

KLASIFIKASI PENYAKIT DAUN SAWI HIJAU DENGAN METODE CNN

Errin Dwi Ratnasari¹, Dhira Ananta Rudira², Anom Surya Buana³

^{1,2,3}Teknik Informatika, Fakultas Teknik dan Ilmu Komputer, Universitas Nusantara PGRI Kediri

E-mail: ¹[*¹errindwi@gmail.com](mailto:errindwi@gmail.com), ²Anantadhira@gmail.com, ³anom3040@gmail.com

Abstrak – Penelitian ini mengusulkan penggunaan *Convolutional Neural Network (CNN)* untuk mengklasifikasikan penyakit terhadap daun sawi hijau. Sistem ini bekerja secara otomatis untuk mengontrol mana tanaman sehat dan mana tanaman sakit. Penelitian ini bertujuan untuk mendeteksi tanaman sakit dan tanaman sehat pada tanaman daun sawi hijau yang bekerja dengan pengambilan gambar secara langsung melalui digital. Tingkat akurasi daun sawi hijau yang tepat menjadi penting dalam industri tanaman daun sawi hijau. CNN merupakan metode *Deep Learning* yang dapat memproses citra dengan akurasi tinggi. Dataset berisi 1071 gambar daun sawi hijau dengan empat tingkat akurasi daun sawi hijau, yaitu dark akurasi, green akurasi, medium akurasi dan light akurasi. Proses training dilakukan dengan *Visual Studio Code* dan library *TensorFlow, Keras, Open CV* dan *Numpy*. Hasil evaluasi menunjukkan akurasi mencapai 99% pada epoch ke 10. Sistem ini diharapkan membantu penyuka daun sawi hijau mengenal tingkat akurasi daun sawi hijau dengan lebih akurat.

Kata Kunci — CNN, Klasifikasi, Tingkat akurasi, daun sawi, *Deep Learning*

1. PENDAHULUAN

Daun sawi hijau adalah daun paling populer di dunia dan banyak diminati. Selain itu pengetahuan tentang manfaat memakan daun sawi hijau bagi kesehatan menjadikan daun sawi hijau sebagai *ekspor* penting bagi Indonesia. Menurut data statistik hortikultura Daerah Istimewa Yogyakarta luas panen sayuran potensi tanaman sawi pada tahun 2016 adalah 539 hektar dan tahun 2017 adalah 438 hektar artinya luas panen sawi mengalami penurunan sebesar 101 hektar (18,74%). Sedangkan pada tahun 2018 menjadi 397 hektar artinya luas panen sawi mengalami penurunan lagi sebesar 41 hektar (9,36%). Pada tahun 2017, secara umum produksi tanaman sayur-sayuran mengalami penurunan yang sangat signifikan. Diantaranya sawi dari jumlah produksi tahun 2016 sebesar 39.105, ditahun 2017 turun menjadi 36.305, dan ditahun 2018 turun menjadi 32.047. Kondisi dimana semakin berkurangnya lahan pertanian terutama di daerah perkotaan mendorong berbagai macam teknologi dan inovasi dan dalam mengembangkan bidang pertanian.[1]

Daun sawi memiliki berbagai manfaat dan kegunaan dalam kegiatan sehari-hari. Selain untuk bahan makanan sayuran, sawi juga dapat dimanfaatkan sebagai sarana pengobatan. Terdapat beberapa jenis hama sayuran sawi diantaranya ialah hama belalang kecil, hama ulat grayak (*Spodoptera litura*), ulat tritip (*Plutela xylostella*), penggerek pucuk (*Crociodolomia pavonana*) dan ulat jengkal (*Plusia sp.*). Dimana tingkat kerusakan yang dihasilkan oleh hama-hama tersebut antara 10 sampai 25%, bahkan ada yang sampai gagal panen. [2]

Deep Learning (Pembelajaran Dalam) merupakan Teknik *Machine Learning* yang berbasis Jaringan Syaraf Tiruan (JST) atau evolusi dari JST. Salah satu algoritma *Deep Learning* yang merupakan pengembangan dari *Multi Layer Perceptron (MLP)* adalah *Convolutional Neural Network (CNN)*. [3] Penelitian menggunakan metode *Convolutional Neural Network* pernah dilakukan oleh Umami Sri Rahmadhani, Noveri Lysbetti Marpaung (2023). Penelitian ini mengklasifikasi objek Jamur berdasarkan *genusnya* dengan metode CNN. Dataset yang digunakan berjumlah 1200 citra dengan perbandingan data uji dan data latih yaitu 30:70. Data tersebut akan dibagi menjadi enam kelas, maka data latih yang digunakan pada setiap kelas berjumlah 840 sedangkan untuk data uji berjumlah 360 data. Hasil dari persentase keberhasilan rata-rata dari penelitian tersebut sebesar 76%. [4]

Dari penjelasan tersebut, penulis akan merancang sistem klasifikasi tingkat akurasi pada penyakit daun sawi hijau berdasarkan gambar digital menggunakan pemrosesan gambar *Convolutional Neural Network (CNN)*. Metode ini bertujuan untuk mengetahui jenis akurasi penyakit daun sawi hijau dari gambar digital untuk menentukan jenis akurasi itu sendiri. Tujuannya agar penyuka daun sawi hijau atau pecinta daun sawi hijau yang awam mengenai jenis akurasi penyakitnya dapat mengenali jenis *level* atau kategori akurasi daun sawi hijau.

2. METODE PENELITIAN

Metode penelitian klasifikasi penyakit daun sawi menggunakan *CNN* melibatkan beberapa tahapan. Sebuah penelitian yang dilakukan menggunakan metode *CNN* untuk klasifikasi varietas pada citra daun sawi mencakup tahap akuisisi data, *preprocessing* data, pembagian data, pelatihan model *CNN* dan evaluasi model.

2.1 Dataset Daun Sawi Hijau

Proses *validasi* bertujuan untuk menguji aplikasi dalam memprediksi data baru yang sebelumnya belum pernah dilihat atau data diluar dari data *training*. Proses validasi menggunakan 60 sampel data validasi dari 2 kelas. Keberhasilan validasi dihitung menggunakan nilai *Accuracy*. Hasil pengujian aplikasi dengan 60 data *validasi* berdasarkan *Accuracy* diperoleh jumlah data yang diklasifikasi dengan benar yaitu 38 data dan yang belum diklasifikasi dengan tepat yaitu 22 data. Berdasarkan *Accuracy* didapatkan nilai akurasi aplikasi dalam mengenali data validasi mana tanaman sehat dan sakit pada tanaman sawi hijau sebagai berikut. $38 \times 100\% = 63\%$, data yang digunakan yaitu data primer yaitu data diperoleh secara langsung melalui wawancara, pengambil gambar tanaman sawi hijau dilakukan dengan satu sisi saja, jadi *accuracy* 63%. Berdasarkan perhitungan diatas alat telah berhasil mengenali data validasi dengan memperoleh *Accuracy* sebesar 63%.

Pada gambar 1 dapat kita lihat bahwa tingkat *akurasi* terdapat pada penyakitnya. Gambar dibawah ini merupakan gambar daun sawi sehat dengan tingkat akurasi *Alga Leaf Spot*. Pada gambar daun sawi dengan tingkat akurasi *Alga Leaf Spot* yang masih awal.



Gambar 1. Kondisi awal daun sawi hijau yang terkena hama.

Pada gambar 2 dapat kita lihat bahwa tingkat akurasi terdapat pada penyakitnya. Gambar dibawah ini merupakan gambar daun sawi sehat dengan tingkat akurasi *light roast*. pada gambar dibawah ini yang merupakan daun sawi hijau dengan tingkat akurasi *light roast* terlihat bahwa penyakit dari daun sawi hijau tersebut telah parah dan dimakan hama.



Gambar 2 Kondisi daun sawi hijau dengan hama yang sudah parah

2.2 Convolutional Neural Network

Convolutional Neural Network (CNN) merupakan salah satu metode *machine learning* yaitu *Multi Layer Perceptron (MLP)* yang dikembangkan lagi. *CNN* dirancang untuk mengolah data dari dua dimensi. Dikarenakan *CNN* memiliki tingkat jaringan dan banyak penerapan yang dilakukan dalam citra, *CNN* masuk kedalam salah satu jenis dari metode *Deep Neural Network*. *CNN* sendiri memiliki dua metode yaitu klasifikasi dengan *feedforward* serta *backpropogation* untuk tahap pembelajarannya. [5]

2.3 Confusion matrix

Confusion Matrix merupakan sebuah alat evaluasi visual yang biasa digunakan dalam *machine learning*. *Confusion matrix* dapat menghitung semua kemungkinan dari kasus masalah klasifikasi karena kolom dalam *confusion matrix* akan mewakili hasil kelas dari prediksi serta baris akan mewakili hasil kelas sebenarnya. *Confusion matrix* akan menampilkan beberapa jenis *output* diantaranya presisi, akurasi, presisi, *recall* serta *f1 score*. Berikut persamaan yang dapat membentuk *confusion matrix*. [6]

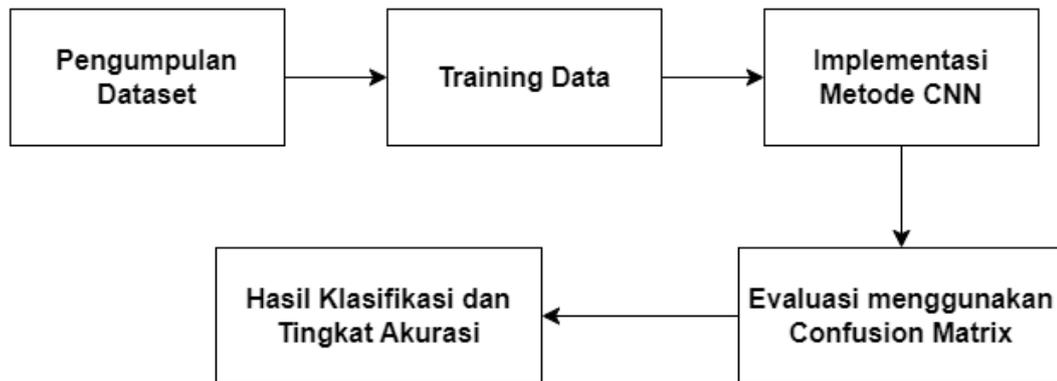
$$Akurasi = \frac{TP+TN}{TP+FP+TN+FN} \dots\dots\dots(1)$$

$$Recall = \frac{TP}{TP+FN} \dots\dots\dots(2)$$

$$Precision = \frac{TP}{TP+FP} \dots\dots\dots(3)$$

$$F1\ Score = \frac{2 \times Recall \times Presisi}{Recall+Presisi} \dots\dots\dots (4)$$

2.4 Alur/tahapan penelitian



Gambar 3. Alur tahapan penelitian

Berikut beberapa alur tahapan yang dilakukan selama proses penelitian:

1. Peneliti melakukan pencarian serta pengumpulan dataset.
2. Peneliti akan melakukan proses pelatihan pada dataset yang telah dikumpulkan dengan mengimplementasikan metode *CNN*.
3. Peneliti akan melakukan uji coba dari hasil pelatihan yang telah dilakukan sebelumnya kemudian akan dilakukan perhitungan akurasi, presisi, *recall*, serta *f1 score* dari hasil pelatihan tersebut.
4. Peneliti akan melakukan evaluasi dari hasil uji coba yang telah dilakukan.
5. Selain itu, peneliti akan menghitung waktu komputasi yang telah dilakukan pada saat menjalankan proses tersebut.

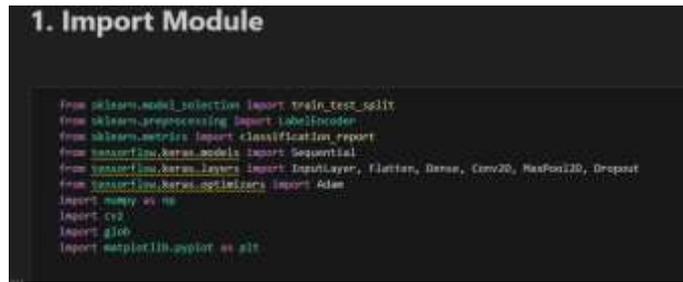
3. HASIL DAN PEMBAHASAN

3.1 Implementasi

Dalam penelitian ini, peneliti menggunakan *software Visual Studio Code* dengan tambahan ekstensi *Jupyter* untuk mempermudah proses penelitian dan dapat menganalisa satu demi satu proses yang sedang dilakukan. Selain itu terdapat beberapa *library* yang digunakan oleh peneliti. Diantaranya *TensorFlow*, *Keras*, *Open CV*, *Numpy*, *Scikit-learn* dan sebagainya. Beberapa tahapan proses yang dilakukan pada penelitian ini adalah sebagai berikut :

1. Melakukan *import Library*

Berbagai *library* yang dipakai pada penelitian ini akan di *import* terlebih dahulu dimana nantinya akan dipanggil pada proses proses selanjutnya. Seperti *library TensorFlow* yang nantinya dapat membantu untuk proses klasifikasi gambar dengan metode *convolutional neural network*. Kemudian terdapat *library LabelEncoder* untuk mengubah label pada kelas yang awalnya merupakan kata akan diubah menjadi angka sehingga memudahkan dalam proses model nantinya. Beberapa *library* lain yang digunakan pada penelitian ini dapat dilihat pada gambar dibawah.



```
1. Import Module

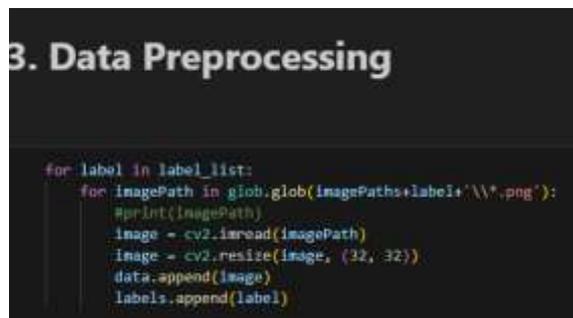
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder
from sklearn.metrics import classification_report
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import InputLayer, Flatten, Dense, Conv2D, MaxPool2D, Dropout
from tensorflow.keras.optimizers import Adam

import numpy as np
import cv2
import glob
import matplotlib.pyplot as plt
```

Gambar 4. Proses *Import Library*

2. Proses pemanggilan *dataset* dan *praprocessing*

Dataset yang terdapat didalam folder lokal komputer, akan dipanggil. Agar proses klasifikasi nantinya berjalan lebih cepat dan tidak menghambat jalannya proses maka data yang berupa gambar akan dilakukan proses *praprocessing* yaitu dengan merubah ukuran dari masing masing gambar. Pada penelitian ini kami akan merubah semua ukuran dari gambar menjadi 32x32.



```
3. Data Preprocessing

for label in label_list:
    for imagePath in glob.glob(imagePaths+label+'\\*.png'):
        #print(imagePath)
        image = cv2.imread(imagePath)
        image = cv2.resize(image, (32, 32))
        data.append(image)
        labels.append(label)
```

Gambar 5. Proses *preprocessing*

3. Proses Perubahan Label Menjadi Angka

Pada proses ini label yang terdapat pada sistem akan dirubah menjadi angka yang awalnya merupakan kata. Proses ini nantinya akan memudahkan proses model.

4. *Split* dataset

Membagi *dataset* menjadi data yang akan digunakan untuk *training* dan data yang akan digunakan untuk *testing*. Proses ini menggunakan *library split* dengan tambahan parameter ukuran *test* yaitu 0.2 serta *random state* dengan nilai 42.

5. Membangun arsitektur CNN

Proses ini akan membuat atau membangun model *CNN* yang akan dilakukan pada saat proses *training* dan *testing* nantinya. Dimana dalam membangun model ini membutuhkan beberapa *layer* agar nantinya klasifikasi akan menghasilkan nilai sesuai dengan yang diharapkan.



```
model = Sequential()
# Extracted Feature Layer
model.add(InputLayer(input_shape=(32, 32, 3)))
model.add(Conv2D(filters=32, kernel_size=2, strides=1,
padding='same', activation='relu'))
model.add(MaxPool2D(pool_size=2, padding='same'))
model.add(Conv2D(filters=50, kernel_size=2, strides=1,
padding='same', activation='relu'))
model.add(MaxPool2D(pool_size=2, padding='same'))
model.add(Dropout(0.25))
model.add(Flatten())
# Fully Connected Layer
model.add(Dense(512, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(1, activation='sigmoid'))
```

Gambar 6. Pembangunan model *CNN*

6. Proses *training*

Setelah membuat beberapa kode untuk membangun model dari *CNN*, akan dilakukan proses *training* pada data *training* yang sudah tersedia. Peneliti menggunakan 10 *epoch* dalam proses *training* ini. Selain itu data *testing* yang sudah dibagi sebelumnya akan dibuat sebagai *validation data*.

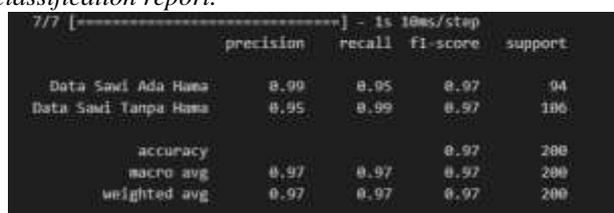
7. Evaluasi

Setelah proses *training* dijalankan, akan tampil proses *training* pada setiap *epoch* dengan beberapa rincian lain yaitu nilai *loss*, akurasi, *val_loss* serta *val accuracy*. Beberapa nilai yang keluar tersebut akan dianalisa oleh peneliti dan kemudian akan dievaluasi apakah model yang digunakan sudah menghasilkan nilai yang sesuai dengan harapan atau belum. Berikut hasil proses *training* yang telah dilakukan oleh peneliti.



Gambar 7. Hasil dari proses *Training*

Dapat dilihat bahwa nilai yang didapatkan oleh peneliti pada *epoch* 10 mendapatkan tingkat akurasi mencapai 99%. Selain menganalisa dari hasil proses *training*, peneliti juga mengevaluasi model menggunakan *classification report*.



	precision	recall	f1-score	support
Data Sawi Ada Hama	0.99	0.95	0.97	94
Data Sawi Tanpa Hama	0.95	0.99	0.97	196
accuracy			0.97	290
macro avg	0.97	0.97	0.97	290
weighted avg	0.97	0.97	0.97	290

Gambar 8. *Classification Report*

Pada Gambar 8 peneliti menggunakan *classification report* untuk mengevaluasi model. Terdapat beberapa nilai yang ditampilkan oleh *classification report* diantaranya *precision*, *recall*, serta *f1 score* pada setiap kelas.

8. Proses *Testing* menggunakan data acak

Setelah menganalisa model, peneliti melakukan *testing* terhadap model dengan memanggil file gambar acak yang kemudian akan diproses oleh model. *Output* yang diberikan berupa tampilan gambar beserta dengan klasifikasinya.



Gambar 9. Hasil dari proses *klasifikasi*

4. SIMPULAN

Dapat dikatakan bahwa metode *Convolutional Neural Network* (CNN) memiliki kemampuan mengetahui penyakit daun sawi hijau. Berdasarkan uji coba model yang menggunakan *cross-validation* sepuluh kali dengan nilai $K=10$ dan menggunakan data pengujian atau data yang tidak terlihat, skenario 6 menunjukkan tingkat akurasi tertinggi sebesar 96,6%, sementara skenario 8 menunjukkan penurunan sebesar 1,9% pada pengujian data pengujian. Nilai spesifisitas total 98,33%, dengan nilai terendah dan tertinggi masing-masing kelas 97% dan 100%, dan rata-rata per kelas 98,35%. Nilai sensitivitas total 96,6%, dengan nilai terendah dan terendah masing-masing kelas 97% dan 100%.

5. SARAN

1. Pada penelitian selanjutnya diharapkan menambahkan lebih banyak *dataset* yang digunakan sebagai data *training* dan data *testing* karena hal tersebut mampu meningkatkan nilai *akurasi* dan performa model dalam mengklasifikasikan gambar.

2. Pada penelitian selanjutnya diharapkan membuat *slider* kamera berbentuk beda supaya bisa mengenali keseluruhan pada daun tanaman *pakcoy*

DAFTAR PUSTAKA

- [1] A. Kurniadi, “Implementasi Convolutional Neural Network Untuk Klasifikasi Varietas Pada Citra Daun Sawi Menggunakan Keras,” *DoubleClick J. Comput. Inf. Technol.*, vol. 4, no. 1, p. 25, 2020, doi: 10.25273/doubleclick.v4i1.5812.
- [2] S. Asikin and Y. Lestari, “Aplikasi Insektisida Nabati Berbahan Tanaman Rawa Dalam Mengendalikan Hama Sawi Di Lahan Rawa Pasang Surut,” *Al Ulum J. Sains Dan Teknol.*, vol. 6, no. 2, p. 32, 2021, doi: 10.31602/ajst.v6i2.4841.
- [3] S. Ilahiyah and A. Nilogiri, “Implementasi Deep Learning Pada Identifikasi Jenis Tumbuhan Berdasarkan Citra Daun Menggunakan Convolutional Neural Network,” *JUSTINDO (Jurnal Sist. dan Teknol. Inf. Indones.)*, vol. 3, no. 2, pp. 49–56, 2018.
- [4] U. S. Rahmadhani and N. L. Marpaung, “Klasifikasi Jamur Berdasarkan Genus Dengan Menggunakan Metode CNN,” *J. Inform. J. Pengemb. IT*, vol. 8, no. 2, pp. 169–173, 2023, doi: 10.30591/jpit.v8i2.5229.
- [5] J. V. P. Putra, F. Ayu, and B. Julianto, “Implementasi Pendeteksi Penyakit pada Daun Alpukat Menggunakan Metode CNN,” *Stain. (Seminar Nas. Teknol. Sains)*, vol. 2, no. 1, pp. 155–162, 2023.
- [6] A. Ridhovan and A. Suharso, “Penerapan Metode Residual Network (Resnet) Dalam Klasifikasi Penyakit Pada Daun Gandum,” *JUPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.)*, vol. 7, no. 1, pp. 58–65, 2022, doi: 10.29100/jipi.v7i1.2410.