

KLASIFIKASI JENIS PENYAKIT PADA DAUN TOMAT DENGAN MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORK

El Primo Gemilang ¹⁾ Chairisni Lubis ²⁾

^{1) 2)} Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Tarumanagara
Jln. Letjen S. Parman No. 1, Jakarta, 11440, Indonesia

¹⁾email : elprimo.535180046@stu.untar.ac.id, ²⁾email : chairisnil@fti.untar.ac.id

ABSTRACT

Tomato is one of the farming commodities in Indonesia, easy to plant but easy to get sick. Analyzing the disease in plain view still not yet achieve high accuracy result, so we use the help of Convolutional Neural Network (CNN) algorithm. This research is quantitative, with image of a single tomato leaf that is infected as the input. The constructed model gains an accuracy of 58.33% with 12.716 image consisting of Bacterial Spot, Early Blight, Late Blight, Leaf Mold, Target Spot, Spider Mites, Mosaic Virus, Yellow Leaf Curl Virus, Septoria Leaf Spot and healthy leaf. The conclusion from this research is that classification of Tomato leaf disease using CNN can help achieve a higher accuracy but using LeNet-5 as the model architecture is not very effective.

Key words

convolutional neural network, tomato leaf, classification, image, adaptive gaussian threshold

1. Pendahuluan

Solanum Lycopersicum atau lebih dikenal sebagai tanaman tomat merupakan salah satu tanaman yang dapat tumbuh di berbagai medium, serta di berbagai ketinggian darat. Tanaman ini dapat menghasilkan buah yang banyak apabila tanaman ini sehat. Adanya beberapa jenis penyakit dapat menyerang tanaman ini dengan tingkat kerusakan yang cukup signifikan. Tanaman yang terserang penyakit dapat dilihat dari kondisinya, namun proses klasifikasi dengan kasat mata masih belum akurat dalam menentukan jenis penyakit yang menyerang.

Dengan adanya bantuan algoritma CNN yang diimplementasikan lewat *website*, maka pemilik daun tomat diharapkan dapat terbantu dalam melakukan proses klasifikasi jenis penyakit yang menyerang tumbuhan tomat tersebut dengan memasukkan gambar daun tomat yang terserang penyakit ke dalam *website* untuk dilakukan *preprocessing* gambar kemudian dilakukan proses klasifikasi.

Dari penjelasan diatas, diperlukan sebuah rancangan sistem berbasis *website* yang dapat melakukan klasifikasi terhadap citra daun tomat. Sistem yang dirancang akan melakukan proses *preprocessing* gambar terlebih dahulu, selanjutnya sistem akan melakukan proses *Training* model CNN yang sudah dibuat dengan arsitektur yang digunakan adalah *LeNet-5*. Model yang sudah dibangun ini akan digunakan untuk melakukan proses klasifikasi terhadap data input yang dimasukkan dari *website*.

Rancangan ini juga mempunyai beberapa kegunaan. Dengan adanya sistem ini, maka sistem dapat membantu pengguna dalam melakukan proses klasifikasi jenis penyakit yang menyerang pada data citra yang dimasukkan, serta dapat memberikan informasi mengenai penyakit pada data citra masukkan dan cara penanganan penyakit tersebut.

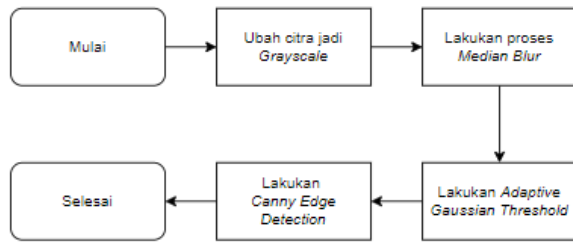
2. Dasar Teori

2.1 Data

Data yang digunakan pada penelitian ini adalah data citra satuan daun tomat. Data citra ini berasal dari Dataset PlantVillage-Dataset dengan jenis gambar daun tomat, serta pengambilan citra langsung pada tanaman tomat yang dijual di Toko Trubus BSD yang berlokasi di Lotte Mart Alam Sutera, BSD Tangerang. Total data yang digunakan adalah 18.260 yang terdiri dari 18.160 citra berasal dari Dataset dan 100 citra berasal dari hasil pengambilan langsung.

2.2 Algoritma

Citra yang sudah didapatkan akan disimpan terlebih dahulu ke dalam database, kemudian akan dilakukan proses *preprocessing* gambar dengan menggunakan algoritma *Grayscale*, *Blurring*, *Adaptive Gaussian Threshold* serta *Canny Edge Detection*. Citra yang sudah dilakukan proses *preprocessing* akan disimpan kembali ke dalam database, kemudian akan digunakan sebagai citra latih dan citra uji dalam pembuatan model CNN.



Gambar 1 Alur proses preprocessing gambar

Pada Gambar 1, citra akan dilakukan perubahan *channel* warna dari RGB menjadi *Grayscale*. Hal ini diperlukan untuk mengurangi informasi yang akan tersimpan dalam sebuah *pixel*. Proses ini menggunakan perhitungan rumus sebagai berikut:

$$\text{Grayscale} = 0.299R + 0.587G + 0.114B \quad (1)$$

Keterangan:

R : Nilai channel Red
G : Nilai channel Green
B : Nilai channel Blue

Selanjutnya, citra *grayscale* akan dilakukan proses *Median Blurring* yang fungsinya untuk mengurangi *noise* yang ada pada citra. Citra yang sudah dilakukan *median blur* akan masuk ke tahap *Adaptive Gaussian Threshold*. Proses ini menggunakan perhitungan rumus sebagai berikut:

$$g(x, y) = \begin{cases} 0, & f(x, y) < T_1 \mid f(x, y) > T_2 \\ 1, & T_1 \leq f(x, y) \leq T_2 \end{cases} \quad (2)$$

Keterangan:

$f(x, y)$: Nilai Gradient pixel
T1 : Nilai batasan threshold bawah
T2 : Nilai batasan threshold atas

Pada tahap ini, citra akan diubah menjadi citra hitam dan putih, sesuai dengan nilai pada *pixel* masing-masing. Citra kemudian akan dilakukan proses *Canny Edge Detection* yang fungsinya untuk menentukan batasan luar daun pada citra.

Pada tahapan ini, terdapat beberapa metode yang digunakan seperti *Gaussian Smoothing*, penentuan arah Gradient, perhitungan *non maxima suppression* pada Gradient, deteksi *double threshold* dan *Edge Connection*. Perhitungan *Gaussian Smoothing* menggunakan rumus sebagai berikut:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (3)$$

Keterangan:

x : Indeks baris dari pixel filter Gaussian
y : Indeks kolom dari pixel filter Gaussian
 σ : Standar Deviasi

Semakin besar standar deviasi yang ditetapkan, maka semakin besar akurasi filter ini dan semakin kecil nilai rasio *signal-to-noise* yang dihasilkan. Pada tahapan selanjutnya, proses penentuan arah Gradient menggunakan rumus berikut:

$$A(i, j) = \sqrt{A_x^2(i, j) + A_y^2(i, j)} ; \theta(i, j) = \arctan \frac{A_y(i, j)}{A_x(i, j)} \quad \dots(4)$$

Keterangan:

$A_x(i, j)$: Magnitudo pada Gaussian Smoothing bidang X

$A_y(i, j)$: Magnitudo pada Gaussian Smoothing bidang Y

$\theta(i, j)$: Arah Magnitudo

Proses *non maxima suppression* akan merubah nilai Magnitudo dengan cara membandingkan nilai Magnitudo pixel $A(i, j)$ dengan nilai tetangga dari pixel tersebut. Apabila nilai $A(i, j)$ lebih besar dari tetangganya, maka nilai pixel tersebut tidak akan dirubah. Namun jika sebaliknya, maka nilai pixel tersebut akan diubah menjadi 0[1].

Proses selanjutnya adalah deteksi *edge* dengan menggunakan dua nilai threshold. Apabila nilai amplitudo dari pixel yang sedang diperiksa berada dibawah threshold terkecil, maka pixel tersebut dianggap sebagai titik *non-edge*. Apabila sebaliknya, pixel tersebut akan dianggap sebagai titik *edge*. Namun, apabila nilai amplitudo pixel tersebut diantara kedua threshold, maka akan diperiksa terhadap pixel tetangga dengan jarak 8 pixel dari pixel tersebut. Jika lebih besar, maka titik pixel tersebut dianggap sebagai titik *edge*, jika tidak maka akan dianggap sebagai titik *non-edge*.

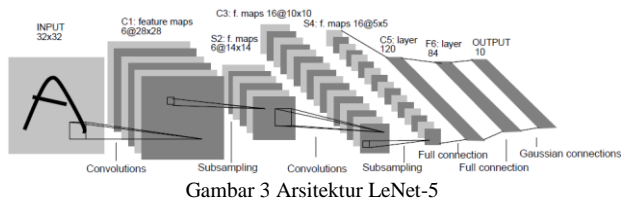
Proses *preprocessing* diakhiri dengan metode *masking*, dimana *mask* yang sudah terbentuk dari proses deteksi *edge* akan dipakai untuk menghilangkan latar belakang pada gambar asli. Proses ini akan menggabungkan gambar dengan keseluruhan pixel berwarna hitam, gambar asli, kemudian *mask* yang terbentuk. Citra keluaran yang akan digunakan dapat dilihat pada Gambar 2.



Gambar 2. Hasil output proses preprocessing citra

2.3 CNN

Convolutional Neural Network merupakan salah satu bentuk jaringan saraf tiruan seperti jaringan saraf manusia. Arsitektur CNN terdiri dari berbagai macam jenisnya, fungsinya adalah membentuk sebuah model jaringan sehingga dapat dilakukan proses pembelajaran. Pada perancangan ini, digunakan arsitektur *LeNet-5*.



Gambar 3 Arsitektur LeNet-5

Arsitektur ini merupakan arsitektur dalam CNN yang mempunyai 7 lapisan tidak termasuk lapisan masukan. Lapisan masukan menerima gambar dengan ukuran 32x32 pixel. Pada Gambar 3, terdapat lapisan C1 yang mempunyai 6 *feature map* yang saling terhubung dengan lapisan masukan dengan jarak 5x5. Lapisan ini mempunyai 156 parameter dan 122.304 koneksi. Selanjutnya, pada lapisan C2 terdapat 6 *feature map* dengan ukuran 14x14 dan masing-masing *feature map* terhubung dengan lapisan masukan pada C1 dengan jarak 2x2. Lapisan ini mempunyai 12 parameter dengan 5.880 koneksi [2].

3. Rancangan dan Pembuatan

3.1 Rancangan sistem

Sistem yang dirancang berbasis *website* yang fungsinya untuk melakukan proses klasifikasi jenis penyakit pada gambar daun tomat yang dimasukkan. Sistem dirancang untuk memberikan informasi mengenai hasil klasifikasi jenis penyakit yang dialami pada gambar daun tomat yang dimasukkan, serta memberikan informasi terhadap jenis penyakit yang dapat diklasifikasikan. Sistem dapat menerima input citra daun tomat dengan tipe ekstensi PNG, JPG, dan JPEG.

Perancangan diawali dengan pengumpulan data berupa Dataset PlantVillage-Dataset, serta pengumpulan citra hasil foto langsung dari tumbuhan daun tomat. Citra yang sudah terkumpul akan dimasukkan ke dalam sistem basis data, dan kemudian setiap citra yang sudah dimasukkan akan dilakukan proses *preprocessing*.

Sistem kemudian akan melanjutkan tahapan pembuatan model CNN, dengan membagi data citra yang sudah dilakukan *preprocessing* sejumlah nilai setting Training pada sistem basis data dan sejumlah nilai setting Testing pada sistem basis data. Jumlah data training yang digunakan pada perancangan ini adalah 70% dari data masing-masing jenis penyakit. Sementara, data testing yang digunakan pada perancangan ini adalah 30% dari data masing-masing jenis penyakit yang tidak dijadikan data training.

3.1.1 Perencanaan Sistem

Pada tahapan ini, dilakukan perencanaan terhadap tujuan, batasan, serta metode yang digunakan dalam perancangan. Tujuan dari perancangan sistem ini adalah pembuatan sistem yang digunakan untuk melakukan proses klasifikasi jenis penyakit pada gambar daun tomat yang dimasukkan, serta memberikan informasi mengenai jenis penyakit yang menyerang pada daun tomat.

Batasan dari perancangan ini yakni format gambar yang dimasukkan berupa file PNG, JPEG dan JPG. Gambar yang dimasukkan merupakan gambar satuan dari daun tomat yang terserang penyakit dengan tingkat kecerahan gambar tidak terlalu gelap. Selanjutnya, metode *preprocessing* yang digunakan seperti metode *Grayscale* adalah metode *Grayscale Luminosity*. Metode ini digunakan untuk merubah nilai channel RGB menjadi *Grayscale* lebih akurat, dengan memperhatikan tingkat kecerahan serta nilai spektrum warna yang dapat ditangkap oleh mata manusia.

3.1.2 Analisis Sistem

Pada tahapan analisis sistem, terdapat beberapa spesifikasi yang diperlukan untuk perancangan sistem ini. Spesifikasi perangkat keras yang diperlukan antara lain sebagai berikut:

1. Processor
2. Memory
3. Storage Unit
4. Graphic Card
5. Layar
6. Keyboard

Sementara itu, spesifikasi perangkat lunak yang dibutuhkan adalah sebagai berikut:

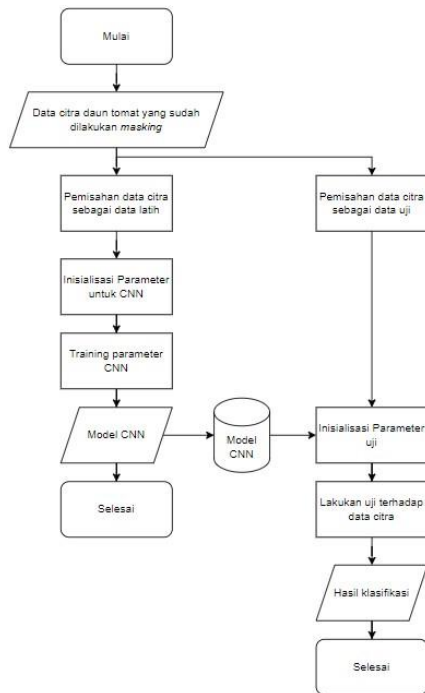
1. Sistem Operasi Microsoft Windows 10
2. Framework .NET 4.8
3. Visual Studio 2019

3.1.3 Perancangan Sistem

Pada tahap ini, diperlukan rancangan diagram alur sistem, diagram hierarki, serta rancangan tampilan antarmuka dari sistem yang akan dirancang.

3.1.3.1 Rancangan Diagram Alur

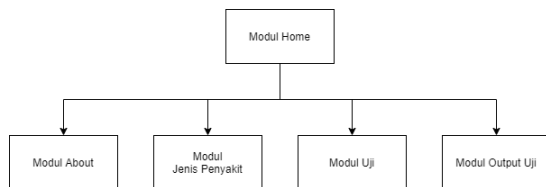
Diagram alur merupakan sebuah diagram yang digunakan untuk menggambarkan alur dari sebuah proses. Pada perancangan ini, diagram alur proses klasifikasi jenis penyakit pada daun tomat mempunyai alur sebagai berikut:



Gambar 4 Diagram alur

3.1.3.2 Rancangan Diagram Hierarki

Diagram hierarki merupakan sebuah diagram yang digunakan untuk mengetahui hubungan antar modul yang dirancang. Diagram Hierarki pada perancangan ini dapat dilihat sebagai berikut:



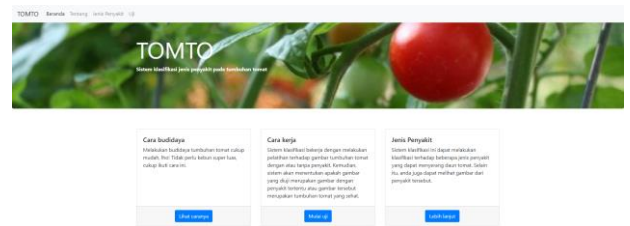
Gambar 5 Diagram hierarki

3.1.3.3 Rancangan Tampilan Antarmuka

Rancangan tampilan antarmuka digunakan untuk menjelaskan tampilan yang akan digunakan oleh pengguna. Rancangan antarmuka pada perancangan ini adalah sebagai berikut:

a. Modul Beranda

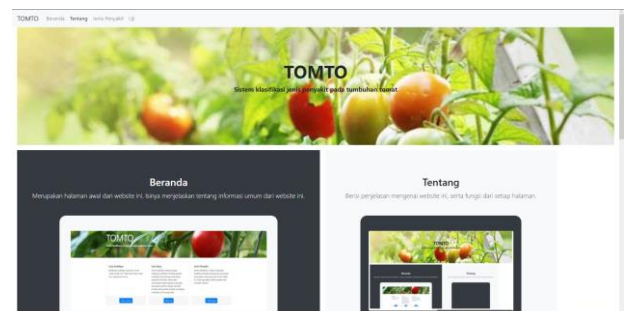
Modul ini merupakan modul pertama yang akan dilihat oleh pengguna ketika mengakses *website*. Modul ini berisi tentang cara melakukan budidaya tanaman tomat, serta tautan mengenai penjelasan modul yang ada pada *website* dan jenis penyakit yang dapat diklasifikasikan.



Gambar 6 Modul Beranda

b. Modul Tentang

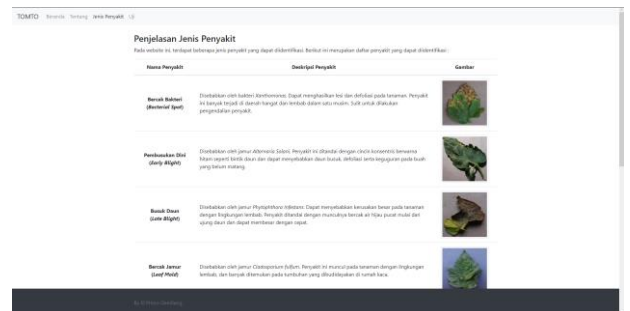
Modul ini berisi mengenai penjelasan modul-modul yang ada pada *website* serta informasi perancang *website*.



Gambar 7 Modul Tentang

c. Modul Jenis Penyakit

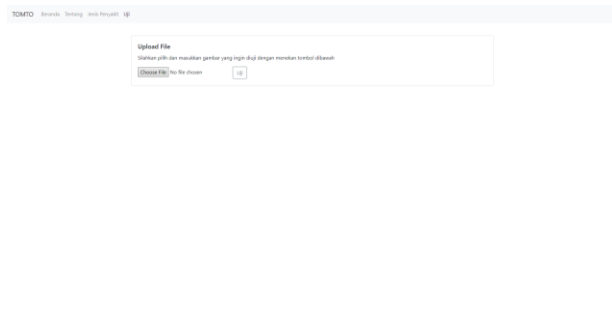
Modul Jenis Penyakit berisi tentang jenis penyakit yang dapat diklasifikasikan oleh sistem, serta informasi mengenai penyakit tersebut dilengkapi dengan gambar penyakitnya.



Gambar 8 Modul Jenis Penyakit

d. Modul Uji

Modul Uji merupakan modul utama dari sistem ini. Pada modul ini, pengguna dapat melakukan proses upload gambar daun tomat yang terserang penyakit, dan sistem akan mengeluarkan hasil klasifikasi serta tingkat akurasi model yang digunakan pada saat Training.

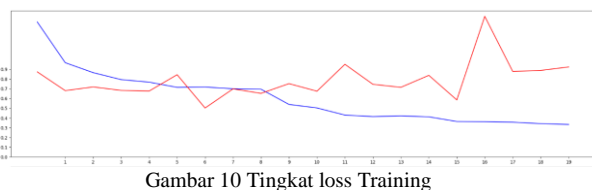


Gambar 9 Modul Uji

4. Pengujian dan Pembahasan

Rancangan dilakukan pengujian dengan menggunakan metode *blackbox testing* dan *User Acceptance Testing (UAT)*. *Blackbox Testing* merupakan pengujian yang dilakukan pada rancangan yang sudah dibuat dengan memperhatikan hasil uji dan mengamati fungsionalitas dari sistem yang dirancang. Pengujian ini hanya melakukan evaluasi ketat terhadap tampilan.

Selanjutnya, pengujian dilakukan dengan mengevaluasi model yang sudah dibentuk. Model mempunyai tingkat akurasi tertinggi sebesar 90.15%, dengan tingkat *loss* sebesar 0.3419. Model juga dibatasi dengan menggunakan 20 Epochs. Grafik hasil pembuatan model dapat dilihat sebagai berikut:

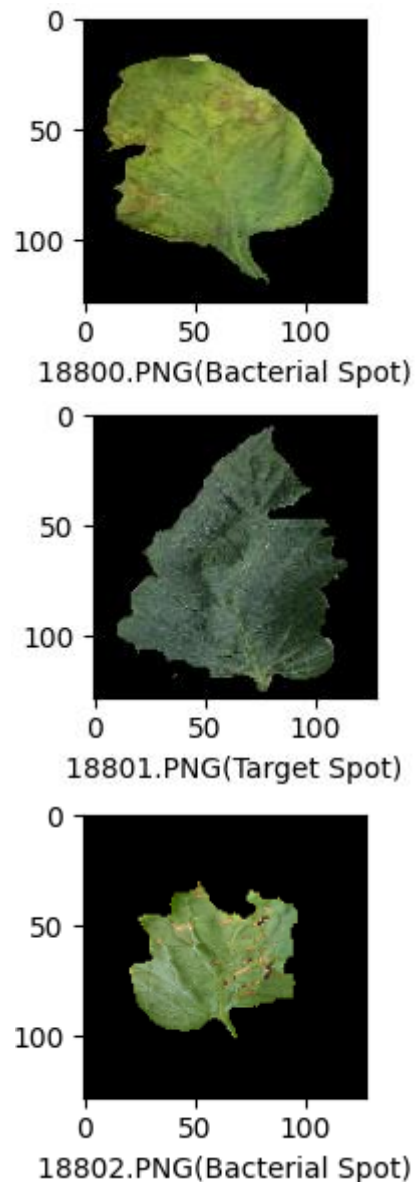


Gambar 10 Tingkat loss Training



Gambar 11 Tingkat accuracy Training

Sementara, hasil prediksi dari beberapa data Testing adalah sebagai berikut:



Gambar 12 Hasil prediksi data Testing

5. Kesimpulan dan Saran

5.1 Kesimpulan

Kesimpulan yang didapat setelah dilakukan tahap pengujian terhadap sistem yang dirancang adalah sebagai berikut:

1. Sistem yang dirancang sudah dapat melakukan proses klasifikasi terhadap gambar input dengan menggunakan metode CNN.
2. Sistem dapat melakukan klasifikasi dengan jumlah kelas penyakit sebanyak 10 buah yaitu Bacterial Spot, Early Blight, Late Blight, Leaf Mold, Target Spot, Spider Mites, Septoria Leaf Spot, Mosaic Virus, Yellow Leaf Curl Virus serta Sehat.
3. Hasil akurasi terbaik pada model yang dirancang adalah 90.15% dengan tingkat *loss* sebesar 0.3419.

5.2 Saran

Saran yang ada untuk sistem klasifikasi jenis penyakit daun tomat dengan menggunakan metode CNN adalah sebagai berikut:

1. Menggunakan arsitektur yang lebih modern agar mendapatkan tingkat *loss* dan akurasi yang lebih baik.
2. Perlu dilakukan optimasi terhadap metode *preprocessing* yang sudah dilakukan agar menghasilkan citra yang lebih jernih.

REFERENSI

- [1] Kewen Liu, Kang Xiao, Hongxia Xiong, "An Image Edge Detection Algorithm Based on Improved Canny", <https://www.atlantispress.com/article/25873669.pdf>
- [2] Yann Le Cun, et al. "Gradient-Based Learning Applied to Document Recognition", <https://yann.lecun.com/exdb/publish/pdf/lecun-98.pdf>

Nama Penulis, memperoleh gelar B.S dan M.Sc dari University Of New South Wales, Australia tahun 1995 dan 1997. Kemudian tahun 2002 memperoleh Ph.D dari Cambridge University, Inggris. Saat ini sebagai Staf Pengajar program studi Teknik Informatika Universitas Maju Sekali.

El Primo Gemilang, mahasiswa S1, program studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Tarumanagara.

Chairisni Lubis, M. Kom, memperoleh gelar S. Kom dari Universitas Indonesia pada tahun 1989 dan gelar M. Kom dari Universitas Indonesia pada tahun 2000.