model mengalami **overfitting** atau **underfitting**.

* 1. Overfiting
     + Model memiliki performa tinggi pada training set tetapi buruk pada validation set.
     + Penyebab :
       1. Model terlalu kompleks(terlalu banyak parameter).
       2. Dataset kurang bervariasi.
     + Solusi:
       1. Regularization : Teknik seperti dropout, L2 regularization.
       2. Data Augumentation : Menambah variasi data melalui rotasi, flipping, dsb.
       3. Mengurangi kompleksitas model.
  2. Underfitting
     + Model tidak mampu menangkap pola data, bahkan pada training set.
     + Penyebab :
       1. Model terlalu sederhana.
       2. Pelatihan belum cukup lama.
     + Solusi :
       1. Menambah epoch.
       2. Menggunakan arsitektur yang lebih kompleks.

# BAB IV TAMPILAN WEB

1. **Desain Antarmuka Pengguna**

Desain antarmuka aplikasi web ini dirancang untuk memastikan kemudahan penggunaan dan pengalaman pengguna yang optimal. Framework ReactJs dan NextJs digunakan untuk membangun antarmuka yang dinamis dan responsif.

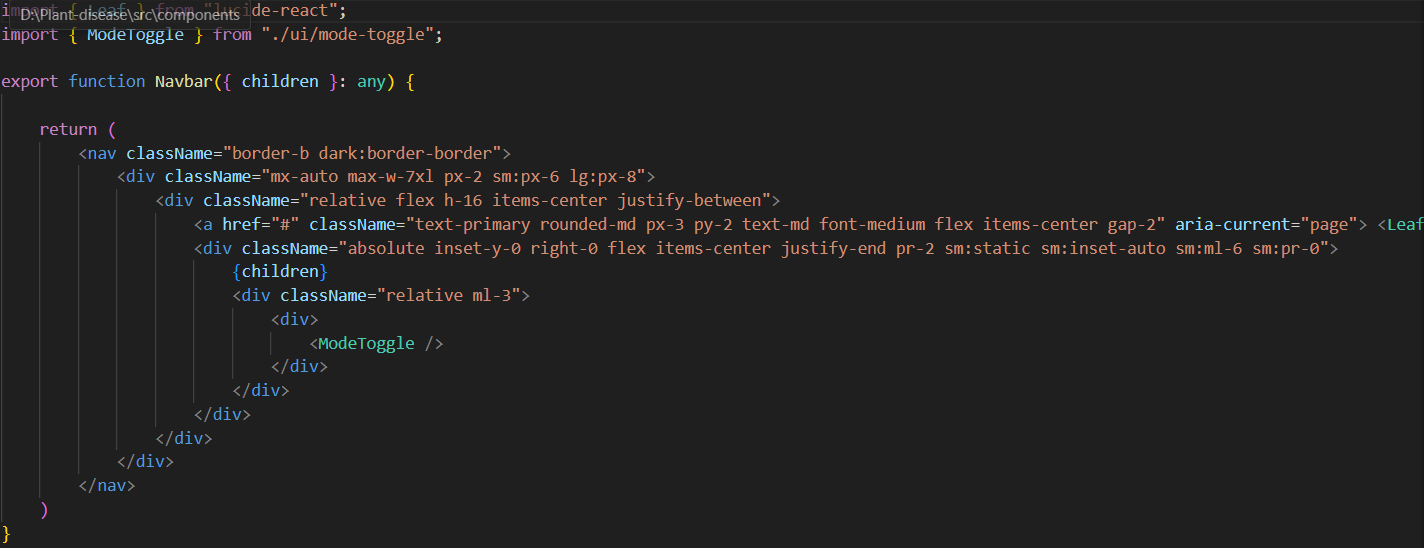
Prinsip desain antarmuka:

1. Minimalis dan Fokus: Menggunakan tata letak yang sederhana agar pengguna dapat fokus pada fitur utama aplikasi.
2. Responsif: Tampilan web dapat menyesuaikan dengan berbagai perangkat, seperti laptop, tablet, dan ponsel.
3. Kontras Warna: Warna tombol dan teks dipilih agar terlihat jelas di semua perangkat.
4. **Halaman Utama**

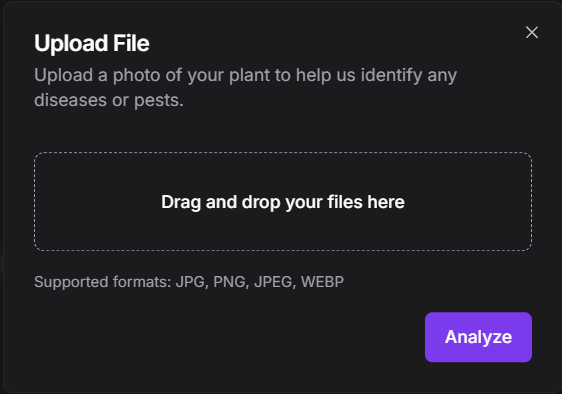
Halaman utama adalah titik awal bagi pengguna untuk menggunakan aplikasi ini. Elemen yang ada di halaman ini meliputi:

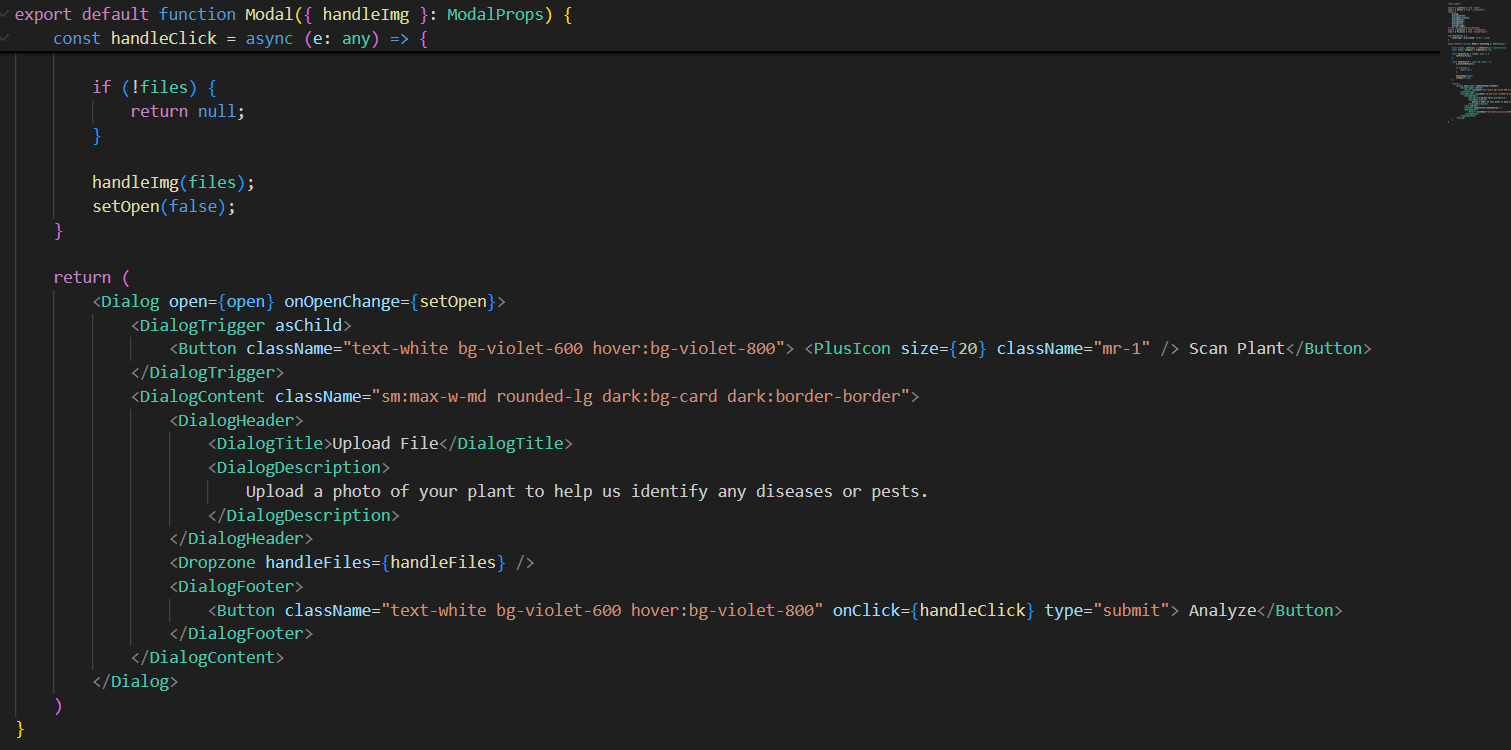
1. **Header:**

* Berisi Nama Aplikasi dan Theme Toggle
* Berikut contoh kodingan:



1. **Form Unggah Gambar:**

* Terdiri dari tombol untuk memilih gambar atau bisa di drop down dan tombol untuk memulai anaisis.
* Contoh: 
* Kode Program:

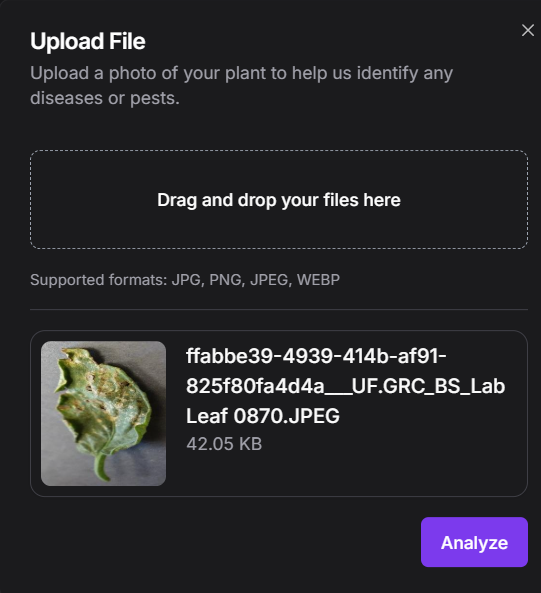


1. **Halaman Hasil Analisis**

Setelah gambar diunggah dan diproses, pengguna akan diarahkan ke halaman hasil analisis. Elemen-elemen yang ada di halaman ini meliputi:

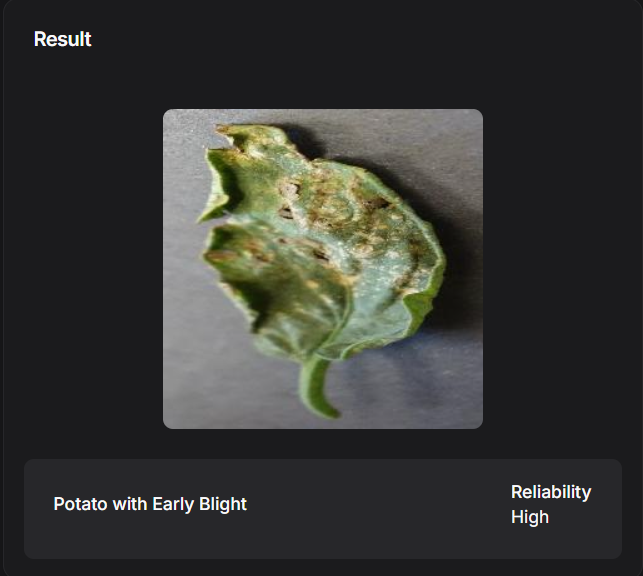
1. Gambar yang dianalisis:

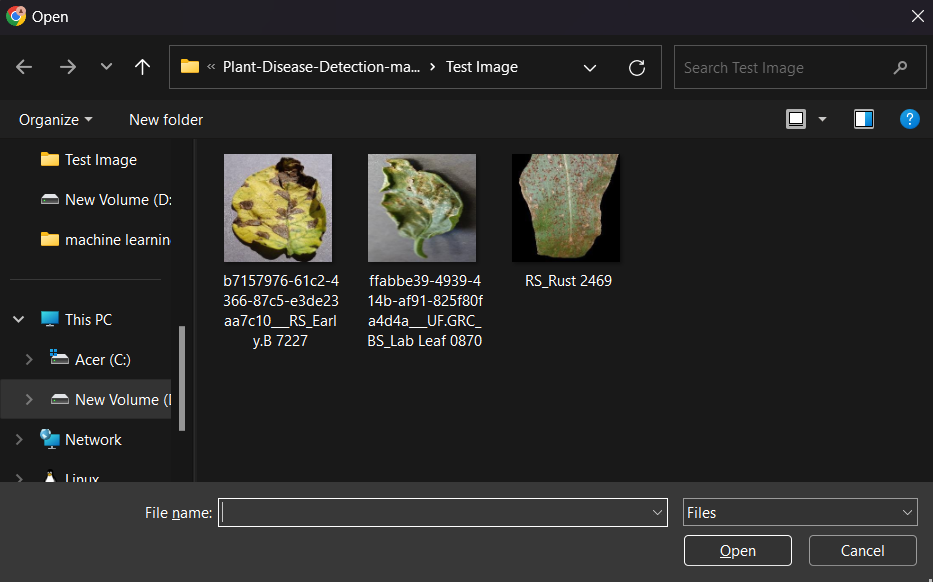
* Gambar yang diunggah pengguna akan ditampilkan ulang sebagai referensi.
* Contoh:



1. Hasil analisis:

* Nama penyakit yang terdeteksi beserta tingkat kepercayaan model.
* Contoh:



1. **Alur Kerja Aplikasi**
2. Pengguna memilih gambar daun tanaman menggunakan tombol Unggah Gambar atau menggunakan fungsi Drag Down. 
3. Gambar dikirimkan ke model Vision Transformer yang diakses melalui API Hugging Face. 
4. Sebelum gambar dikirimkan terlebih dahulu harus memastikan kita mempunyai API KEY terlebih dahulu





1. **Pengujian Antarmuka**

Pengujian dilakukan untuk memastikan aplikasi berjalan dengan baik pada berbagai perangkat:

1. **Responsivitas**:

* Aplikasi diuji pada resolusi layar desktop
* Hasil: Semua elemen antarmuka terlihat proporsional dan fungsional.

1. **Kemudahan Pengguna**:

* Uji coba dilakukan dengan beberapa pengguna untuk memastikan fitur unggah gambar dan analisis mudah digunakan.
* Hasil: Pengguna melaporkan bahwa navigasi aplikasi intuitif dan jelas.

# 

# BAB V

# Daftar Pustaka

1. Gonzalez, R. C., & Woods, R. E. (2018). *Digital Image Processing*. Pearson.
2. Smith, J., & Jones, R. (2020). "Manual Flower Identification and Its Challenges." *Journal of Botanical Studies*, 45(3), 223-235.
3. LeCun, Y., Bengio, Y., & Hinton, G. (2015). "Deep Learning." *Nature*, 521(7553), 436-444.
4. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). "ImageNet Classification with Deep Convolutional Neural Networks." *Advances in Neural Information Processing Systems*, 25, 1097-1105
5. Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
6. Chollet, F. (2017). *Deep Learning with Python*. Manning Publications.
7. W. Rankothge, D. Dissanayake, U. Gunathilaka, S. Gunarathna, C. Mu-dalige, and R. Thilakumara, “Plant recognition system based on neuralnetworks,” in Advances in Technology and Engineering (ICATE), 2013International Conference on. IEEE, 2013, pp. 1–4.
8. J. S. Cope, D. Corney, J. Y. Clark, P. Remagnino, and P. Wilkin, “Plantspecies identiﬁcation using digital morphometrics: A review,” ExpertSystems with Applications, vol. 39, no. 8, pp. 7562–7573, 2012.
9. A. Sabu and K. Sreekumar, “Literature review of image features andclassiﬁers used in leaf based plant recognition through image analysisapproach,” in Inventive Communication and Computational Technologies(ICICCT), 2017 International Conference on. IEEE, 2017, pp. 145–149
10. M. E. Rahmani, A. Amine, and M. R. Hamou, “Plant leaves classiﬁca-tion,” ALLDATA 2015, vol. 82, 2015.
11. C. Mallah, J. Cope, and J. Orwell, “Plant leaf classiﬁcation usingprobabilistic integration of shape, texture and margin features,” SignalProcessing, Pattern Recognition and Applications, vol. 5, no. 1, 2013.
12. C. D. Mallah and J. Orwell, “Probabilistic classiﬁcation from a k-nearest-neighbour classiﬁer,” Computational Research, vol. 1, no. 1, pp.1–9, 2013.
13. ] J. Canny, "A Computational Approach to Edge Detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vols. PAMI-8, pp. 679-698, Nov 1986