

# Identifikasi Penyakit Tanaman Padi Melalui Citra Daun Menggunakan DenseNet 201

## Identification of Rice Plant Diseases Through Leaf Image Using DenseNet 201

Primatua Sitompul<sup>1</sup>, Harly Okprana<sup>2</sup>, Annas Prasetyo<sup>3</sup>

<sup>1,3</sup> Program Studi Teknik dan Ilmu Komputer, Universitas Potensi Utama, Medan, Indonesia

<sup>2</sup> STIKOM Tunas Bangsa, Pematangsiantar, Indonesia

### Article Info

#### Genesis Artikel:

Diterima, 7 September 2022  
Direvisi, 20 September 2022  
Disetujui, 21 September 2022

#### Kata Kunci:

Identifikasi Penyakit  
Tanaman Padi  
DenseNet  
Identifikasi  
Machine Learning

### ABSTRAK

Indonesia merupakan negara agraria dengan mayoritas penduduk menggunakan beras sebagai bahan makanan pokok sangat tergantung terhadap produksi beras. Berkurangnya kuantitas dan kualitas produksi beras seringkali disebabkan oleh serangan penyakit pada tanaman padi yang terlambat diketahui. Hal tersebut disebabkan karena masih kurangnya penyuluhan yang membantu para petani cara mengidentifikasi dan menangani penyakit pada tanaman padi dengan cepat dan tepat. Penelitian ini melakukan eksperimen identifikasi penyakit tanaman padi berdasarkan citra daun menggunakan dataset yang memiliki empat penggolongan kondisi daun yang terkena penyakit, yaitu healthy, brown spot, hispa dan leaf blast. Identifikasi dilakukan dengan menggunakan arsitektur Densely Connected Convolutional Networks (DenseNet)-201 dan evaluasi berdasarkan nilai accuracy, precision, recall dan F1-Score menggunakan confusion matrix. Berdasarkan hasil evaluasi, diperoleh hasil yang cukup baik, yakni nilai akurasi pada data training sebesar 88.4% dan 82.99% pada data testing. Sementara nilai precision yang diperoleh sebesar 0.85%, recall 0.83% dan f1-score sebesar 0.83%. Dari hasil penelitian disimpulkan bahwa DenseNet201 cukup baik dan layak digunakan untuk melakukan identifikasi penyakit pada tanaman padi agar jenis penyakit yang menyerang dapat segera diketahui dan penanggulangan secara dini terhadap penyakit tanaman segera dapat dilakukan. Dengan demikian, ketahanan pangan dapat tetap terjaga dan tidak menyebabkan kerugian akibat panen yang merugikan para petani.

### ABSTRACT

Indonesia is an agricultural country with most of the population using rice as a staple food and is highly dependent on rice production. Late disease attacks on rice plants often cause a reduction in the quantity and quality of rice production. This is because there is still a lack of counseling that helps farmers identify and deal with diseases in rice plants quickly and accurately. This study conducted an experimental identification of rice plant diseases based on leaf imagery using a dataset with four classifications of leaf conditions affected by the disease: healthy, brown spot, hispa, and leaf blast. Identification is made using the Densely Connected Convolutional Networks (DenseNet)-201 architecture and evaluation based on the values of accuracy, precision, recall, and F1-Score using a confusion matrix. Based on the evaluation results, the results obtained are quite good; namely, the accuracy value of the training data is 88.4%, and 82.99% on the testing data. While the precision value obtained is 0.85%, recall is 0.83% and f1-score is 0.83%. From the results of the study, it was concluded that DenseNet201 is quite good and feasible to be used to identify diseases in rice plants so that the types of diseases that attack can be immediately identified and early prevention of plant diseases can be carried out immediately. Thus, food security can be maintained and not cause losses due to crop failures that harm farmers.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



### Penulis Korespondensi:

Primatua Sitompul,  
Program Studi Teknik dan Ilmu Komputer,  
Universitas Potensi Utama, Medan, Indonesia  
Email: prima.sitompul@gmail.com

## 1. PENDAHULUAN

Stabilitas kuantitas dan kualitas produksi beras merupakan aspek penting sebagai indikator ketahanan pangan bagi negara yang mayoritas masyarakatnya menjadikan beras sebagai bahan pangan pokok utama seperti negara Indonesia [1]–[4]. Produksi beras yang tidak seimbang dapat merembet pada permasalahan lain yang lebih serius dan bahkan dapat mempengaruhi keadaan ekonomi sebuah negara [5]. Pada masa penanaman, perawatan dan pengendalian penyakit tanaman merupakan bagian penting dan harus dilakukan dengan tepat agar tidak terjadi penurunan kuantitas dan kualitas saat panen. Diperkirakan sekitar 200.000-300.000 ton padi terserang hama setiap tahun di Indonesia. Faktor penyebabnya adalah kurangnya penyuluh pertanian, terbatasnya informasi dan pengetahuan petani dalam menangani masalah penyakit tanaman padi [6]. Keberhasilan pengendalian penyakit pada tanaman padi berarti menjaga produktifitas tanaman padi agar beras yang dihasilkan saat panen dapat maksimal.

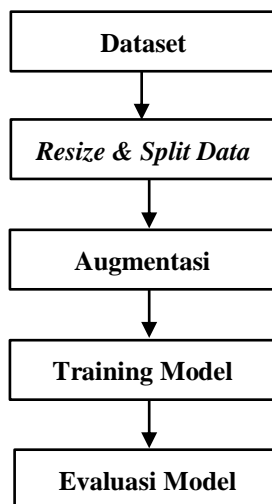
Penyakit pada tanaman biasanya disebabkan oleh bakteri dan jamur yang dapat mengakibatkan dampak negatif terhadap produksi dan kualitas tanaman [7]. Penyakit tanaman padi dapat dilihat dari perubahan fisik yang terdapat pada permukaan daun. Namun, apabila gejala-gejala sudah lanjut, penyakit pada tanaman akan sulit dibedakan [8]. Penyakit pada tanaman padi dapat disebabkan oleh kondisi cuaca, keadaan tanah, virus, jamur, atau hama lainnya [9]. Secara umum, terdapat empat jenis penyakit pada tanaman padi yang terdapat pada permukaan daun seperti: penyakit hawar daun, penyakit *blast*, penyakit hawar pada batang, penyakit kresak, dan lainnya. Masing-masing penyakit mengharuskan penanganan yang berbeda, jika hal tersebut terlambat untuk disadari dan tidak ditangani dengan benar maka dapat menyebabkan kerugian berupa gagal panen yang berimbas pada penurunan produksi tanaman padi itu sendiri [6].

*Convolutional Neural Network* telah menjadi pendekatan *machine learning* yang dominan untuk pengenalan objek visual [10] dan mendapat perhatian dibidang pertanian seperti deteksi tanaman, deteksi buah, identifikasi penyakit, deteksi gulma, pengenalan hama, dan lain-lain [7], [11]. *Densely Connected Convolutional Networks (DenseNet)* dikembangkan secara khusus untuk meningkatkan akurasi yang disebabkan oleh gradien yang hilang di jaringan saraf tingkat tinggi karena jarak yang jauh antara lapisan *input* dan *output* & informasi menghilang sebelum mencapai tujuannya [10].

Pada penelitian ini, proses identifikasi penyakit tanaman padi secara otomatis berdasarkan citra daun dilakukan dengan menggunakan *image processing* dan *DenseNet201*. Tujuan dari hasil identifikasi secara otomatis adalah menentukan apakah tanaman masih dalam keadaan sehat/*healthy* atau sudah terkena penyakit *brown spot*, *hispa* atau *leaf blast*. Sistem identifikasi penyakit tanaman padi secara otomatis diharapkan dapat membantu sektor pertanian dalam perawatan tanaman padi, sehingga menghemat banyak biaya, waktu, tenaga, dan terjaganya stabilitas kuantitas dan kualitas produksi beras.

## 2. METODE PENELITIAN

Tahapan proses identifikasi penyakit tanaman berdasarkan citra daun padi dari awal sampai akhir yang dilakukan pada penelitian ini diilustrasikan dengan diagram alir yang ditunjukkan pada gambar 1 berikut.



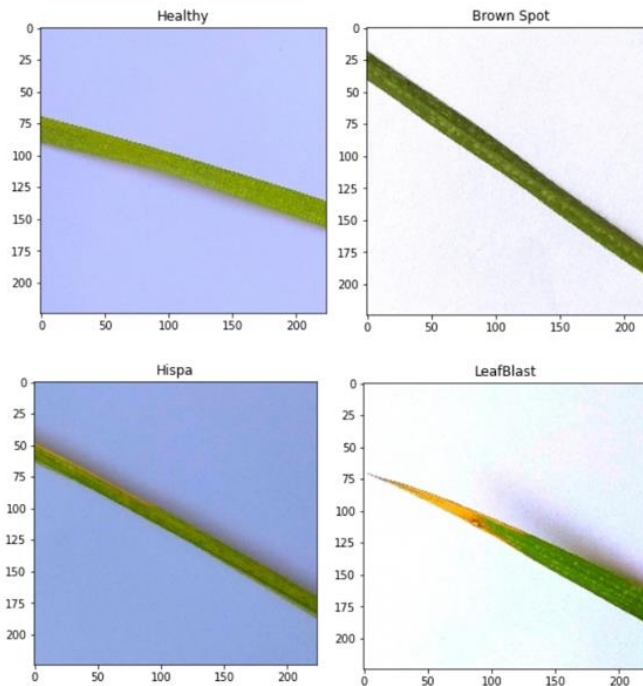
Gambar 1. Diagram alur penelitian

Penelitian dilakukan dengan menggunakan tools *jupyter notebook* dengan bahasa *python* dengan spesifikasi perangkat keras menggunakan *processor* AMD Ryzen 7, 2.9GHz, RAM 8GB, NVIDIA GTX1660Ti

### 2.1. Dataset

Dataset diperoleh dari situs *Kaggle* dengan judul *Rice Leaf* yang bersumber dari *UCI Machine Learning Repository* [12]. Pada dataset terdapat empat folder yang menggolongkan citra daun tanaman padi berdasarkan penyakitnya dengan format file ‘.jpg’. Setiap citra hanya memiliki photo satu daun dengan *background* putih akan tetapi karena faktor pencahayaan, beberapa

background terlihat berwarna biru. Sampel citra penyakit pada daun tanaman padi yang terdapat pada dataset dapat dilihat pada gambar 2.



Gambar 2. Sampel citra penyakit tanaman pada daun padi

Total data citra yang terdapat di dalam dataest sebanyak 3.355 citra, yang terdiri dari empat kelas penyakit, yaitu *healthy* dengan jumlah data sebanyak 1.488 citra, *brown spot* dengan jumlah data sebanyak 523 citra, *hispa* dengan jumlah data sebanyak 565 citra, dan terakhir *leaf blast* dengan jumlah data sebanyak 779 citra. Tabel distribusi citra pada dataset dapat dilihat pada tabel 1.

Tabel 1. Distribusi citra pada dataset

Kelas	Jumlah citra	Format
<i>Healthy</i>	1.488	*.jpg
<i>Brown spot</i>	523	*.jpg
<i>Hispa</i>	565	*.jpg
<i>Leaf blast</i>	779	*.jpg
<b>TOTAL</b>	<b>3.355</b>	

## 2.2. Resizing & Split Data

Dikarenakan ukuran asli citra pada dataset bervariasi dan cukup besar, pada penelitian ini langkah pertama yang dilakukan adalah melakukan perubahan ukuran citra. Seluruh citra direduksi menjadi 224×224 piksel. Selanjutnya dilakukan pemisahan data citra yang akan digunakan untuk *training* dan data citra untuk *testing* pada saat evaluasi model. Dataset dibagi menjadi tiga bagian, yakni data training, data validasi dan data testing. Data citra untuk training dipisah dengan rasio 70%, data citra untuk validasi dipisah dengan rasio 20%, sementara data citra untuk testing dipisah sebesar 10%. Dari hasil pemisahan dataset, alokasi jumlah data untuk training adalah sebanyak 2.348 citra, jumlah data untuk validasi adalah sebanyak 672 citra, dan jumlah data untuk testing sebanyak 335 citra. Distribusi dataset hasil *split data* dapat dilihat pada tabel 2.

Tabel 2. Distribusi citra hasil *split data*

	Kelas	Jumlah	Jumlah
<i>Data training</i>	<i>Healthy</i>	1.040	2.348
	<i>Brown spot</i>	366	
	<i>Hispa</i>	379	
	<i>Leaf blast</i>	563	
<i>Data validasi</i>	<i>Healthy</i>	289	672
	<i>Brown spot</i>	110	
	<i>Hispa</i>	131	
	<i>Leaf blast</i>	142	
<i>Data testing</i>	<i>Healthy</i>	159	335
	<i>Brown spot</i>	47	
	<i>Hispa</i>	55	
	<i>Leaf blast</i>	74	
<b>TOTAL</b>		<b>3.355</b>	<b>3.355</b>

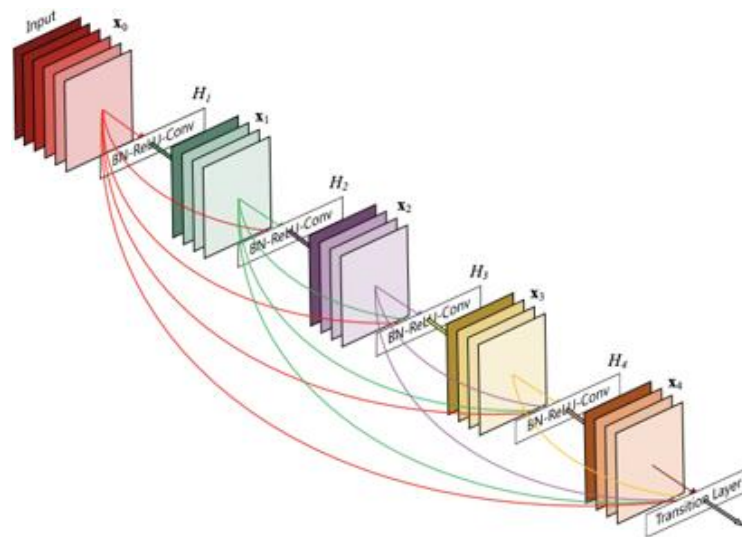
### 2.3. Augmentasi Data

Augmentasi merupakan proses pengolahan citra dengan teknik duplikasi dan modifikasi citra. Proses augmentasi data bertujuan untuk meningkatkan jumlah variasi citra sehingga memperluas dataset pelatihan [6], [13]. Augmentasi pada penelitian ini memanfaatkan fitur *preprocessing image* dari *Tensorflow* yakni *ImageDataGenerator*. *ImageDataGenerator* bekerja dengan melakukan regenerasi citra berdasarkan parameter yang sudah ditentukan.

### 2.4. Training Model

#### 1. Arsitektur DenseNet201

*Dense Convolutional Network (DenseNet)* merupakan salah satu model arsitektur *Deep Learning* yang menghubungkan setiap *layer* beserta *feature-maps* ke seluruh *layer* berikutnya. *Layer* berikutnya akan menerima *input feature-maps* dari seluruh *layer* sebelumnya. *DenseNet* menghubungkan setiap lapisan/blok ke setiap lapisan/blok lainnya dengan cara umpan maju. Sedangkan jaringan konvolusional tradisional dengan  $L$  lapisan memiliki koneksi  $L - 1$  antara setiap lapisan dan lapisan berikutnya jaringan memiliki koneksi langsung  $L(L + 1)/2$ . *DenseNet* memiliki beberapa keunggulan menarik: meringankan masalah gradien-gradien, memperkuat penyebaran fitur, mendorong penggunaan kembali fitur, dan secara substansial mengurangi jumlah parameter [14].



Gambar 3. Arsitektur DenseNet [10]

Penelitian ini menggunakan arsitektur *DenseNet201* sebagai *base model* dengan menggunakan *weights* dari “*imagenet*” dan penambahan beberapa *layer* pada lapisan selanjutnya. Penambahan lapisan dimaksud adalah *average pooling layer*, *dropout layer* dan *dense layer* dan *activation function*. *Average pooling* mengembalikan nilai rata-rata dari bagian gambar yang dicakup oleh kernel [15]. Sementara penggunaan *dropout* telah terbukti dapat meningkatkan akurasi dan juga dapat membantu menghindari terjadinya *overfitting* pada penelitian [6].

#### 2. Optimizer, Loss & Hyperparameter

Pada penelitian ini, “*Adam*” digunakan sebagai *optimizer*, dan *loss function* yang digunakan adalah *categorical\_crossentropy*. *Adam* adalah algoritma optimasi *objective function* yang sering dipergunakan pada *deep neural network*. Perubahan parameter *Adam* tergantung pada *gradien*, *learning rate*, nilai momen pertama dan nilai momen kedua dari *gradien*. *Cross entropy* digunakan untuk mengukur seberapa besar simpangan atau jarak antara  $p$  dan  $q$  pada distribusi  $p$  dan  $q$ . *Cross entropy* mengukur hasil yang diprediksi oleh model dibandingkan dengan hasil yang diharapkan. *Hyperparameter* adalah parameter yang dapat disesuaikan secara manual untuk mengontrol proses pelatihan model. Setting *hyperparameter* dilakukan dengan merubah nilai yang terdapat pada *learning rate* dan *batch size*.

### 2.5. Evaluasi Model

Hasil yang telah diperoleh pada tahap training model selanjutnya di evaluasi untuk mengetahui performa model dalam mengidentifikasi penyakit tanaman padi melalui citra daun. Pada tahap evaluasi, performa model diukur dari nilai *metric score* yang diperoleh berdasarkan *confusion matrix*. Berdasarkan *confusion matrix* dapat dilihat *precision*, *recall* dan *f1-score* yang diperoleh pada tahap evaluasi yang menggambarkan tingkat keberhasilan model dalam mengidentifikasi penyakit tanaman padi berdasarkan citra daun.

#### 1. Confusion Matrix

*Confusion Matrix* digunakan untuk merepresentasikan hasil prediksi dengan kondisi sebenarnya dari dataset yang telah dilatih menggunakan arsitektur *DenseNet*. *Confusion matrix* dapat dilihat pada tabel 3 berikut.

Tabel 1. *Confussion Matrix*

		Prediksi	
		Positif	Negatif
Aktual	Positif	<i>True Positive (TP)</i>	<i>False Negative (FN)</i>
	Negatif	<i>False Positive (FP)</i>	<i>True Negative (TN)</i>

*True Positive (TP)* merupakan kondisi dimana model memprediksi data ada di kelas Positif dan yang sebenarnya data memang ada di kelas Positif. *False Positive (FP)* merupakan kondisi dimana model memprediksi data berada di kelas positif, tetapi sebenarnya data ada di kelas negatif. *True Negative (TN)* merupakan kondisi dimana model memprediksi data ada di kelas negatif dan sebenarnya data memang berada di kelas negatif. Sementara *False Negative (FN)* merupakan kondisi dimana model memprediksi data ada di kelas negatif, namun yang sebenarnya data ada di kelas positif.

## 2. Accuracy

*Accuracy* merupakan *metric* yang digunakan untuk menghitung rasio prediksi benar (positif dan negatif) dengan keseluruhan data saat melakukan pelatihan. *Metric accuracy* dirumuskan pada persamaan (1).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

Dimana : *TP* merupakan *True Positive*, *TN* merupakan *True Negative*, *FP* merupakan *False Positive* dan *FN* merupakan *False Negative*

## 3. Precision

*Precision* digunakan untuk menghitung presisi atau data prediksi positif dari keseluruhan hasil yang diprediksi positif. *Precision* dirumuskan pada persamaan (2).

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

Dimana : *TP* merupakan *True Positive*, *FP* merupakan *False Positive*

## 4. Recall

*Recall* digunakan untuk menghitung jumlah prediksi pada kelas yang sama. Secara definisi *recall* adalah perbandingan antara *True Positive (TP)* dengan banyaknya data yang sebenarnya positif. Rumus *Recall* dituliskan dalam persamaan (3).

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

Dimana : *TP* merupakan *True Positive* dan *FN* merupakan *False Negative*

## 5. F1-Score

*F1-Score* merupakan *harmonic-mean* atau perbandingan rata-rata *precision* dan *recall* yang dibobotkan. Rentang nilai *F1 Score* adalah 0 sampai 1 dengan nilai terburuk adalah 0 dan nilai terbaik adalah 1. *F1-Score* mengindikasikan bahwa model klasifikasi memiliki *precision* dan *recall* yang baik. dan nilai *F1 Score* dirumuskan pada persamaan (4).

$$F1\ Score = 2 * (Recall * Precision) / (Recall + Precision) \quad (4)$$

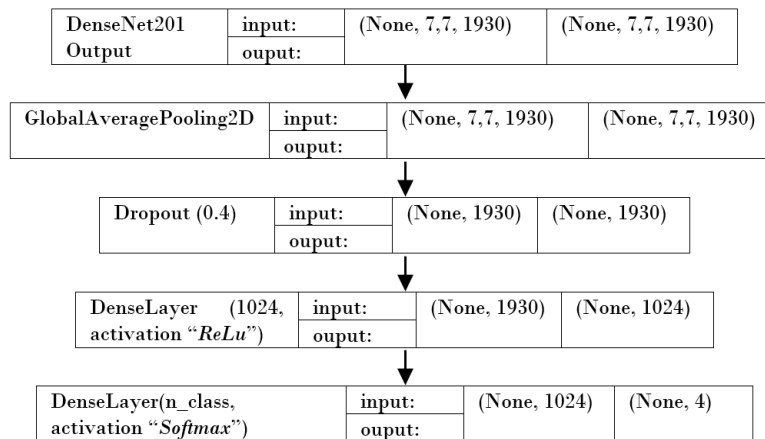
## 3. HASIL DAN PEMBAHASAN

### 3.1. Hasil Pelatihan Model

Setelah tahap *resizing* dan *split data* dilakukan, tahap selanjutnya adalah melakukan augmentasi data. Augmentasi pada penelitian ini memanfaatkan fitur *ImageDataGenerator* dengan nilai *shear\_range=0.3*, *brightness\_range=0.2-1.0*, *rotation\_range=0.2*, *zoom\_range=0.2*, *horizontal\_flip=True*, *vertical\_flip=True*. Parameter diterapkan baik untuk data training dan data validasi. Fitur *ImageDataGenerator* akan terus meregenerasi citra hasil augmentasi baru pada saat proses training berlangsung.

Selanjutnya, proses berikutnya adalah melakukan proses pelatihan. Pada penelitian ini arsitektur *DenseNet201* digunakan bersamaan dengan *weights* dari "imagenet". Karena pelatihan menggunakan *weights* yang sudah mengalami pelatihan sebelumnya, maka *top layer* pada arsitektur *DenseNet201* tidak diikutsertakan dalam pelatihan. Parameter selanjutnya adalah dengan menambahkan beberapa lapisan setelah *output*. Untuk mengetahui jumlah lapisan yang ideal dengan arsitektur *DenseNet201* pada identifikasi penyakit tanaman padi melalui citra daun, dilakukan eksperimen-eksperimen dengan merubah nilai *batch size*, *learning rate*, pemilihan *optimizer*, *batch normalization* dan jumlah *dense layer* yang digunakan.

Berdasarkan hasil eksperimen yang dilakukan, penambahan *layer* pada model, perubahan nilai *learning rate*, pemilihan *optimizer* dan memberikan pengaruh yang sangat signifikan terhadap hasil akurasi. Model yang paling ideal adalah dengan penambahan *GlobalAveragePooling2D*, *Dropout layer* dengan nilai 0.4 dan *Dense layer* dengan nilai 1024 dengan fungsi aktivasi "ReLU". Sementara pada lapisan output ditambahkan empat layer sesuai dengan jumlah kelas untuk prediksi dengan fungsi aktivasi "softmax" arsitektur model dapat dilihat pada gambar 4. Jumlah empat kelas pada training ditentukan berdasarkan jumlah kelas penyakit tanaman padi pada dataset yang terdiri dari *healthy*, *brown spot*, *hispa* dan *leaf blast*.



Gambar 4. Arsitektur model dengan penambahan layer

Nilai *epoch* pada eksperimen yang dilakukan adalah 150 dengan nilai *batch size* 8. Optimizer *Adam* dipilih karena dari beberapa percobaan yang dilakukan menghasilkan nilai akurasi terbaik dibanding optimizer lainnya. Nilai *learning rate* terbaik yang telah dicoba adalah 0.0001. Pada fitur *callbacks* ; *early stopping* dan *learning rate reduction* yang tersedia pada library *keras* juga digunakan pada proses training. Nilai *patience* pada fitur *early stopping* sebesar 35, sementara pada *learning rate reduction* nilai *patience* adalah 10 dan nilai *factor* = 0.5, *monitor* = “*val\_accuracy*”.

Berdasarkan hasil training yang dilakukan, diperoleh bahwa nilai akurasi pada data training adalah 0.8884, dengan nilai *loss* 0.3261. Sementara nilai akurasi pada data validasi adalah 0.8006 dengan *loss* 0.6061. Dan akurasi pada data testing sebesar 0.8299 dengan nilai *loss* 0.4954. Grafik hasil training dapat dilihat pada gambar 5.



Gambar 5. Grafik *loss* dan *accuracy* pada saat training

Untuk melihat hasil prediksi atau identifikasi penyakit tanaman pada setiap citra dilakukan pemeriksaan terhadap *score* yang diperoleh berdasarkan prediksi model. Dari pemeriksaan nilai *score* prediksi yang diperoleh untuk setiap citra pada data testing, dominasi nilai *score* tertinggi diperoleh pada citra dengan jenis *leafblast* dan *brownspot*. Pengamatan pada citra dengan *score* yang rendah, prediksi jenis penyakit cenderung salah. Pada tabel 4 dapat dilihat data citra pada data testing berdasarkan nama file dan prediksi yang diperoleh.

Tabel 2. *Score* prediksi masing-masing citra pada data testing

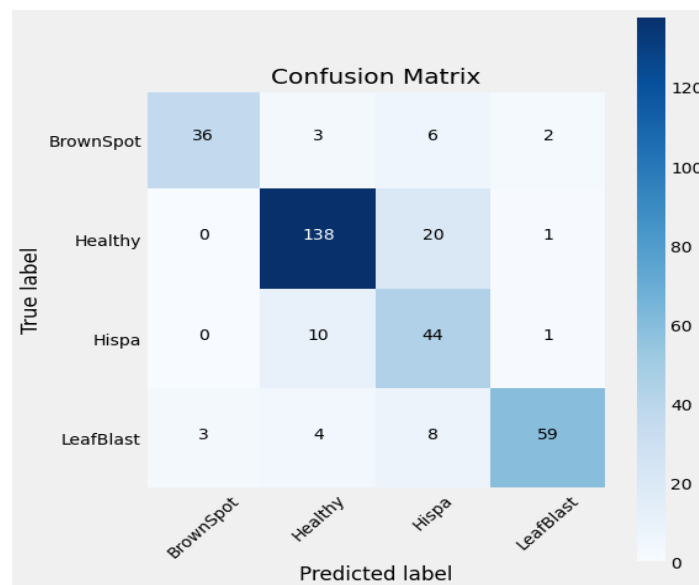
No	Kelas>Nama File Citra	Prediksi	Score
1	LeafBlast\IMG_20190419_173204.jpg	LeafBlast	0.99979013
2	LeafBlast\IMG_4939.jpg	LeafBlast	0.9992576
3	LeafBlast\IMG_5519.jpg	LeafBlast	0.9992204
4	BrownSpot\IMG_20190421_200111.jpg	BrownSpot	0.9991192
5	LeafBlast\IMG_20190420_194957.jpg	LeafBlast	0.99864155
6	LeafBlast\IMG_4846.jpg	LeafBlast	0.99850035
7	LeafBlast\IMG_4769.jpg	LeafBlast	0.99835604
8	LeafBlast\IMG_20190419_163411.jpg	LeafBlast	0.99830914
9	BrownSpot\IMG_20190420_195219.jpg	BrownSpot	0.9978036
10	LeafBlast\IMG_4852.jpg	LeafBlast	0.9977245
...	.....	.....	.....
326	Healthy\IMG_20190419_095745.jpg	LeafBlast	0.45516652

No	Kelas\Nama File Citra	Prediksi	Score
327	Healthy\IMG_20190419_102311.jpg	Healthy	0.451563
328	Healthy\IMG_20190419_141410.jpg	Healthy	0.45147452
329	Healthy\IMG_20190424_130352.jpg	BrownSpot	0.4445496
330	BrownSpot\IMG_20190420_193445.jpg	Healthy	0.435715
331	Healthy\IMG_20190419_134816.jpg	Hispa	0.4334215
332	LeafBlast\IMG_20190419_170811.jpg	LeafBlast	0.4220818
333	Healthy\IMG_20190424_132240.jpg	Hispa	0.37545958
334	Hispa\IMG_20190424_125824.jpg	BrownSpot	0.36609963
335	Hispa\IMG_20190419_105733.jpg	LeafBlast	0.34396672

### 3.2. Hasil Pengujian Model

Untuk mempermudah evaluasi model dan hasil training, dilakukan visualisasi dengan menggunakan *confusion matrix*. Berdasarkan tabel *confusion matriks* ditemukan bahwa dari 47 data citra penyakit brown spot, 36 diantaranya berhasil diidentifikasi dengan benar. Tetapi 11 diantaranya tidak berhasil diprediksi dengan baik. Sebanyak 3 citra diprediksi sebagai *healthy*, sebanyak 6 citra diprediksi sebagai *hispa*, dan 2 citra diprediksi sebagai *leaf blast*.

Pada 159 data citra daun tanaman padi sehat yang digunakan sebagai data testing, 138 diprediksi dengan benar *healthy*, 20 data citra diprediksi sebagai *hispa* dan 1 diprediksi sebagai *leafblast*. Dari 55 data citra penyakit *hispa* yang digunakan sebagai data testing, 44 diprediksi benar, 10 data citra diprediksi sebagai *healthy*, dan 1 diprediksi sebagai *leafblast*. Sementara dari 74 data citra *leaf blast* yang digunakan sebagai data testing, 59 diprediksi benar, 8 diantaranya diprediksi *brownspot*, 4 diprediksi *healthy*, 8 diprediksi *hispa*. Tabel *Confusion matrix* dapat dilihat pada gambar 6.



Gambar 6. Tabel *Confusion Matrix*

Dengan mengamati hasil visualisasi *confussion matrix* pada gambar 6, dapat dilihat bahwa persentase prediksi yang paling sedikit diperoleh pada jenis penyakit *brownspot* karena dari 47 sampel yang digunakan, 36 diprediksi benar, sementara 11 citra tidak diprediksi dengan baik. Untuk dapat melihat lebih detail performa model dalam memprediksi jenis tanaman padi, dilakukan pengamatan pada *classification report* untuk melihat nilai *precision*, *recall* dan *f1-score*. *Precision* merupakan parameter yang digunakan untuk mengukur ketepatan model dalam memprediksi jenis penyakit tanaman padi. *Recall* atau *sensifitas* adalah parameter yang digunakan untuk mengetahui performa model dalam menghitung persentase jumlah data citra yang terindikasi penyakit dengan benar dari jumlah data citra yang telah diprediksi. Berdasarkan tabel *classification report* diperoleh nilai *precision* sebesar 85%, *recall* 83% dan *f1-score* sebesar 83%. *Classification report* dapat dilihat pada gambar 7.

	precision	recall	f1-score	support
BrownSpot	0.92	0.77	0.84	47
Healthy	0.89	0.87	0.88	159
Hispa	0.56	0.80	0.66	55
LeafBlast	0.94	0.80	0.86	74
accuracy			0.83	335
macro avg	0.83	0.81	0.81	335
weighted avg	0.85	0.83	0.83	335

Gambar 7. Classification Report.

#### 4. KESIMPULAN

Pada percobaan yang telah dilakukan untuk mengidentifikasi penyakit tanaman padi berdasarkan citra daun menggunakan model *DenseNet201*, diperoleh nilai akurasi sebesar 92.59% pada data *training* dan nilai akurasi sebesar 82.99% pada data *testing*. Evaluasi model dengan menggunakan *confussion matriks* diperoleh nilai *precision* sebesar 0.85%, *recall* 0.83% dan *f1-score* sebesar 0.83%. Nilai akurasi pada data *testing* menunjukkan kemampuan model memprediksi citra baru penyakit tanaman padi cukup baik. Dari hasil penelitian disimpulkan bahwa *DenseNet201* cukup baik dan layak digunakan untuk melakukan identifikasi penyakit pada tanaman padi. Jumlah *dense layer*, *drop out* dan *learning rate* memberikan pengaruh yang besar terhadap nilai akurasi pada saat eksperimen. Pada penelitian ini masih terdapat keterbatasan, salah satunya masih belum menerapkan *image pre-processing* atau *segmentasi* dan jumlah dataset yang digunakan masih terbatas pada empat kelas. Pada penelitian selanjutnya dapat mencoba melakukan eksperimen dengan dataset yang berbeda atau dengan penambahan kelas penyakit pada tanaman padi untuk mendapatkan hasil identifikasi yang lebih baik agar jenis penyakit yang menyerang dapat segera diketahui dan diberikan penanggulangan secara dini agar tidak menyebabkan kerugian akibat gagal panen yang merugikan para petani.

#### REFERENSI

- [1] A. A. Fardhani, D. Insani, N. Simanjuntak, and A. Wanto, "Prediksi Harga Eceran Beras Di Pasar Tradisional Di 33 Kota Di Indonesia Menggunakan Algoritma Backpropagation," *Jurnal Infomedia*, vol. 3, no. 1, pp. 25–30, 2018.
- [2] Sudirman, A. P. Windarto, and A. Wanto, "Data mining tools | rapidminer: K-means method on clustering of rice crops by province as efforts to stabilize food crops in Indonesia," *IOP Conference Series: Materials Science and Engineering*, vol. 420, no. 1, pp. 1–8, 2018.
- [3] J. Nijar and T. Abbas, "Faktor-faktor yang Mempengaruhi Impor Beras di Indonesia Tahun 2001-2019," *E-Journal Perdagangan Industri dan Moneter*, vol. 9, no. 3, pp. 181–192, 2021.
- [4] S. Nurkasanah, A. Prasetyo, and M. B. Setyawan, "Implementasi Logika Fuzzy untuk prediksi hasil Panen Padi dengan Metode Tsukamoto," *Jurnal Rekayasa Teknologi dan Komputasi*, vol. 1, no. 1, pp. 1–11, 2022.
- [5] M. A. P. Hutabarat, M. Julham, and A. Wanto, "Penerapan Algoritma Backpropagation Dalam Memprediksi Produksi Tanaman Padi Sawah Menurut Kabupaten/Kota di Sumatera Utara," *Jurnal semanTIK*, vol. 4, no. 1, pp. 77–86, 2018.
- [6] Ulfah Nur Oktaviana, Ricky Hendrawan, Alfian Dwi Khoirul Annas, and Galih Wasis Wicaksono, "Klasifikasi Penyakit Padi berdasarkan Citra Daun Menggunakan Model Terlatih Resnet101," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 6, pp. 1216–1222, 2021.
- [7] S. M. Hassan and A. K. Maji, "Plant Disease Identification Using a Novel Convolutional Neural Network," *IEEE Access*, vol. 10, pp. 5390–5401, 2022.
- [8] E. Anggiratih, S. Siswanti, S. K. Octaviani, and A. Sari, "Klasifikasi Penyakit Tanaman Padi Menggunakan Model Deep Learning Efficientnet B3 dengan Transfer Learning," *Jurnal Ilmiah SINUS*, vol. 19, no. 1, p. 75, 2021.
- [9] I. Y. Purbasari, B. Rahmat, and C. S. Putra PN, "Detection of Rice Plant Diseases using Convolutional Neural Network," *IOP Conference Series: Materials Science and Engineering*, vol. 1125, no. 1, p. 012021, 2021.
- [10] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-January, pp. 2261–2269, 2017.
- [11] F. Masykur, M. B. Setyawan, and K. Winangun, "Optimalisasi Epoch Pada Klasifikasi Citra Daun Tanaman Padi Menggunakan Convolutional Neural Network ( CNN ) MobileNet t Epoch Optimization on Rice Leaf Image Classification Using Convolutional Neural Network ( CNN ) MobileNet," no. July, pp. 581–590, 2022.
- [12] "Rice Leaf Dataset."
- [13] N. Sakinah, T. Badriyah, and I. Syarif, "Analisis Kinerja Algoritma Mesin Pembelajaran untuk Klarifikasi Penyakit Stroke Menggunakan Citra CT Scan," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 7, no. 4, p. 833, 2020.
- [14] J. Pardede and D. A. L. Putra, "Implementasi DenseNet Untuk Mengidentifikasi Kanker Kulit Melanoma," *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 6, no. 3, pp. 425–433, 2020.
- [15] M. R. Alwanda, R. P. K. Ramadhan, and D. Alamsyah, "Implementasi Metode Convolutional Neural Network Menggunakan Arsitektur LeNet-5 untuk Pengenalan Doodle," *Jurnal Algoritme*, vol. 1, no. 1, pp. 45–56, 2020.