

## Deteksi Penyakit Tomat Melalui Citra Daun menggunakan Metode *Convolutional Neural Network*

Roni Halim Saputra, Rito Cipta Sigitta Hariyono\*, Fathulloh  
Program Studi Informatika, Universitas Peradaban, Indonesia

### Article Info

#### Article history:

Submitted December 5, 2022

Accepted January 31, 2023

Published February 2, 2023

#### Keywords:

*convolutional neural network*,  
deteksi penyakit tomat,  
*deep learning*,  
*neural network*

*convolutional neural network*,  
*leaf disease of tomato*,  
*deep learning*,  
*neural network*

### ABSTRACT

Tomat merupakan tanaman hortikultura yang memiliki nilai ekonomi tinggi dan banyak ditanam oleh masyarakat Indonesia sebagai negara agraris. Menurut data BPS tahun 2019, produktivitas tomat mengalami peningkatan dalam empat tahun terakhir, namun dalam proses perawatan tanaman sering terjadi kendala dalam mengidentifikasi jenis penyakit yang terdapat pada daun tanaman tomat karena memiliki gejala yang hampir mirip dan sulit dibedakan. Penelitian ini mencoba mengidentifikasi penyakit pada tanaman tomat melalui citra daun menggunakan *deep learning* dengan metode *convolutional neural network*. Penerapan model deteksi menggunakan dua jenis arsitektur *convolutional neural network* yaitu standar LeNet-5 dan LeNet-5 (*custom*) yang telah dimodifikasi. LeNet-5 standar berhasil mendeteksi penyakit pada daun tomat dengan akurasi 90%, sedangkan pada arsitektur LeNet-5 yang telah dimodifikasi pada *network layer* dan parameter yang digunakan, menghasilkan akurasi yang lebih optimal dengan akurasi sebesar 95%.

*Tomatoes are horticultural crops that have a high economic value and widely planted by the people of Indonesia as an agricultural country. According to BPS data for 2019, tomato productivity has increased in the last four years, but in the process of caring for plants, there are often problems in identifying the types of diseases found on the leaves of tomato plants because they have almost similar symptoms and are difficult to distinguish. This study tries to identify diseases in tomato plants through leaf images using Deep Learning with the Convolutional Neural Network method. The implementation of the detection model uses two types of Convolutional Neural Network architectures, namely standard LeNet-5 and modified LeNet-5 (custom). The standard LeNet-5 successfully detects disease on tomato leaves with an accuracy of 90%, while the modified LeNet-5 architecture on the network layer and parameters used produces more optimal accuracy with an accuracy of 95%.*



### Corresponding Author:

Rito Cipta Sigitta Hariyono

Program Studi Informatika, Universitas Peradaban

Jl. Raya Pagojengan Km. 3 Paguyangan, Brebes, Jawa Tengah 52276

Email: \*ritocipta@peradaban.ac.id

## 1. PENDAHULUAN

Perangkat lunak dan perangkat keras yang didukung teknologi kecerdasan buatan (*Artificial Intelligence*) dikembangkan untuk mempercepat, mempermudah dan membantu pekerjaan manusia di era industri 4.0, termasuk dibidang pertanian. Indonesia sebagai negara agraris menghasilkan berbagai macam jenis produk industri pertanian dan perkebunan yang merupakan sektor penting sebagai penopang ekonomi masyarakat. Salah satu jenis tanaman yang banyak ditanam oleh penduduk Indonesia adalah tomat. Tomat merupakan tanaman hortikultura yang banyak dibudidayakan masyarakat karena memiliki nilai ekonomi yang tinggi sebagai komoditas olahan makanan, minuman, sayur, buah dan kosmetik. Berdasarkan data Badan Pusat Statistik Nasional (BPS) tahun 2019, produktivitas tanaman tomat mengalami peningkatan sebesar 24% dalam empat tahun terakhir. Peningkatan ini perlu didukung dengan menjaga kualitas dan kuantitas produksi tomat dari proses penanaman hingga proses panen.

Daun tanaman tomat merupakan bagian yang memiliki peran dan fungsi penting dalam proses pertumbuhan tanaman. Penyakit pada daun tanaman dapat menjadi salah satu faktor yang mempengaruhi proses

pertumbuhan tanaman. Jika daun terserang suatu penyakit, hal ini akan membahayakan kondisi tanaman dan akan berpengaruh pada hasil produksi panen [1]. Jenis penyakit pada daun dapat dijelaskan dengan gejala yang berbeda, walaupun gejala yang ada tidak selalu dapat menjelaskan nama penyakit yang menyerang. Hal ini terjadi karena banyaknya jenis penyakit yang memiliki gejala serupa dan hampir terlihat sama. Identifikasi penyakit pada daun tanaman tomat biasanya dilakukan di laboratorium yang membutuhkan banyak waktu dan biaya.

*Deep convolutional neural network* (CNN) adalah jenis jaringan syaraf tiruan (JST) khusus yang menggunakan konvolusi sebagai pengganti perkalian matriks umum di setidaknya salah satu lapisannya [2]. CNN merupakan metode yang digunakan dalam pengembangan model pengenalan pola berdasarkan citra digital dengan pendekatan hierarki yang dapat mempelajari lapisan fitur dari data masukan berupa citra digital [3], sehingga dengan menggunakan CNN tidak diperlukan proses ekstraksi fitur morfologi pada citra daun yang terkena penyakit. Proses identifikasi penyakit pada citra daun dilakukan secara otomatis, tidak bergantung pada pendefinisian fitur dan proses segmentasi citra daun. Semua proses identifikasi penyakit dilakukan menggunakan metode CNN. Aktivitas pelatihan pada CNN melibatkan 4 parameter yang nilainya dapat diubah-ubah yaitu *epoch*, *mini-batch*, *learning rate* dan *momentum*. Pada proses pelatihan semua *data set* melewati satu proses *forward* dan *backward* di mana nilai bobot diubah hingga menjadi konvergen, siklus ini dinamakan dengan *epoch* [4]. *Mini-Batch* merupakan variasi dari *batch* yang mengambil nilai rata-rata *loss* dan mereduksi variasi *update* parameter agar konvergensi lebih stabil dan mengoptimalkan penggunaan memori sehingga lebih efisien [5]. *Learning rate* mempengaruhi waktu pencapaian target, di mana secara perlahan akan mengoptimalkan perubahan nilai *weight* dan menghasilkan *loss* yang lebih kecil. *Learning rate* merupakan nilai konstan antara 0 dan 1. Tidak adanya nilai yang pasti ini menyebabkan dibutuhkan pengalaman praktis agar diperoleh hasil yang maksimal [6].

Penelitian sebelumnya telah dilakukan untuk mengidentifikasi jenis tumbuhan menggunakan citra daun berbasis jaringan syaraf tiruan [7], di mana penelitian ini menghasilkan persentase keberhasilan sebesar 93,6%. Selain itu, penelitian tentang deteksi citra penyakit daun pada tanaman jagung dengan segmentasi berbasis *k-means* menghasilkan akurasi 90% [8]. Penelitian sebelumnya juga sudah dilakukan untuk mengidentifikasi penyakit pada daun tanaman [1]. Penelitian tersebut berhasil mendeteksi penyakit pada daun tembakau dengan nilai akurasi sebesar 73%. Penelitian juga dilakukan pada kasus klasifikasi jenis penyakit pada daun tanaman tomat [9] dengan metode *back propagation* yang menghasilkan nilai akurasi sebesar 78%. Kedua penelitian tersebut menggunakan metode *Artificial Neural Network* (ANN). Metode ANN ini memerlukan proses untuk mengidentifikasi ciri morfologi pada daun tanaman dan proses ekstraksi fitur pada setiap citra daun harus dilakukan secara manual untuk membedakan beberapa penyakit atau mengklasifikasi jenis daun. Hal ini akan mempersulit jika terdapat ragam jenis penyakit dengan banyak jenis fitur morfologi dan harus melalui proses identifikasi dan ekstraksi fitur dalam bentuk data tabular.

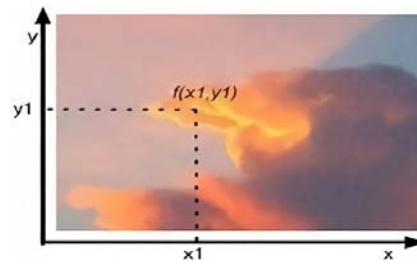
Pada penelitian ini, penulis melakukan identifikasi penyakit pada tanaman tomat melalui citra daun menggunakan *deep learning* dengan metode CNN LeNet-5 yang telah dimodifikasi. Sebagai perbandingan, penerapan model deteksi menggunakan dua jenis arsitektur CNN yaitu standar LeNet-5 dan LeNet-5 (*Custom*) yang telah dimodifikasi dengan maksud untuk mendapatkan hasil akurasi yang lebih optimal.

## 2. METODE PENELITIAN

### 2.1 Citra Digital

Citra adalah sebuah istilah lain dari gambar, sebagai salah satu komponen multimedia yang memiliki peranan sangat penting sebagai bentuk informasi visual. Citra memiliki karakteristik yang tidak dimiliki oleh jenis data lain karena kaya dengan informasi. Sebuah citra digital bisa termasuk jenis vektor atau bitmap yang tergantung pada resolusi citra tersebut tetap atau berubah. Citra bitmap mempunyai jumlah piksel yang terbatas dan banyaknya piksel di baris dan kolom sebuah citra bitmap tidak bisa berubah. Piksel yaitu sebuah elemen individu terkecil di sebuah citra dan menyimpan nilai keterangan warna di titik tersebut. Jika citra bitmap diperbesar, maka citra terlihat pecah. Umumnya, piksel tersimpan di komputer sebagai citra bitmap atau raster map, berbentuk susunan 2D integer kecil dan ditransmisi atau disimpan dalam bentuk terkompres. Dan asal citra vektor yaitu dari matematika geometri, suatu vektor terdiri dari titik yang mempunyai arah dan panjang. Umumnya, elemen bitmap dan vektor tergabung dalam sebuah citra. Misalnya, sebuah *billboard* yang mempunyai teks (vektor) dan foto (bitmap). Citra digital dapat dilihat dengan perangkat lunak untuk menampilkan citra seperti *Windows photo viewer*. *Web browser* dapat melihat format citra standar, seperti GIF, JPEG, dan PNG, sedangkan web browser melihat format SVG [10].

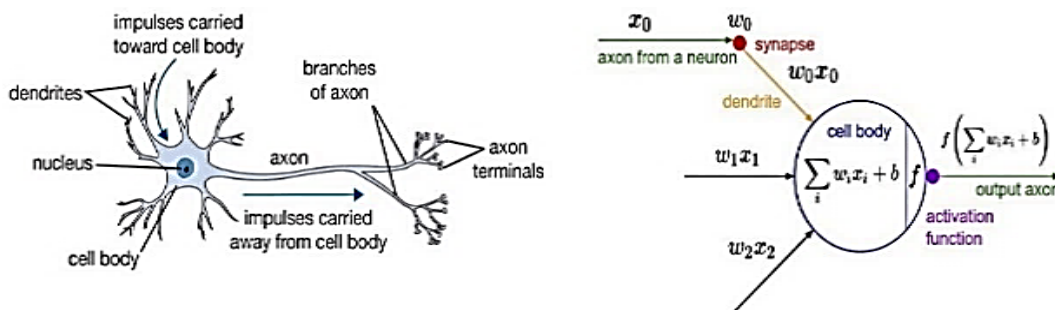
Sebuah citra yang merepresentasikan informasi visual dari sebuah objek yang dapat didefinisikan sebagai sebuah fungsi 2 dimensi  $f(x,y)$ , di mana  $x$  dan  $y$  adalah koordinat spasial dan  $f$  merupakan besaran pada setiap pasang koordinat  $(x,y)$  disebut intensitas atau tingkat/derajat keabuan, jika semua  $x,y$  dan  $f$  adalah berhingga dan dalam jumlah diskrit maka citra tersebut disebut sebagai citra digital [11], sebagaimana Gambar 1. Citra digital direpresentasikan melalui piksel (*pixel*), merupakan kependekan dari '*picture element*' yang diindeks sebagai lokasi koordinat  $(x,y)$  atau *column-row* ( $c,r$ ) pada citra. Piksel merepresentasikan unsur elemen terkecil dalam citra digital dan bernilai numerik yang merupakan informasi dasar dalam sebuah citra.



Gambar 1. Citra Digital [11]

## 2.2 Artificial Neural Network (ANN)

*Artificial Neural Network* (ANN) dalam bahasa Indonesia disebut jaringan syaraf tiruan adalah sebuah *model* algoritma kecerdasan buatan (*artificial intelligence*) yang terinspirasi dari sistem jaringan syaraf makhluk hidup (otak manusia). Di bidang ilmu komputer istilah ANN merupakan sebuah *model* algoritma komputasi yang dibuat dengan kemampuan mesin untuk beradaptasi, belajar dan mengelompokkan data berbasis pemrosesan secara paralel dan sistematis. ANN terdiri atas komponen yang mendasari algoritma jaringan syaraf tiruan yang disebut *Neuron*, atau disebut juga sebagai *node* yang saling terhubung satu sama lain, di mana setiap *neuron* mendapatkan nilai masukan dari banyak *neuron* yang saling terhubung, kemudian dilakukan operasi perhitungan matematis untuk mendapatkan sebuah nilai keluaran, sebagaimana diilustrasikan pada Gambar 2 [12].



Gambar 2. Struktur neuron *biologi* dan *neural network*

Komponen utama *neuron* biologi dan komponen pada *neural network* dapat dikelompokkan menjadi 3 bagian yaitu:

1. *Dendrites* sebagai penerima informasi atau masukan data,
2. *Cell body* sebagai tempat pengolahan, terdapat inti sel atau *nucleus* sebagai pengolahan informasi,
3. *Axon* bertugas mengirimkan *impuls* sinyal ke jalur keluaran kepada *neuron* lain.

Hubungan *neuron* secara biologi dan *neuron* pada *neural network*, *dendrites* direpresentasikan sebagai masukan atau informasi yang akan diolah oleh *cell body*, selanjutnya hasil perhitungan yang telah dilakukan akan dikeluarkan pada *axon* yang merepresentasikan sebuah keluaran.

## 2.3 Penyakit Tanaman Tomat

Tanaman pada umumnya dikatakan terserang penyakit bila pertumbuhannya menyimpang dari keadaan normal. Penyebabnya bisa dari beberapa macam, di antaranya disebabkan oleh jamur atau cendawan, bakteri dan virus. Beberapa jenis penyakit yang terdapat pada tanaman tomat di antaranya penyakit *yellow leaf curl*, *potato virus y*, *tomato mozaik virus*, dan bercak daun [13].

## 2.4 Convolutional Neural Network

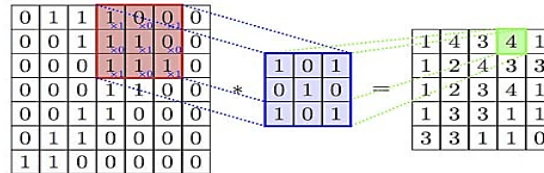
*Convolutional neural network* (CNN) merupakan salah satu arsitektur jaringan syaraf tiruan *feed-forward network* dan merupakan pengembangan dari *Multilayer Perceptron* (MLP) didesain untuk membuat pemodelan dengan masukan berupa data dua dimensi seperti gambar atau video[14]. CNN termasuk dalam sebuah teknik *Deep Learning* karena kedalaman arsitektur jaringan yang digunakan memiliki jumlah lapisan yang banyak. Secara teknis CNN adalah sebuah arsitektur yang dapat dibangun menyesuaikan kebutuhan untuk data yang bersifat tidak terstruktur (*unstructured data*) seperti dalam penelitian ini menggunakan data citra digital yang bersifat 2 dimensi, namun pada penerapannya CNN juga dapat digunakan untuk tipe data 1 dimensi dan 3 dimensi. *LeNet-5* merupakan salah satu jenis arsitektur yang banyak digunakan, secara umum untuk tujuan pengenalan dan pembelajaran [4]. Terdapat beberapa jenis arsitektur selain *LeNet-5*, yaitu *AlexNet*, *VGG-16*, *Inception*, *ResNet-5* dan lain sebagainya. Penggunaan arsitektur disesuaikan dengan tujuan pembangunan *model* CNN dan permasalahan yang ingin diselesaikan.

## 2.5 Feature Learning

Lapisan-lapisan yang terdapat pada *feature learning* berguna untuk mentranslasikan sebuah masukan menjadi fitur berdasarkan ciri dari sebuah masukan tersebut [3]. Secara sederhana lapisan ini bertujuan untuk ekstraksi fitur pada gambar (masukan) menjadi sebuah vektor dan terdiri dari *Convolutional Layer* dan *Pooling Layer*.

### 2.5.1 Convolutional Layer

Lapisan ini dilakukan operasi konvolusi pada keluaran *layer* sebelumnya berupa citra digital [11]. Dalam pengolahan citra digital, operasi konvolusi berarti mengaplikasikan sebuah *kernel* atau dapat mudah dipahami sebagai *filter*.



Gambar 3. Convolution layer

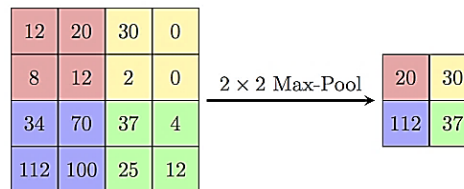
Gambar 3 merupakan sebuah citra satu dimensi dengan ukuran  $(7 \times 7)$  dioperasikan dengan *filter* berukuran  $(3 \times 3)$  sehingga mendapatkan dimensi citra baru dengan ukuran  $(5 \times 5)$ , *filter* bergerak dari sudut kiri atas ke kanan bawah dengan Persamaan (1) sebagai berikut:

$$h(x) = f(x) * g(x) = \int_{-\infty}^{\infty} f(a)g(x - a)da \quad (1)$$

di mana,  $h(x)$  adalah hasil konvolusi,  $f(x)$  adalah sinyal masukan, dan  $g(x)$  adalah kernel atau *filter*. Tanda (\*) menyatakan operasi konvolusi dan  $a$  merupakan variabel bantu (*dummy variable*), *filter*  $g(x)$  merupakan suatu jendela yang dioperasikan secara bergeser pada sinyal masukan  $f(x)$ , yang dalam hal ini, jumlah perkalian dua fungsi pada setiap titik merupakan hasil konvolusi yang dinyatakan dengan  $h(x)$ .

### 2.5.2 Pooling Layer

*Pooling layer* merupakan lapisan yang bertujuan mereduksi ukuran sebuah citra digital. Dalam arsitektur CNN metode *pooling* yang digunakan adalah *Maxpooling*, yaitu dengan membagi keluaran dari *convolutional layer* menjadi beberapa *grid* kecil kemudian mengambil nilai maksimal dari setiap *grid* untuk menyusun matriks citra yang telah direduksi.



Gambar 4. Pooling layer

Gambar 4 merupakan proses *pooling layer*, mengaplikasikan jenis *Maxpooling* pada citra berukuran dimensi  $(4 \times 4)$  dan *filter* berukuran  $(2 \times 2)$ .

### 2.5.3 Classification

Lapisan *Classification* bertujuan untuk mengklasifikasi setiap *neuron* yang telah diekstraksi pada *feature learning*, proses pada lapisan ini melalui tiga bagian yaitu:

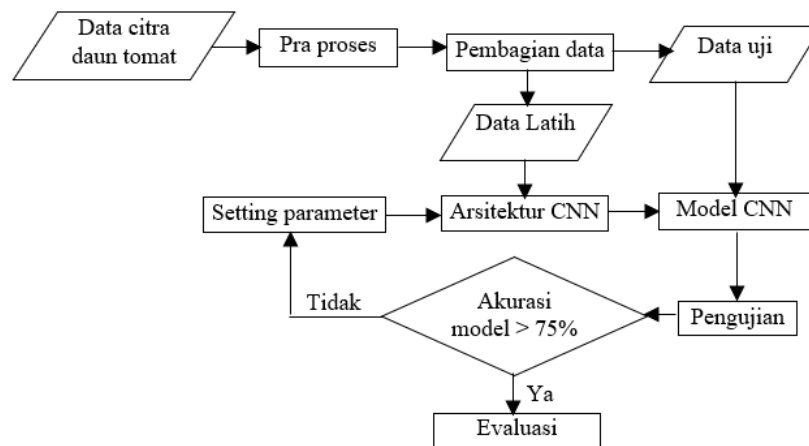
1. *Flatten*, yaitu proses membentuk ulang sebuah fitur (*reshape feature map*), menjadi sebuah vektor yang akan digunakan sebagai masukan pada jaringan syaraf tiruan.
2. *Fully-connected*, lapisan ini merupakan proses penerapan *Multi Layer Preceptron* (MLP) dengan tujuan untuk melakukan transformasi pada dimensi data agar dapat diklasifikasikan secara *linear* dengan data masukan berupa vektor.
3. *Softmax*, *softmax function* merupakan fungsi aktivasi *non-linear* dengan *range* nilai 0 hingga 1 dengan Persamaan (2) sebagai berikut.

$$f_i(x^{\rightarrow}) = \frac{e^{x_i}}{\sum_{j=1}^J e^{x_j}} \quad (2)$$

Fungsi aktivasi ini digunakan untuk menghitung probabilitas dari setiap kelas *target* keluaran. Pada arsitektur *LeNet-5*, *softmax* digunakan untuk permasalahan non-linearitas dengan banyak jumlah kelas, penentuan fungsi aktivasi pada bagian ini disesuaikan dengan jenis arsitektur dan jenis data yang digunakan [3].

## 2.6 Proses Perancangan CNN

Pada penelitian ini, metode CNN diterapkan dimulai dari tahap persiapan data, praproses data, pembagian data, perancangan arsitektur, *setting parameter*, pengujian *model* dan evaluasi yang dapat digambarkan dengan diagram alir berikut.



Gambar 5. Diagram alir penerapan CNN

Pada diagram alir Gambar 5, data citra daun tomat dilakukan praproses untuk mengubah ukuran dimensi citra. Kemudian, dilakukan pembagian data menjadi dua bagian yaitu data latih dan data uji. Dalam perancangan arsitektur dan model CNN, dilakukan *setting parameter* agar didapatkan akurasi yang diinginkan. Bila dalam pengujian didapatkan akurasi model > 75%, dilanjutkan dengan evaluasi. Namun bila akurasi model < 75%, dilakukan *setting parameter* ulang pada arsitektur CNN sampai didapatkan akurasi di atas 75%. Bagian-bagian per blok dapat dijelaskan sebagai berikut.

### 2.6.1 Data Citra Daun Tomat

Data citra daun tomat terdiri atas 4 jenis kelas. Penentuan jumlah kelas penyakit bergantung pada kondisi cuaca dan iklim karena menentukan jenis penyakit yang dapat ditemukan pada tanaman tomat. Berdasarkan hasil observasi pada saat pengumpulan data citra daun yang teridentifikasi penyakit, peneliti hanya dapat menemukan data daun sehat dan 3 jenis penyakit yaitu *Layu Fusarium*, *Tomato Crinivirus* dan *Tomato Yellow Leaf Curl*.

### 2.6.2 Praproses data

Praproses data dilakukan dengan mengubah ukuran atau *resize* dimensi citra asli dari ukuran (2448×3264) piksel menjadi (64×64) dan (32×32) piksel. Dalam penentuan ukuran dimensi citra, dapat disesuaikan dengan jenis data citra sesuai kebutuhan arsitektur yang akan digunakan. Proses mereduksi dimensi citra dilakukan karena pada CNN dimensi masukan citra harus memiliki ukuran yang sama, sehingga dengan melakukan reduksi dimensi dapat meminimalisir penggunaan memori dan waktu komputasi yang diperlukan.

### 2.6.3 Pembagian Data

Teknik pembagian data dalam penelitian ini dibagi menjadi dua bagian yaitu data latih dan data uji [4]. Dalam penelitian ini, dilakukan pembagian data citra menjadi data latih dan data uji dengan perbandingan 80:20 data. Jumlah data citra yang diambil sebanyak 2000 data citra yang terbagi dalam 4 kategori atau kelas sehingga terdapat 1600 data latih dan 400 data uji, sehingga pada setiap jenis kelas penyakit masing-masing terdapat 400 data citra latih dan 100 data citra uji.

### 2.6.4 Perancangan Arsitektur CNN

Arsitektur CNN yang digunakan dalam penelitian ini adalah jenis arsitektur LeNet-5. Arsitektur LeNet-5 merupakan arsitektur yang paling umum digunakan pada metode CNN karena memiliki tahapan dan aturan yang mudah dipahami dalam proses pembangunan model pengenalan pola pada citra digital secara sederhana. Untuk mendapatkan sebuah arsitektur CNN yang optimal dalam membangun sebuah model deteksi penyakit tomat pada citra daun, peneliti juga menggunakan sebuah arsitektur CNN yang telah dimodifikasi (*Custom*) sesuai dengan pemahaman yang diperoleh. Tujuan menggunakan dua jenis arsitektur yang berbeda ini diharapkan dapat melakukan perbandingan hasil akurasi terbaik dalam pembangunan model dan mendapatkan arsitektur CNN dengan proses komputasi yang optimal.

### 2.6.5 Setting Parameter

Parameter yang akan ditentukan meliputi jumlah ukuran *batch*, jenis *optimasi*, ukuran *patience* dan *learning rate*. Penentuan nilai parameter tidak memiliki konsensus, dengan kata lain tidak ada aturan pasti.

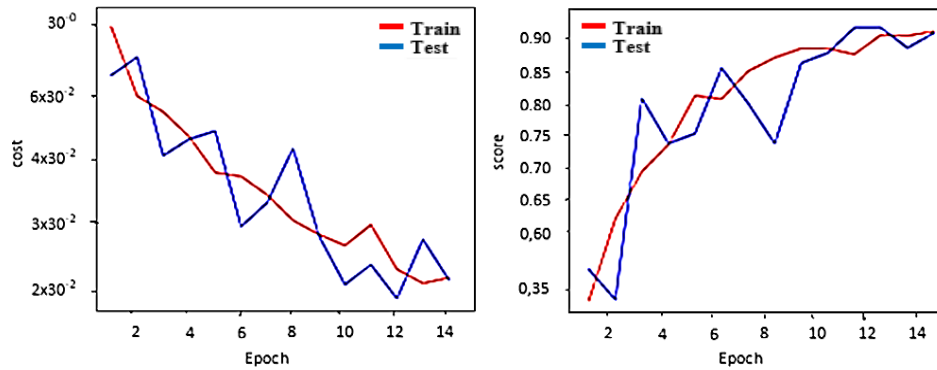
### 3. HASIL DAN PEMBAHASAN

#### 3.1 Hasil Penelitian

Berdasarkan hasil perancangan arsitektur CNN yang digunakan dan penentuan parameter, didapatkan hasil pelatihan model sebagai berikut.

##### 3.1.1 Hasil Training LeNet-5 Model

Hasil *training* LeNet-5 model dari data latih/*train* (merah) dan data uji/*test* (biru) dapat ditunjukkan pada grafik Gambar 6, di mana grafik sebelah kiri adalah nilai minimum galat (*cost*) dan grafik sebelah kanan adalah nilai akurasi (*score*) terbaik.



Gambar 6. Hasil *training* LeNet-5 model

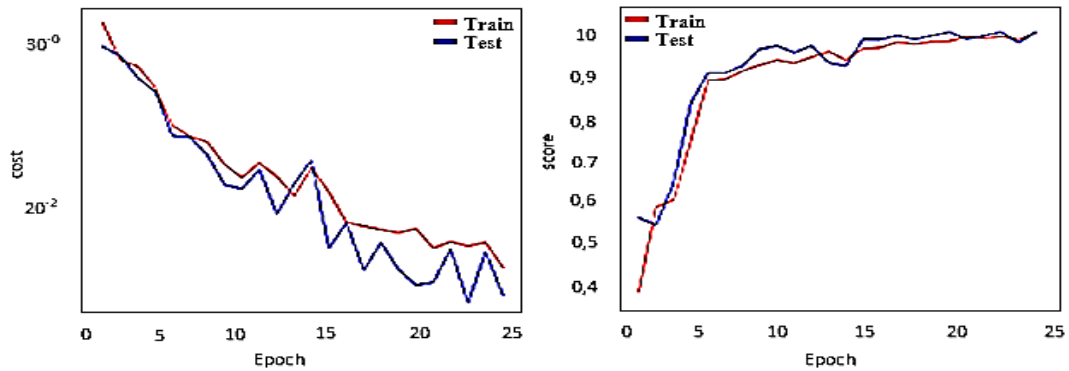
Dari grafik Gambar 6, dapat dilihat nilai akurasi (*score*) terbaik sebesar 91% dan nilai minimum galat (*cost*) terendah sebesar 0,21% pada data latih dan data uji dengan waktu *training* sebanyak 6 menit. Proses *training* berhenti secara otomatis dengan nilai *patience* 10, yaitu pada *epoch* ke 14. Visualisasi hasil *Training* LeNet-5 dapat ditunjukkan pada Tabel 1.

Tabel 1. Visualisasi hasil *training* LeNet-5

Penyakit	Olah	Citra asli (32x32)	Konvolusi pertama (28x28)	Pooling+ Tanh pertama (14x14)	Konvolusi kedua (10x10)	Pooling+ Tanh kedua (5x5)
<i>Layu Fusarium</i>						
<i>Crinivirus</i>						
<i>Curly top</i>						

##### 3.1.2 Hasil Training Custom Model

Hasil *training Custom* model dari data latih (merah) dan data uji (biru) dapat ditunjukkan pada grafik Gambar 7, di mana grafik sebelah kiri adalah nilai minimum galat (*cost*) dan grafik sebelah kanan adalah nilai akurasi (*score*) terbaik.



Gambar 7. Hasil *Training Custom Model*

Dari grafik Gambar 7, dapat dilihat nilai akurasi (*score*) terbaik sebesar 98% pada data latih dan data uji dan nilai minimum galat (*cost*) sebesar 0,029% dan menghabiskan waktu proses *training* selama 10 menit. Proses *training* berhenti secara otomatis pada *epoch* ke 25 dengan nilai *patience* 3. Visualisasi hasil *training Custom model* dapat ditunjukkan pada Tabel 2.

Tabel 2. Visualisasi hasil *training custom*

Penyakit	Olah	Citra asli (64x64)	Konvolusi+ ReLU Pooling	Konvolusi+ ReLU Pooling	Konvolusi+ ReLU Pooling	Konvolusi+ ReLU Pooling
<i>Layu Fusarium</i>						
<i>Crinivirus</i>						
<i>Curly top</i>						

Perbandingan nilai akurasi (*score*) dan nilai minimum galat (*cost*) hasil *training LeNet-5 Model* dengan *Custom Model* dapat ditunjukkan pada Tabel 3.

Tabel 3. Perbandingan nilai akurasi *LeNet-5 Model* dan *Custom Model*

Model	Akurasi ( <i>score</i> )	Galat ( <i>cost</i> )
<i>LeNet-5 Model</i>	91%	0,21%
<i>Custom Model</i>	98%	0,029%

### 3.1.3 Hasil *Testing LeNet-5 Model*

Untuk menentukan prediksi benar dan salah pada *LeNet-5 Model* dan *Custom Model* digunakan *confusion matriks* sebagaimana ditunjukkan pada Tabel 4 dan Tabel 5.

Tabel 4. *Confusion matriks LeNet-5 Model*

Matriks		Predict class			
		<i>Crinivirus</i>	<i>Curly top</i>	<i>Layufusarium</i>	<i>Healthy</i>
Actual Class	<i>Crinivirus</i>	20	0	0	0
	<i>Curly top</i>	0	19	2	4
	<i>Layufusarium</i>	0	0	17	0
	<i>Healthy</i>	0	0	0	16

Hasil prediksi benar terhadap data citra dengan kelas *Crinivirus* sebanyak 20, *Curly top* sebanyak 19 dengan prediksi salah sebanyak 1, *Layu fusarium* sejumlah 17 dengan prediksi salah sebanyak 3 gambar dan *Healthy* sebanyak 16 gambar dengan prediksi salah sejumlah 5 gambar. Perhitungan seluruh matriks pada Model LeNet-5 adalah sebagai berikut:

$$overall_{acc} = \frac{Total\ true\ prediction}{Total\ number\ of\ testing\ enteries} \times 100\% = \frac{72}{80} \times 100\% = 90\%$$

Berdasarkan pengujian menggunakan *confusion matriks* nilai akurasi pada model dengan arsitektur LeNet-5 menghasilkan nilai rata-rata (*score*) sebesar 90%. Kemudian, *Confusion matriks* Custom Model dapat ditunjukkan pada Tabel 5.

Tabel 5. *Confusion Matriks Custom Model*

Matriks		Predict class			
		<i>Crinivirus</i>	<i>Curly top</i>	<i>Layufusarium</i>	<i>Healthy</i>
Actual Class	<i>Crinivirus</i>	20	0	0	0
	<i>Curly top</i>	0	19	2	2
	<i>Layufusarium</i>	0	1	19	0
	<i>Healthy</i>	0	0	1	18

Prediksi menggunakan data baru pada *model* dengan arsitektur *Custom* menunjukkan hasil prediksi benar terhadap data citra baru dengan kelas *Crinivirus* sebanyak 20 gambar, prediksi benar pada kelas *Curly top* sebanyak 19 gambar dengan prediksi salah sebanyak 1 gambar, pada kelas *Layu fusarium* dengan prediksi benar sejumlah 19 gambar dan prediksi salah sebanyak 1 gambar, kemudian pada kelas *Healthy* prediksi benar sebanyak 18 gambar dengan prediksi salah sejumlah 2 gambar. Perhitungan seluruh matriks pada Model *Custom* adalah sebagai berikut:

$$overall_{acc} = \frac{Total\ true\ prediction}{Total\ number\ of\ testing\ enteries} \times 100\% = \frac{76}{80} \times 100\% = 95\%$$

Berdasarkan pengujian menggunakan *confusion matriks* nilai akurasi pada model dengan arsitektur Model *Custom* menghasilkan nilai rata-rata (*score*) sebesar 95%.

Keluaran *Convolutional Neural Network* pada proses deteksi penyakit tomat melalui citra daun ini selanjutnya akan dibandingkan dengan hasil penelitian Muhammad Islahfari Wahid et al. [15] yang memodelkan deteksi penyakit daun tomat dengan model arsitektur *Inception V4*. Perbandingan hasil menunjukkan bahwa metode *Convolutional Neural Network* dengan arsitektur *Custom* lebih besar nilai akurasinya yaitu sebesar 95% dibanding dengan model arsitektur *Inception V4* yaitu 90%.

#### 4. KESIMPULAN

Berdasarkan hasil pelatihan model dan pengujian menggunakan *Confusion matrix* pada Model *Convolutional Neural Network* yang telah dibangun menggunakan dua jenis arsitektur yang berbeda dan menggunakan parameter yang ditentukan pada proses *training*, dapat disimpulkan bahwa metode *Convolutional Neural Network* dapat diimplementasikan untuk mendeteksi 4 jenis kelas penyakit tomat melalui citra daun dengan nilai akurasi presisi untuk model jenis arsitektur LeNet-5 sebesar 90% dan untuk jenis arsitektur Custom sebesar 95%. Nilai akurasi yang dihasilkan pada Arsitektur LeNet-5 dapat dioptimalkan dengan cara menambahkan jumlah *convolution layer* dan menggunakan jenis dan nilai parameter yang berbeda. Bila dibandingkan dengan model arsitektur *Inception V4* pada penelitian sebelumnya, metode Custom yang dikembangkan memiliki nilai akurasi lebih besar. Dari hasil penelitian ini, diharapkan dapat menambah jumlah kelas penyakit pada daun tomat yang dapat dideteksi menggunakan *Convolutional Neural Network*. Untuk implementasi model, sistem dapat dikembangkan pada aplikasi berbasis web dan aplikasi *mobile*.

#### REFERENSI

- [1] D. Laily, "Deteksi Penyakit Pada Daun Tembakau dengan Menerapkan Algoritma Artificial Neural Network." *Simetris Journal Teknik Industri, Mesin, Elektro dan Ilmu Komputer*, vol. 3, no. 1, 2013. <https://doi.org/10.24176/simet.v3i1.88>
- [2] I. Goodfellow, Y. Bengio, and A. Courville, "Deep Learning (Adaptive Competition and Machine Learning)", The MIT Press, 2016.
- [3] Simonyan, Karen, and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv:1409.1556, 2014. <https://doi.org/10.48550/arXiv.1409.1556>
- [4] M. S. Wibawa, "Pengaruh Fungsi Aktivasi, Optimisasi dan Jumlah Epoch Terhadap Performa Jaringan Saraf Tiruan.", *Jurnal Sistem dan Informatika (JSI)*, vol. 11, no. 2, hal. 167-174, May 2017. <https://jsi.stikom-bali.ac.id/index.php/jsi/article/view/129>
- [5] Ruder, Sebastian. "An overview of gradient descent optimization algorithms." arXiv preprint arXiv:1609.04747, 2016. <https://doi.org/10.48550/arXiv.1609.04747>



- [6] Hinton, Geoffrey E. "A practical guide to *training* restricted Boltzmann machines." In *Neural networks: Tricks of the trade*, hal. 599-619. Springer, Berlin, Heidelberg, 2012.
- [7] Rahmadewi, Reni. "Identifikasi Jenis Tumbuhan Menggunakan Citra Daun Berbasis Jaringan Saraf Tiruan (Artificial Neural Networks)", *Media Elektro Journal*, vol.7, no.2, hal. 38-43, 2018. <https://doi.org/10.35508/jme.v0i0.427>
- [8] Rosiani, Ulla Delfana, Cahya Rahmad, Marcelina Alifia Rahmawati, and Frangky Tupamahu. "Segmentasi berbasis k-means pada deteksi citra penyakit daun tanaman jagung.", *Jurnal Informatika Polinema*, vol. 6, no. 3, pp. 37-42, 2020. <https://doi.org/10.33795/jip.v6i3.331>
- [9] Putri, Anglita Wigina. "Implementasi Artificial Neural Network (ANN) Backpropagation Untuk Klasifikasi Jenis Penyakit Pada Daun Tanaman Tomat.", *MATHunesa: Jurnal Ilmiah Matematika*, vol. 9, no. 2, hal. 344-350, 2021. <https://doi.org/10.26740/mathunesa.v9n2.p344-350>
- [10] Sitohang, Beriman, and Anita Sindar. "Analisis Dan Perbandingan Metode Sobel Edge Detection Dan Prewit Pada Deteksi Tepi Citra Daun Srilangka.", *Jurnal Nasional Komputasi dan Teknologi Informasi*, vol. 3, no. 3, hal. 314-322, 2020.
- [11] Solomon, Chris, and Toby Breckon. "Fundamentals of Digital Image Processing: A practical approach with examples in Matlab", John Wiley & Sons, 2011.
- [12] Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. "Deep learning", MIT press, 2016.
- [13] Alviansyah, Fahri, I. Ruslianto, and M. Diponegoro. "Identifikasi Penyakit Pada Tanaman Tomat Berdasarkan Warna Dan Bentuk Daun Dengan Metode Naive Bayes Classifier Berbasis Web.", *Coding Jurnal Komputer dan Aplikasi*, vol. 5, no. 1, hal. 23-32, 2017.
- [14] Albawi, Saad, Tareq Abed Mohammed, and Saad Al-Zawi. "Understanding of a convolutional neural network." In *2017 International conference on engineering and technology (ICET)*, hal. 1-6. IEEE, 2017. <https://doi.org/10.1109/ICEngTechnol.2017.8308186>
- [15] Wahid, M. Islahfari, S.A. Mustamin, and A. Lawi. "Identifikasi Dan Klasifikasi Citra Penyakit Daun Tomat Menggunakan Arsitektur Inception V4", *Proceeding KONIK (Konferensi Nasional Ilmu Komputer)*, vol. 5, hal. 257-264, 2021.