

SISTEM PENDETEKSI PENYAKIT PADA DAUN TANAMAN SINGKONG MENGGUNAKAN *DEEP LEARNING* DAN *TENSORFLOW* BERBASIS *ANDROID*

DISEASE DETECTION SYSTEM IN CASSAVA LEAVES USING ANDROID-BASED DEEP LEARNING AND TENSORFLOW

Mirza Faturrachman¹, Indra Yustiana², Somantri³

¹Fakultas Teknik Komputer Dan Desain, Program Studi Teknik Informatika
Universitas Nusa Putra Sukabumi
Email: mirza.faturrachmanti18@nusaputra.ac.id

Abstrak

Tanaman singkong merupakan produksi hasil pertanian terbesar kedua di Indonesia, namun sebagaimana mestinya tanaman singkong tidak luput dari serangan hama dan penyakit. Untuk mengidentifikasi penyakit pada tanaman singkong, paling mudah adalah dengan melihat gejala yang muncul pada permukaan daunnya. Seiring berkembangnya bidang teknologi informasi, terdapat ilmu yang bisa membantu proses pendeteksian penyakit pada daun tanaman singkong yaitu dengan memanfaatkan pengolahan citra atau disebut juga *Image Processing*. Untuk itu, penulis mengambil penelitian tentang pendeteksian penyakit pada daun tanaman singkong dengan menggunakan terapan ilmu *Deep Learning* dengan metode *Convolutional Neural Network (CNN)* dan dengan bantuan *Framework Tensorflow* yang berbasis *Android*. Metode pengambilan data pada penelitian ini menggunakan metode kualitatif, dan metode pengembangan sistem yang digunakan adalah *Prototyping*, dimana metode ini adalah metode yang cepat dalam proses pengembangannya. Metode *Convolutional Neural Network (CNN)* adalah salah satu metode yang memiliki hasil bagus dalam pengolahan citra. Penelitian yang dilakukan ini menggunakan dataset berjumlah 21,367 data gambar daun singkong dari situs *Kaggle*. Penelitian ini mendapat hasil pengujian dengan tingkat akurasi mencapai 86%. Akurasi pengklasifikasian gambar tersebut bergantung pada pencahayaan.

Kata Kunci: Singkong, *Deep Learning*, *Convolutional Neural Network*

Abstract

The cassava plant is the 2nd largest agricultural production in Indonesia. But as it should be, cassava plants are not free from pests and disease. To identify disease in a plant, the easiest way is to look at the symptoms that appear on the leaves' surfaces. Along with the increasing development of information and

technology, there is knowledge that can help the process of detecting disease in cassava leaves by utilizing Image Processing. For this reason, the authors took research on disease detection in cassava leaves using the knowledge of Deep Learning science with the Convolutional Neural Network (CNN) method and Tensorflow Framework. The data collection method in this study uses qualitative methods, and the system development method used in this system is Prototyping, where this method is fast in the development process. The Convolutional Neural Network (CNN) method is one method that has good results in Image Processing. This research used a dataset of 21,367 cassava leaf Image data from the Kaggle site. This study got test results with an accuracy rate of 86%. The accuracy of the Image classification depends on the lighting.

Keywords: Cassava, Deep Learning, Convolutional Neural Network

PENDAHULUAN

Singkong merupakan salah satu tanaman yang banyak ditemukan di Indonesia dan merupakan tanaman dengan banyak manfaat ^[1]. Di Indonesia, singkong merupakan produksi hasil pertanian pangan ke dua terbesar setelah padi dengan menyumbang lebih dari 18 juta produksi setiap tahunnya (Monavia Ayu Rizaty, 2020)

Sebagaimana tanaman lainnya, tanaman singkong juga tidak luput dari serangan hama dan penyakit tanaman. Untuk mengidentifikasi penyakit tanaman singkong bisa dilihat dari gejala-gejala dan munculnya perubahan warna pada daun^[2]. Daun yang diserang oleh penyakit akan mempengaruhi hasil dari tanaman singkong karena daun merupakan bagian vital dari tanaman yang berfungsi sebagai tempat berlangsungnya proses fotosintesis.

Di bidang informatika memanfaatkan *Image Processing* atau

disebut juga pengolahan citra digital dalam melakukan identifikasi dapat membantu para pengelola pertanian untuk memberikan penanganan secara efektif dan efisien pada tanaman yang tidak sehat atau tidak normal^[3].

Image Processing ini mengimplementasikan teknologi *Deep Learning* yang merupakan bagian dari disiplin ilmu *Machine Learning*. Metode *Deep Learning* yang biasa digunakan dalam pengolahan citra digital adalah metode *Convolutional Neural Network (CNN)*. *CNN* merupakan pengembangan dari *Multi Layer Perceptron (MLP)* dan merupakan salah satu algoritma dari *Deep Learning*^[4].

Maka dari itu pada penelitian kali ini, penulis melakukan penelitian yang berjudul “SISTEM PENDETEKSI PENYAKIT PADA DAUN TANAMAN SINGKONG MENGGUNAKAN *DEEP*

LEARNING DAN TENSORFLOW BERBASIS ANDROID", pembuatan aplikasi ini menggunakan teknologi *Deep Learning* dan metode *Convolutional Neural Network (CNN)* dengan bantuan *Framework Tensorflow* dan *Library Package Keras*, lalu untuk menjadikannya hasil yang bisa dikembangkan di perangkat *smartphone* yaitu menggunakan bantuan *Framework Tensorflow Lite* dan dengan menggunakan *software Android Studio* sebagai *Integrated Development Environment (IDE)* dan juga dikembangkan menggunakan bahasa pemrograman *Kotlin*.

LANDASAN TEORI

Deep Learning

Deep Learning adalah merupakan suatu pembelajaran mesin (*machine learning*) yang terdiri dari algoritma pemodelan abstraksi tingkat tinggi pada data. Algoritma pada *Deep Learning* dapat digunakan untuk kebutuhan pembelajaran terarah (*supervised learning*), pembelajaran tak terarah (*unsupervised learning*) dan semi-terarah (*semi-supervised learning*), yang dapat digunakan juga diberbagai aplikasi, seperti pengenalan wajah, pengolah citra maupun suara, mengklasifikasi teks, dan lain – lain.

Image Processing

Image processing atau yang sering disebut pengolahan citra adalah suatu bentuk pengolahan atau pemrosesan sinyal dengan input berupa gambar

(*image*) dan ditransformasikan menjadi gambar lain sebagai keluarannya. *Image processing* dilakukan untuk memperbaiki kesalahan data sinyal gambar yang terjadi akibat transmisi dan selama akuisisi sinyal, serta untuk meningkatkan kualitas penampakan gambar agar lebih mudah diinterpretasi oleh sistem.

Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) adalah salah satu jenis neural network yang termasuk kedalam *Deep Learning* yang biasa digunakan pada pengolahan citra gambar. *CNN* bisa digunakan untuk mendeteksi dan mengenali object pada sebuah gambar.

Tensorflow

Tensorflow merupakan salah satu *framework deep learning* dan juga salah satu *library* untuk *data science* yang bersifat *free open source* yang dikembangkan oleh para peneliti dari tim *Google* untuk tujuan melakukan pembelajaran mesin dan penelitian jaringan syaraf dalam.

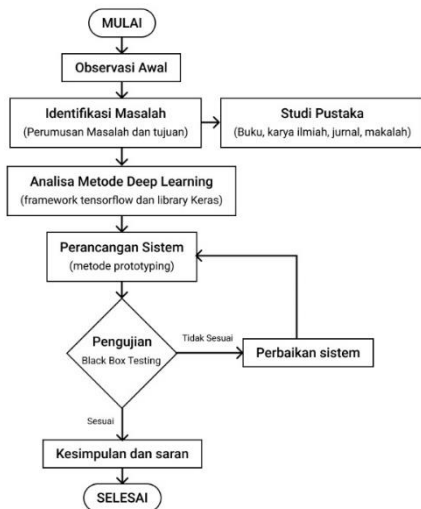
Android

Android adalah sistem operasi untuk perangkat *mobile* atau *smartphone* berbasis *linux* yang meliputi sistem operasi, *middleware* dan aplikasi inti yang mengadopsi

sistem operasi *linux* yang dimodifikasi

METODE PENELITIAN

Tahapan penelitian yang dilakukan oleh penulis dalam membuat penelitian ini digambarkan pada sebuah *flowchart*.



Gambar 1. Alur Tahapan Penelitian

2.1 Pengumpulan Dataset

Dataset yang digunakan adalah data yang tersedia di *Kaggle competition page*. Data yang disimpan pada direktori yang bernama "training_Images" dengan satu file CSV yang sudah disediakan. Data ini dikoleksi oleh *Makerere Al Lab*, dengan jumlah data sebanyak 21,367 gambar daun singkong yang telah dilabeli hasil dari survey di Uganda. Kebanyakan gambar diambil langsung oleh para petani di kebun singkong. *Makerere Al Lab* berkolaborasi dengan para ahli di *National Crops Resource Research Institute (NaCRRI)*.

2.2 Metode Pengembangan Sistem

Metode dalam penelitian ini menggunakan metode *Prototyping*, metode *prototyping* adalah salah satu metode siklus hidup sistem yang didasarkan pada konsep model bekerja (*working model*). Adapun tujuan metode *Prototyping* adalah mengembangkan model menjadi sistem final. Sehingga sistem ini akan dikembangkan dengan cepat dan biayanya menjadi lebih rendah^[5].

Berikut adalah tahapan proses metode *Prototyping* pada pengembangan sistem aplikasi :

1. Pengumpulan Kebutuhan

Langkah pertama kali yang harus dilakukan dalam tahapan metode *Prototyping* adalah mengidentifikasi seluruh perangkat dan permasalahan, analisis dan identifikasi kebutuhan garis besar dari system

2. Membangun Prototyping

Tahap kedua yaitu membangun *Prototyping* dengan membuat perancangan sementara yang berfokus pada penyajian kepada pengguna.

3. Evaluasi Prototyping

Pada tahap evaluasi ini dilakukan untuk mengetahui apakah *Prototyping* sudah sesuai dengan harapan dari pengguna.

4. Pengkodean Sistem

Pada tahap ini *Prototyping* yang sudah disetujui akan

diubah ke dalam bahasa pemrograman.

5. Pengujian Sistem

Pada tahap ini dilakukan pengujian sistem dari perangkat lunak yang sudah dibuat.

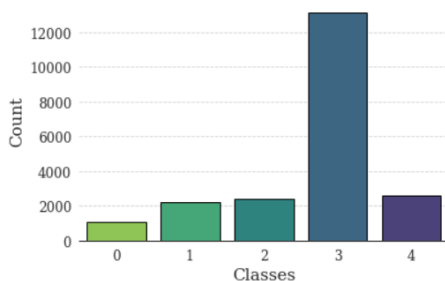
PERANCANGAN SISTEM

Analisa Algoritma CNN

Berikut ini adalah langkah algoritma CNN yang penulis terapkan untuk melakukan klasifikasi citra gambar pada dataset tanaman singkong.

a. Visualisasi Data

Dengan data gambar yang berjumlah 21,367 gambar daun singkong yang terdiri dari 4 jenis, dengan jumlah gambar terbanyak dimiliki oleh gambar label 3 dengan jenis *Cassava Mosaic Disease (CMD)*. Visualisasi data ditunjukkan oleh **Gambar 2**.



Gambar 2. Visualisasi Data

b. Augmentasi Citra (Tensorflow)

Augmentasi Citra sangat penting dalam semua masalah *Machine Learning*, oleh karena itu diperlukan augmentasi terhadap gambar di dataset agar tidak *overfitting* pada training set. Salah satu cara augmentasi gambar

yang tersedia adalah dengan menggunakan *Tensorflow* dengan bantuan *Library Package Keras*.

c. Pelatihan Data

Pada pelatihan data ini menggunakan metode *Stratified K-Fold Cross Validation*. *K-Fold Cross Validation* adalah kumpulan data yang dibagi menjadi sejumlah *K* bagian atau *fold* dimana setiap *fold* digunakan sebagai set pengujian di beberapa titik. Untuk memastikan model tidak *overfitting* pada saat pelatihan data. *Stratified K-Fold Cross Validation* memastikan setiap lipatan memiliki representasi data kelas yang sama. Pada pelatihan ini menggunakan 90% data untuk dilatih, dan 10% untuk *Testing*. Karena memiliki data yang tidak seimbang, maka kelas yang memiliki data paling banyak akan lebih mudah diprediksi. Oleh karena itu, memisahkan dataset secara acak dapat membuat perbedaan besar dalam kinerja model. Dengan menggunakan *Stratified K-Fold Cross Validation* dapat dipastikan bahwa setiap subset data yang berbeda dibagi pada setiap putaran pelatihan dan pengujian dilakukan. Pelatihan ini menggunakan 3 *fold* dan 8 *epoch*. Berikut adalah hasil training pada gambar 3.

```
Epoch 8/8  
1284/1284 [=====] - 300s 24ms/step - loss: 1.1693 - accuracy: 0.8077 - val_loss: 1.1739 - val_ac  
curacy: 0.8626  
Training finished!
```

Gambar 3. Pelatihan Data

d. *Mengubah Ke Tensorflow Lite*

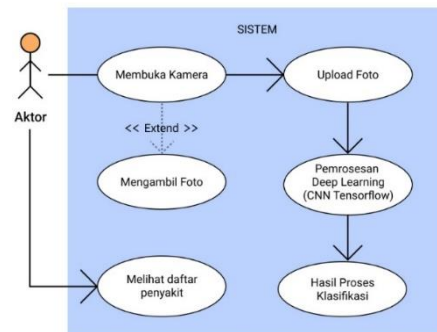
Hasil pelatihan ini kemudian disimpan kedalam model *Tensorflow*. Model ini kemudian dikonversi menjadi model *Tensorflow Lite* dengan ekstensi file (.*tfLite*). *Tensorflow Lite* digunakan karena dapat mengurangi ukuran file dan meningkatkan kecepatan eksekusi tanpa memengaruhi keakuratan sehingga dapat digunakan untuk pengembangan aplikasi pada platform *Android*.

Perancangan Sistem Aplikasi

Setelah proses analisa algoritma *CNN*, selanjutnya adalah melakukan perancangan sistem. Perancangan sistem ini terdiri dari *Use Case Diagram* dan *Activity Diagram*.

a. *Use Case Diagram*

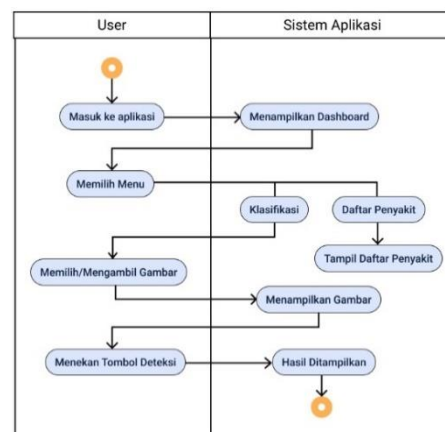
Diagram use case adalah diagram yang mendefinisikan desain hubungan antara pengguna dan sistem. Kasus penggunaan untuk aplikasi ini dimulai dengan aktor membuka kamera, memilih gambar, lalu mengunggahnya, dalam proses tersebut gambar akan dilakukan proses klasifikasi data. Kasus penggunaan sistem ditunjukkan pada Gambar 4.



Gambar 4. *Use Case Diagram*

b. *Activity Diagram*

Diagram Activity adalah grafik yang mendefinisikan diagram alir aktivitas sistem pengguna dan aplikasi, aktivitas akan dipisahkan oleh sejumlah tabel yang memisahkan pengguna dan sistem. Diagram alir ditunjukkan pada Gambar 5.



Gambar 5. *Activity Diagram*

HASIL DAN PEMBAHASAN Pengkodean Aplikasi

Setelah melewati proses rancangan sistem maka proses selanjutnya yang dilakukan adalah melakukan pengodean aplikasi,

berikut ini hasil dari masing- masing pengodean pada pembuatan tampilan yang dilakukan.

a. *Halaman Splash Screen*

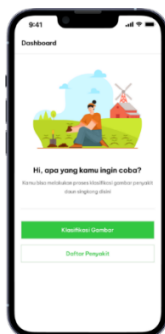
Tampilan *Splash Screen* ditampilkan dalam Gambar ini



Gambar 6. Halaman Splash Screen

b. *Halaman Dashboard*

Tampilan halaman *Dashboard* berisi 2 tombol utama yang menuju ke halaman fitur utama aplikasi ini yaitu klasifikasi penyakit dan juga ke halaman daftar penyakit. Halaman *Dashboard* ditampilkan pada Gambar ini



Gambar 7. Halaman Dashboard

c. *Halaman Klasifikasi Gambar*

Tampilan halaman klasifikasi gambar ini berisi 1 tombol utama yaitu tombol untuk mendeteksi gambar.

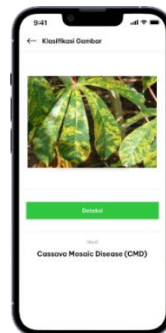
Halaman klasifikasi gambar ditampilkan pada Gambar ini



Gambar 8. Halaman Klasifikasi

d. *Proses Klasifikasi Gambar*

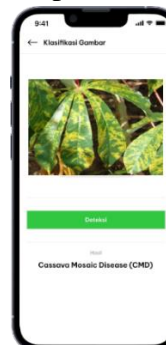
Mengupload gambar yang diambil dari kamera ataupun diambil dari galeri. Proses *upload* dan klasifikasi ini ditampilkan pada Gambar ini



Gambar 9. Proses Klasifikasi

e. *Halaman Daftar Penyakit*

Halaman daftar penyakit dapat dilihat pada Gambar ini



Gambar 10. Halaman Daftar Penyakit

Pengujian Sistem

Setelah tahapan pengkodean sistem, maka proses yang dilakukan selanjutnya adalah melakukan testing pada aplikasi yang telah dibangun. Berikut adalah hasil testing yang telah dilakukan.

a. Black Box Testing

Hasil dari sistem yang telah dibuat dilakukan pengujian sistem menggunakan metode pengujian *Black Box Testing* yang berfokus pada fungsionalitas dari aplikasi. Hasil pengujian *Black Box Testing* terdapat pada Tabel ini

Tabel 1. Hasil Pengujian Black Box Testing.

Skenario	Hasil
Membuka Aplikasi	Sesuai
Izin Akses Kamera	Sesuai
Membuka Halaman Klasifikasi	Sesuai
Membuka Daftar Penyakit	Sesuai
Ambil Gambar Dari Kamera	Sesuai
Ambil Gambar Dari Galeri	Sesuai
Proses Klasifikasi Gambar	Sesuai

b. Usability Testing

Menurut Nielsen (2012), jumlah responden untuk menguji *usability testing* minimal menggunakan 20 orang. Untuk mendapatkan kesimpulan dari hasil sistem yang telah

dibuat ini dilakukan pengujian *Usability Testing* terhadap 20 orang responden. Hasil pengujian ini ditampilkan pada Tabel ini.

Tabel 2. Hasil Pengujian Usability Testing.

Skenario	Hasil
Mengambil Gambar Dengan Kamera	90% Responden Menyukai
Mengambil Gambar Dari Galeri	85% Responden Menyukai
Mendeteksi Penyakit Daun Singkong	100% Responden Menyukai
Melihat Daftar Penyakit	75% Responden Menyukai
Menilai UI dan UX	85% Responden Menyukai
Nilai Guna Aplikasi	95% Responden Menyukai

Hasil dari pengujian Usability Testing pada tabel 4.6 terhadap 20 responden menggunakan 6 skenario pengujian, mendapat hasil yang cukup baik dengan rata-rata 88,3% user menyukai aplikasi ini.

KESIMPULAN

Kesimpulan pada penelitian ini adalah sistem yang dapat melakukan klasifikasi gambar penyakit daun tanaman singkong menggunakan *Deep Learning* dengan Algoritma *Convolutional Neural Network (CNN)* dengan hasil

akurasi mencapai 86%. Hasil dari blackbox testing menunjukkan sistem aplikasi berjalan dengan baik dan sesuai. Lalu hasil usability testing mendapatkan respon yang baik dengan tingkat kepuasan pengguna mencapai 88,3%.

Saran untuk aplikasi ini untuk meningkatkan tingkat akurasi pada sistem ini, bisa dengan menggunakan metode algoritma yang lain dan juga peningkatan hardware yang digunakan, lalu bisa ditingkatkan lagi dari segi UI dan UX aplikasi guna meningkatkan kepuasan pengguna dalam menggunakan aplikasi.

DAFTAR PUSTAKA

- [1] Yunita, S., Jasuma, A., & Sudir, M. (2019). *Sistem Pakar Deteksi Penyakit Pada Tanaman Singkong Expert System to Detect the Disease of Cassava Plants*. *Jurnal Ilmiah Sisfotenika*, 9(1), 24–3
- [2] Botutihe, M. H. (2018). *Sistem Pakar Diagnosa Penyakit Tanaman Singkong Menggunakan Metode Case Based Reasoning*. *Jurnal Rekayasa Teknologi*, Vol 3(1), 28–34. Retrieved from <https://journal.uhamka.ac.id/index.php/rektek/article/view/113>.
- [3] Rozaqi, A. J., Sunyoto, A., & Arief, M. rudyanto. (2021). *Deteksi Penyakit Pada Daun Kentang Menggunakan Pengolahan Citra dengan Metode Convolutional Neural Network*. *Creative Information Technology Journal*, 8(1), 22. <https://doi.org/10.24076/citec.2021v8i1.263>
- [4] Peryanto, A., Yudhana, A., & Umar, R. (2020). *Klasifikasi Citra Menggunakan Convolutional Neural Network dan K Fold Cross Validation*. *Journal of Applied Informatics and Computing*, 4(1), 45–51. <https://doi.org/10.30871/jaic.v4i1.2017>
- [5] Faturrachman, M., & Yustiana, I. (2021). *Sistem Keamanan Pintu Rumah dengan Sidik Jari Berbasis Internet of Things (IOT)*. *Jurnal Teknik Informatika Unika St. Thomas*, 6(2), 379–385.