

# Klasifikasi Hama Serangga pada Pertanian Menggunakan Metode Convolutional Neural Network

Ar'rafi Akram<sup>1</sup>, Kun Fayakun<sup>2</sup>, Harry Ramza<sup>3\*</sup>

Fakultas Teknologi Industri & Informatika, Teknik Elektro, Universitas Muhammadiyah Prof. Dr. Hamka, Jakarta, Indonesia

Email: <sup>1</sup>rafiakram54@gmail.com, <sup>2</sup>kun\_fayakun@uhamka.ac.id, <sup>3,\*</sup>hramza@uhamka.ac.id

Email Penulis Korespondensi: hramza@uhamka.ac.id

Submitted: 10/08/2023; Accepted: 25/09/2023; Published: 27/09/2023

**Abstrak**—Serangan hama serangga merupakan ancaman serius yang berpotensi menyebabkan kerugian yang besar dalam produksi pertanian. Oleh karena itu, pengenalan dan pengendalian hama serangga dengan cara yang efektif sangat penting untuk menjaga produktivitas dan kualitas hasil pertanian. Dalam perkembangan teknologi komputer dan kecerdasan buatan saat ini, teknologi komputer dapat digunakan untuk mengenali citra secara otomatis dalam pengenalan objek, khususnya untuk melakukan klasifikasi hama serangga menggunakan metode Convolutional Neural Network (CNN) dengan arsitektur Xception. CNN merupakan salah satu jenis jaringan saraf tiruan deep feed-forward yang banyak digunakan dalam analisis citra digital dan dapat mengolah data dalam bentuk pola grid. CNN terdiri dari tiga jenis layer, yaitu convolutional layer, pooling layer, dan fully connected layer. Penggunaan CNN dalam penelitian ini bertujuan untuk memudahkan klasifikasi hama serangga. Proses CNN melibatkan tahapan training, testing, dan validation terhadap hama serangga untuk menentukan klasifikasi citra dari berbagai jenis hama serangga tersebut. Dalam penelitian ini, digunakan 1363 gambar citra dengan 13 kelas hama serangga. Proses training CNN melibatkan beberapa parameter seperti batch size, jumlah epoch, learning rate, dan optimizer. Hasil eksperimen menunjukkan akurasi terbaik pada model ini adalah 93,81% pada tahap pelatihan dan 81,75% pada tahap validasi. Hal ini menandakan bahwa model berhasil melakukan klasifikasi hama serangga dengan menggunakan metode CNN.

**Kata Kunci:** Convolutional Neural Network; Citra Digital; Deep Learning; Klasifikasi; Hama Serangga

**Abstract**—Insect pest attacks pose a serious threat that can potentially cause significant losses in agricultural production. Therefore, the effective recognition and control of insect pests are crucial for maintaining agricultural productivity and quality of yields. With the advancement of computer technology and artificial intelligence, computer technology can be utilized to automatically recognize images in object recognition, particularly for insect pest classification using the Convolutional Neural Network (CNN) method with the Xception architecture. CNN is one of the types of deep feed-forward artificial neural networks widely used in digital image analysis and can process data in the form of grid patterns. CNN consists of three types of layers: convolutional layer, pooling layer, and fully connected layer. The use of CNN in this research aims to facilitate the classification of insect pests. The CNN process involves stages of training, testing, and validation on insect pests to determine the classification of images of various insect pest species. This research utilizes 1363 image samples with 13 classes of insect pests. The training process of CNN involves several parameters such as batch size, number of epochs, learning rate, and optimizer. The experiment's results indicate that the best accuracy achieved by this model is 93.81% during the training phase and 81.75% during the validation phase. This demonstrates that the model successfully performs insect pest classification using the CNN method.

**Keywords:** Convolutional Neural Network; Image Processing; Deep Learning; Classification; Insect Pests

## 1. PENDAHULUAN

Pertanian merupakan sektor yang sangat penting dalam perekonomian dan memenuhi kebutuhan pangan di tingkat global. Akan tetapi, serangan hama serangga merupakan ancaman serius yang berpotensi menyebabkan kerugian yang besar dalam produksi pertanian. Oleh karena itu, pengenalan dan pengendalian hama serangga dengan cara yang efektif sangatlah penting untuk mempertahankan produktivitas dan kualitas hasil pertanian[1].

Dalam beberapa tahun terakhir, perkembangan teknologi komputer dan kecerdasan buatan, khususnya *Convolutional Neural Network* (CNN), telah membuka potensi baru dalam pengenalan pola dan klasifikasi berbasis citra digital. CNN merupakan Salah satu jenis jaringan saraf tiruan *deep feed-forward* yang banyak digunakan dalam analisis citra digital dan data dapat diolah dalam bentuk pola grid dengan menggunakan tiga jenis layer: *convolutional layer*, *pooling layer*, dan *fully connected layer*. Setiap layer memiliki peran yang berbeda, *convolutional layer* dan *pooling layer* bertugas untuk mengekstraksi fitur, sementara *fully connected layer* bertanggung jawab memetakan fitur hasil ekstraksi untuk mendapatkan *output* klasifikasi. CNN memiliki keunggulan karena tidak memerlukan metode ekstraksi ciri khusus. Dalam proses ini, beberapa tahap digunakan untuk menghasilkan atribut yang diperlukan dalam klasifikasi. Namun, metode CNN memerlukan jumlah data pelatihan yang besar. Oleh karena itu, diperlukan biaya yang signifikan untuk melatih model. Setelah model CNN telah dibuat, penting untuk melakukan evaluasi kinerja sistem klasifikasi. Evaluasi kinerja ini mencerminkan seberapa baik sistem dalam melakukan klasifikasi data[2]. CNN adalah bagian dari model *deep learning* yang dipilih untuk memudahkan klasifikasi hama serangga pada pertanian.

*Deep Learning* adalah cabang dari *Machine Learning* yang menggunakan algoritma berdasarkan hukum matematika dan beroperasi serupa dengan otak manusia. Salah satu metode dalam *Deep Learning* yang digunakan dalam *Image Processing* adalah *Convolutional Neural Network* (CNN). CNN merupakan perkembangan dari *Multilayer Perceptron* (MLP) yang dirancang untuk memproses data seperti gambar atau suara. CNN memiliki kemampuan untuk belajar langsung dari citra, sehingga mengurangi ketergantungan pada pemrograman yang kompleks. Dalam bidang ilmu *Deep Learning*, metode ini sangat efektif dalam klasifikasi gambar dan mampu

memberikan prediksi yang akurat dalam pengenalan gambar. Tujuannya adalah untuk meniru sistem pengenalan gambar manusia dalam memproses informasi pada gambar[3].

Pada penelitian yang dilakukan oleh Susy Yuliany, dkk pada tahun 2022 menggunakan metode CNN untuk klasifikasi hama tanaman padi ditemukan Hasil evaluasi menunjukkan bahwa pembagian data dengan rasio 90%:10% adalah pembagian data yang paling sesuai untuk dataset ini. Arsitektur model yang digunakan mencapai akurasi training sebesar 83,02%, 78,30%, dan 81,13%. Sedangkan untuk pengujian, ketiga model tersebut mencapai akurasi sebesar 69,33%, 77,33%, dan 76% [4].

Pada penelitian yang dilakukan oleh Derry Alamsyah dan Dicky Pratama pada tahun 2020 dilakukan eksperimen tanpa *dropout* menggunakan beberapa optimizer, namun mengalami masalah *overfitting*. Hasil eksperimen menunjukkan bahwa akurasi validasi tertinggi tercapai saat menggunakan optimizer Adam, yakni mencapai 66%, sedangkan akurasi training mencapai 84%. Namun, saat model tersebut diuji, akurasi yang dihasilkan hanya mencapai 64% [5].

Pada penelitian yang dilakukan oleh Mellynia Sanjaya & Eddy Nurraharjo pada tahun 2023 untuk deteksi pada rempah-rempah menggunakan dataset berupa 1800 gambar yang terdiri dari 12 jenis rempah-rempah, berhasil mencapai akurasi tertinggi 86% dengan rata-rata akurasi sebesar 60% [6].

Pada penelitian yang dilakukan oleh Rijal Abdulhakim, dkk pada tahun 2021 untuk klasifikasi kendaraan prioritas disimpulkan bahwa skenario terbaik terjadi pada skenario 2, di mana data pelatihan sebesar 60%, data validasi sebesar 20%, dan data pengujian sebesar 20%. Dalam skenario ini, diperoleh validation accuracy sebesar 66,15% dan testing accuracy sebesar 69,231% [7].

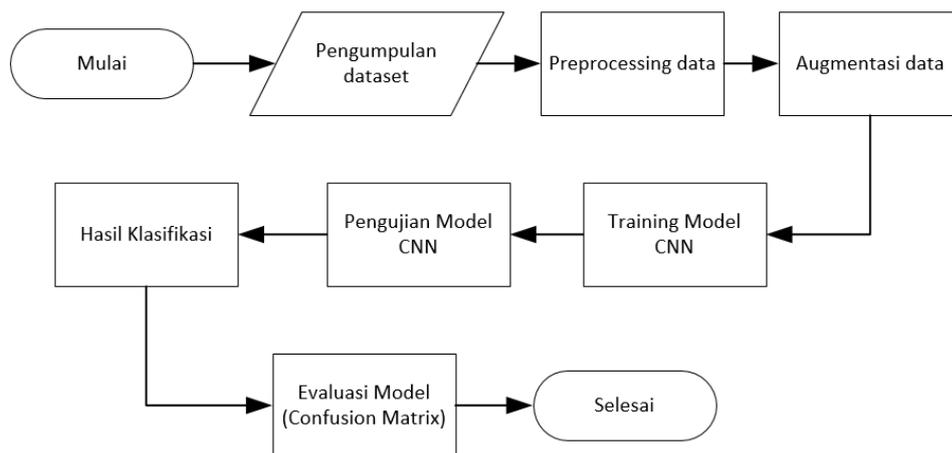
Pada penelitian yang dilakukan oleh Samuel, dkk pada tahun 2019 Hasil pelatihan model menggunakan citra UAV perkebunan pada bulan November 2021 menyimpulkan bahwa model ini merupakan model terbaik dengan F1 Score sebesar 84% selama proses pelatihan. Sementara itu, rata-rata nilai F1 Score untuk data uji adalah sebesar 71%. Meskipun demikian, terdapat kendala yang mempengaruhi kurang maksimalnya akurasi dalam mendeteksi pokok pohon sawit, yaitu perbedaan kualitas citra UAV yang digunakan dalam proses pengambilan data[8].

Berdasarkan penjelasan di atas, penelitian ini akan menggunakan dataset publik dari Kaggle. Tujuan dari penelitian ini adalah untuk melakukan klasifikasi hama serangga pada pertanian menggunakan metode *Convolutional Neural Network* (CNN) dengan menggunakan arsitektur CNN Xception yang memiliki dampak signifikan dalam memajukan sektor pertanian, dengan harapan melalui penggunaan metode CNN, penelitian ini memungkinkan identifikasi hama serangga dengan tingkat akurasi yang lebih tinggi daripada metode tradisional. Hal ini sangat penting karena membantu petani dalam mengenali jenis-jenis serangga yang dapat merusak tanaman dengan cepat dan tepat, dan memungkinkan deteksi dini serangan hama serangga pada tahap awal. Hal ini memungkinkan tindakan respons yang lebih cepat, seperti penerapan pengendalian yang lebih efektif, sehingga mengurangi kerugian akibat serangan hama. Untuk mencapai tujuan ini, akan dilakukan beberapa percobaan yang telah ditentukan sebelumnya. Selain itu, akan diterapkan beberapa parameter untuk meningkatkan akurasi pada pelatihan dan teknik regularisasi yaitu *dropout* untuk mengurangi potensi terjadinya *overfitting* pada model CNN.

## 2. METODOLOGI PENELITIAN

### 2.1 Tahapan Penelitian

Dalam penelitian ini, terdapat rangkaian tahapan penelitian yang dijalankan dalam prosesnya untuk mengumpulkan data, menganalisis informasi, dan mencapai tujuan penelitian. Setiap tahapan tersebut memiliki kepentingannya masing-masing dalam memastikan bahwa penelitian dilaksanakan secara sistematis dan dapat diandalkan. sehingga sesuai dengan harapan dalam Klasifikasi Hama Serangga pada Pertanian Menggunakan Metode *Convolutional Neural Network*. Berikut adalah urutan tahapan penelitian tersebut :



**Gambar 1.** Tahapan Penelitian

a. Pengumpulan *Dataset*

Pada langkah ini, *dataset* dikumpulkan dari situs publik Kaggle sebagai sumber data dan bahan untuk penelitian. *Dataset* ini mencakup data mentah yang akan digunakan dalam proses pengenalan citra.

b. *Preprocessing Data*

*Preprocessing* merupakan langkah yang dilakukan sebelum melakukan pelatihan atau pengujian model[9]. Sebelum melakukan pembagian dataset menjadi data pelatihan, data pengujian, dan data validasi, terlebih dahulu dilakukan pra-proses dataset. Tahapannya dimulai dengan seleksi dataset awal yang berjumlah 1591 citra menjadi 1363 citra. Hal ini dilakukan karena ada beberapa gambar yang sama dengan kelas yang berbeda, dan ada kekhawatiran bahwa hal ini akan mengganggu proses pembelajaran model *Convolutional Neural Network (CNN)* nantinya. *Preprocessing* dilakukan untuk meningkatkan kualitas citra, mengurangi *noise*, serta menyesuaikan karakteristik citra agar sesuai dengan persyaratan analisis atau pengolahan citra selanjutnya. Dalam penelitian ini, *preprocessing* yang dilakukan adalah *resizing*, yaitu proses untuk menyesuaikan ukuran gambar yang digunakan dalam dataset pelatihan, pengujian, atau klasifikasi gambar. Semua gambar diubah menjadi ukuran 224x224 pixel. Citra selanjutnya diubah menjadi format RGB dan dinormalisasi dengan pembagian 255 sehingga setiap pixel akan memiliki nilai antara 0 hingga 1, hal ini bertujuan untuk mempercepat waktu komputasi[10].

c. Augmentasi Data

Augmentasi data adalah teknik yang digunakan untuk menciptakan variasi baru dari data latih dengan melakukan transformasi pada gambar-gambar tersebut[11]. Proses augmentasi data bertujuan untuk meningkatkan jumlah sampel data latih dan menciptakan variasi dalam data, sehingga model dapat lebih baik dalam menggeneralisasi dan mengatasi *overfitting*. Parameter yang digunakan dalam penelitian ini adalah *zoom\_range* dengan nilai 0.2, *fill\_mode* dengan nilai *nearest*, *rotation\_range* dengan nilai 30, *width\_shift\_range* dengan nilai 0.2, *height\_shift\_range* dengan nilai 0.2, *shear\_range* dengan nilai 0,2, dan *horizontal\_flip* dengan nilai *true*.

d. Training Model CNN

Data training terdiri dari *dataset* yang telah disesuaikan jumlahnya untuk diproses menggunakan *Convolutional Neural Network (CNN)*[12]. Total dataset yang digunakan berjumlah 1363 gambar dengan 13 kelas, dimana 1090 dataset digunakan untuk pelatihan (*training*), 136 dataset digunakan untuk pengujian (*testing*), dan 137 dataset digunakan untuk validasi (*validation*). Sebelum digunakan, dataset tersebut telah melalui proses *preprocessing* dan augmentasi data[13]. Penelitian ini menggunakan arsitektur CNN yaitu Xception.

```

Model: "sequential_3"
-----
Layer (type)                Output Shape                Param #
-----
xception (Functional)      (None, 7, 7, 2048)         20861480
conv2d_19 (Conv2D)         (None, 5, 5, 32)           589856
max_pooling2d_3 (MaxPoolin (None, 2, 2, 32)           0
g2D)
dropout_3 (Dropout)        (None, 2, 2, 32)           0
flatten_3 (Flatten)        (None, 128)                 0
dense_3 (Dense)            (None, 13)                  1677
-----
Total params: 21453013 (81.84 MB)
Trainable params: 591533 (2.26 MB)
Non-trainable params: 20861480 (79.58 MB)
    
```

Gambar 2. Model yang digunakan

1. Xception

Xception adalah suatu struktur *Convolutional Neural Network* yang mengadopsi pendekatan konvolusi yang dapat dipisahkan secara mendalam. Arsitektur ini merupakan hasil pengembangan dari struktur Inception[16]. Arsitektur Xception menggabungkan *depthwise convolution* dan *pointwise convolution* untuk membentuk blok-blok residu yang memungkinkan informasi untuk mengalir melalui lapisan-lapisan secara lebih efisien. Salah satu keunggulan utama dari Xception terletak pada efisiensinya, di mana jumlah parameter dan komputasi yang diperlukan lebih sedikit dibandingkan dengan arsitektur CNN lainnya. Keefisienan ini membuat Xception menjadi pilihan yang optimal untuk aplikasi yang memiliki keterbatasan sumber daya komputasi.

2. *Convolution layer*

Convolution Layer adalah langkah kunci yang menjadi dasar bagi metode CNN, yang melibatkan proses konvolusi terhadap keluaran dari lapisan sebelumnya. Layer Konvolusi menggunakan filter yang disebut kernel untuk mengekstraksi objek atau fitur dari gambar input. Kernel ini terdiri dari bobot yang berperan dalam mendeteksi karakteristik dari objek. Selanjutnya, langkah konvolusi dilakukan untuk menghasilkan transformasi linear dari gambar input yang mencerminkan informasi spasial pada data tersebut[21].

### 3. Pooling Layer

*Pooling Layer* berfungsi untuk mempertahankan dimensi data selama proses konvolusi, melalui pengurangan sampel atau pengurangan spasial input (pengurangan parameter). Melalui operasi pooling, kita dapat mengubah representasi data menjadi lebih ringkas, lebih mudah dikelola, dan membantu mengontrol overfitting. Dalam praktiknya, metode pooling yang paling umum digunakan adalah *Max Pooling*. Dalam metode ini, kita memilih nilai maksimum dari area tertentu. Selain itu, ada juga metode pooling lain yang disebut *average pooling*, di mana kita mencari nilai rata-rata dari area yang sama[17].

### 4. Dropout

*Dropout* adalah metode yang digunakan untuk mengatasi *overfitting* pada jaringan saraf dengan menghalangi adaptasi yang terlalu kompleks saat melatih data. Tambahan lagi, *dropout* juga bisa mempercepat proses pembelajaran dengan mengacak dan mengabaikan neuron yang tidak aktif selama pelatihan[21].

### 5. Flatten

Flattening melibatkan mengubah matriks yang terdapat di lapisan pooling menjadi satu baris atau vektor tunggal. Vektor ini kemudian digunakan sebagai bagian dari lapisan input pada jaringan saraf tiruan. Dari hasil pemetaan fitur yang telah di-pool sebelumnya, kita hanya perlu mengambil setiap baris dan menggabungkannya menjadi satu baris dengan satu kolom[12].

### 6. Dense

Ini adalah lapisan *fully connected* (dense layer), yang digunakan untuk melakukan klasifikasi berdasarkan fitur-fitur yang diambil dari lapisan sebelumnya. Dalam langkah klasifikasi, metode *softmax* melakukan perhitungan peluang untuk setiap kelas target relatif terhadap seluruh kelas yang ada dalam tujuan klasifikasi. Kisaran nilai probabilitas yang dihasilkan oleh softmax adalah antara 0 dan 1. Jika semua probabilitas kelas target dijumlahkan, hasilnya akan sama dengan satu. Softmax mengambil nilai eksponensial dari input yang diberikan dan menjumlahkan nilai-nilai eksponensial tersebut untuk semua kelas dalam hasil keluaran[17].

### e. Pengujian Model CNN

Pengujian model CNN merujuk pada proses evaluasi dan pengujian kinerja CNN setelah dilakukan pelatihan atau training menggunakan data pelatihan (*training data*). Setelah model CNN berhasil dilatih dengan dataset pelatihan, langkah selanjutnya adalah menguji kinerjanya dengan menggunakan dataset uji (*testing data*) dan dataset validasi data yang sebelumnya tidak pernah digunakan dalam proses pelatihan. Tujuan dari pengujian model CNN adalah untuk mengukur seberapa baik model tersebut dapat menggeneralisasi dan mampu mengklasifikasikan data baru yang tidak pernah dilihat sebelumnya. Pengujian ini juga membantu untuk memastikan bahwa model tidak mengalami *overfitting*, yaitu kondisi ketika model terlalu menghafal data pelatihan dan kinerjanya menurun ketika diberikan data baru.

### f. Klasifikasi Citra

Pada proses ini akan menggunakan model untuk melakukan klasifikasi pada dataset pengujian. Klasifikasi citra pada *Convolutional Neural Network* (CNN) merujuk pada kemampuan model CNN untuk mengidentifikasi hama serangga dalam citra berdasarkan pembelajaran yang telah dilakukannya selama proses pelatihan dengan menggunakan data pelatihan[15].

### g. Confusion Matrix

Pada tahap ini, dilakukan perhitungan performa dari model arsitektur *transfer learning* yang digunakan. Performa diukur menggunakan metrik akurasi, presisi, recall, dan F1 Score, yang dapat dilihat secara berurutan pada persamaan (1), (2), (3), dan (4). Variabel TP mewakili jumlah prediksi benar positif (*True Positive*), TN mewakili jumlah prediksi benar negatif (*True Negative*), FP mewakili jumlah prediksi salah positif (*False Positive*), dan FN mewakili jumlah prediksi salah negatif (*False Negative*) [14][20]. Berikut ini nilai untuk mengevaluasi model klasifikasi :

#### 1. Accuracy

*Accuracy* adalah matriks yang digunakan untuk mengukur perbandingan keseluruhan prediksi yang benar terhadap total nilai kelas[14].

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

#### 2. Precision

*Precision* adalah matriks yang dipergunakan untuk menghitung akurasi atau hasil prediksi yang positif dari keseluruhan hasil yang diprediksi sebagai positif[14].

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

3. *Recall*

*Recall* adalah sebuah matriks yang digunakan untuk menghitung jumlah prediksi yang relevan dengan kelas yang sama[18].

$$Recall = \frac{TP}{TP+FN} \tag{3}$$

4. *F1 Score*

*F1 Score* adalah perbandingan rata-rata *precision* dan *recall* yang dibobotkan[18].

$$F1\ Score = \frac{Precision \times Recall}{Precision+Recall} \tag{4}$$

Keterangan dari formula diatas adalah :

- True Positive* (TP) = Variabel TP mewakili jumlah prediksi benar positif
- True Negative* (TN) = Variabel TN mewakili jumlah prediksi benar negatif
- False Positive* (FP) = Variabel FP mewakili jumlah prediksi salah positif
- False Negative* (FN) = Variabel FN mewakili jumlah prediksi salah negatif

### 3. HASIL DAN PEMBAHASAN

Pada bagian hasil dan pembahasan, disajikan hasil dan analisis dari pengujian klasifikasi hama serangga pada pertanian menggunakan metode *Convolutional Neural Network* (CNN) dengan variasi beberapa parameter inisialisasi. Berdasarkan pada bagian pendahuluan, terdapat beberapa penelitian terkait yang belum mencapai akurasi maksimal. Ini dapat disebabkan oleh beberapa faktor, seperti tidak menggunakan arsitektur CNN dan tidak memanfaatkan parameter *dropout*. Untuk itu pada penelitian ini akan menggunakan parameter-parameter yang diuji meliputi *batch size*, jumlah *epoch*, *Dropout*, *learning rate*, dan *optimizer*. Penentuan nilai awal parameter-parameter ini bertujuan untuk mencari akurasi terbaik dan tingkat pembelajaran yang tidak diketahui sebelumnya. Dalam penelitian ini, dilakukan pengujian dengan menggunakan nilai *batch size* sebesar 16, 32, dan 64. Selanjutnya, jumlah *epoch* yang digunakan adalah 10, 25, dan 50. *Dropout* yang diterapkan memiliki nilai 0.3. Sementara itu, *learning rate* yang digunakan adalah 0.0001 dengan penggunaan *optimizer* Adam. Hasil pengujian dilakukan menggunakan confusion matrix untuk mendapatkan nilai akurasi, presisi, loss, recall, f-1 score[19], serta prediksi pengujian citra pada hama serangga. Diharapkan hasil dan analisis dari pengujian ini dapat memberikan wawasan lebih lanjut mengenai kinerja model CNN dalam mengatasi masalah klasifikasi hama serangga.

#### 3.1 Hasil Pengumpulan Dataset

Pada tahap pengumpulan data, penulis melakukan eksplorasi data dari berbagai sumber dataset, termasuk di antaranya adalah situs dataset publik Kaggle, di mana penulis berhasil mendapatkan dataset hama serangga. Setelah dataset uji terkumpul, langkah berikutnya adalah melakukan pengolahan citra uji (*image preprocessing*), di mana dataset uji akan diubah ukurannya menjadi 224x224 piksel untuk mempercepat proses komputasi citra [17]. Pada tahap ini, dilakukan konfigurasi data augmentasi pada model yang akan dibangun, dengan tujuan untuk menghindari *overfitting* dan meningkatkan akurasi model[18]. Dataset hama serangga telah dikelompokkan ke dalam 13 kelas, yaitu *Africanized Honey Bees (Killer Bees)*, *Aphids*, *Armyworms*, *Brown Marmorated Stink Bugs*, *Cabbage Loopers*, *Citrus Canker*, *Colorado Potato Beetles*, *Corn Earworms*, *Fruit Flies*, *Spider Mites*, *Thrips*, *Tomato Hornworms*, dan *Western Corn Rootworms*. Berikut ini adalah contoh gambar dari 13 kelas hama serangga tersebut :



Gambar 3. Contoh dataset citra hama serangga

#### 3.2 Hardware dan Software

Dalam penelitian ini, penulis memanfaatkan perangkat lunak dan perangkat keras yang ada. Proses pelatihan model dilakukan secara eksklusif menggunakan beberapa perangkat keras. Untuk informasi lebih rinci mengenai spesifikasi, dapat dilihat pada Tabel 1.

Tabel 1. Perangkat yang digunakan

Hardware	Software
CPU : Intel i5-7200U 2.5Ghz	Python 3.11
Nvidia GeForce 920MX 2GB	Jupyter Notebook
RAM : 8GB	Tensorflow & Keras Framework
Harddisk : 1TB	Command Prompt
SSD : 256GB	OS : Windows 10

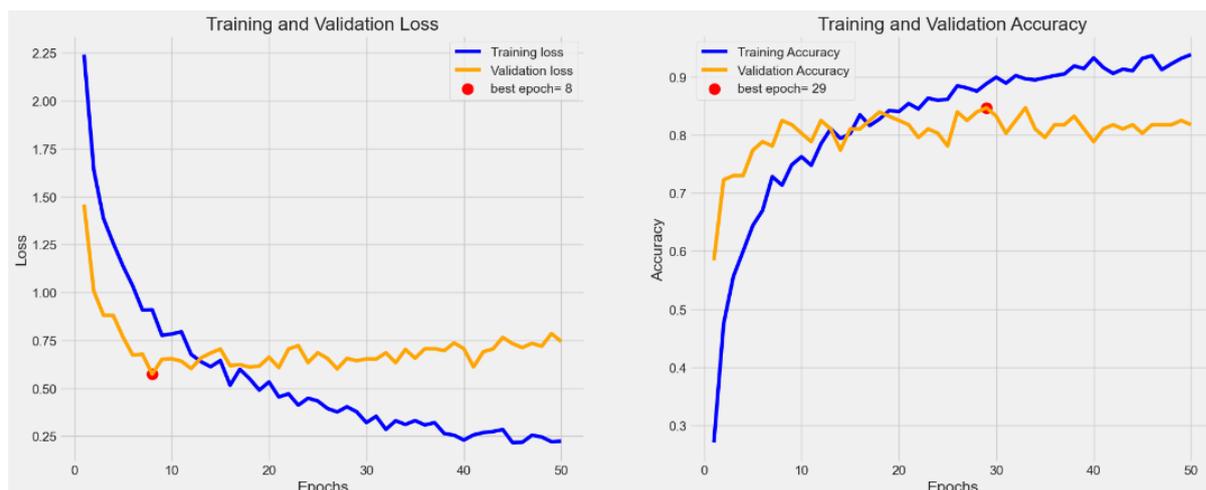
### 3.3 Hasil Pelatihan Model CNN

Hasil pelatihan model *Convolutional Neural Network* (CNN) merujuk pada performa dan kualitas model setelah proses pelatihan menggunakan data pelatihan yang ada. Proses pelatihan melibatkan penyesuaian parameter model (misalnya, bobot dan bias pada lapisan-lapisan CNN) berdasarkan data pelatihan, dengan tujuan agar model dapat belajar untuk memahami pola dan fitur pada data tersebut. Hasil pelatihan model CNN dapat dievaluasi berdasarkan berbagai metrik dan indikator untuk mengukur sejauh mana model dapat menggeneralisasi data yang belum pernah dilihat sebelumnya. Hasil dari proses pelatihan dapat dilihat pada Tabel 2 di bawah ini :

Tabel 2. Hasil dari Pelatihan Model CNN

No.	Ukuran Citra (Pixel)	Batch Size	Epoch	Training Accuracy	Validation Accuracy
1.	224x224	16	10	0.7699	0.8029
2.	224x224	16	25	0.8891	0.7956
3.	224x224	16	50	0.9381	0.8175
4.	224x224	32	10	0.7311	0.7810
5.	224x224	32	25	0.8484	0.8029
6.	224x224	32	50	0.9251	0.8175
7.	224x224	64	10	0.6913	0.7372
8.	224x224	64	25	0.8124	0.7956
9.	224x224	64	50	0.9113	0.8175

Berdasarkan Tabel 2, hasil pelatihan model *Convolutional Neural Network* (CNN) dari 9 percobaan ini merupakan upaya untuk mencari nilai akurasi yang optimal pada model yang sedang dikembangkan. Proses ini melibatkan variasi beberapa parameter yang dapat mempengaruhi performa model, seperti ukuran batch dan jumlah epoch. Dengan melakukan percobaan yang berbeda menggunakan berbagai kombinasi nilai-nilai parameter, penulis dapat memahami bagaimana model merespons perubahan tersebut. Pada setiap percobaan, model CNN dilatih dengan menggunakan konfigurasi parameter yang berbeda. Setelah proses pelatihan selesai, nilai akurasi pada tahap pelatihan dan tahap validasi diukur dan dicatat. Selanjutnya, hasil-hasil dari setiap percobaan dievaluasi dan dibandingkan untuk menentukan konfigurasi parameter yang menghasilkan akurasi yang baik. Dalam penjelasan hasil pelatihan ini, biasanya akan disertakan analisis dari setiap percobaan, termasuk grafik akurasi dan loss selama pelatihan, serta perbandingan performa model pada setiap konfigurasi parameter. Analisis ini membantu dalam memahami bagaimana perubahan nilai parameter dapat mempengaruhi kinerja model, mengidentifikasi apakah terdapat *overfitting* atau *underfitting*, dan menemukan nilai parameter yang dapat menghasilkan akurasi yang tinggi dan stabil pada data yang tidak digunakan dalam pelatihan (data validasi). Dengan demikian, dari hasil pelatihan model CNN pada 9 percobaan ini, ditemukan bahwa hasil akurasi terbaik diperoleh pada percobaan pelatihan ke-3. Grafik di bawah ini menunjukkan hasil dari percobaan pelatihan ke-3.



Gambar 4. Grafik Akurasi dan loss terbaik pada pelatihan 3

Pada Gambar 4 diatas menunjukkan grafik hasil pelatihan model *Convolutional Neural Network* (CNN) dengan menggunakan arsitektur Xception menunjukkan akurasi yang sangat baik. Pada tahap *training*, model mencapai akurasi sebesar 0.9381, yang menunjukkan bahwa model mampu dengan sangat baik dalam memprediksi data yang telah digunakan selama proses pelatihan. Akurasi training yang tinggi ini menandakan bahwa model mampu mengenali pola dan fitur pada data pelatihan dengan baik. Selain itu, pada tahap validasi, model mencapai akurasi sebesar 0.8175. Akurasi validasi mengukur performa model pada data yang belum pernah dilihat sebelumnya (data validasi). Akurasi validasi yang cukup baik ini menunjukkan bahwa model juga mampu menggeneralisasi dengan baik pada data baru yang tidak ada dalam dataset pelatihan. Model memiliki kemampuan untuk melakukan prediksi yang akurat pada data yang tidak terlibat dalam proses pelatihan.

Model tersebut dilatih dengan menggunakan parameter *batch size* sebesar 16 dan jumlah *epoch* sebanyak 50[22]. *Batch size* mengontrol seberapa banyak sampel yang diproses oleh model sebelum bobotnya diperbarui berdasarkan hasil pelatihan pada setiap iterasi. Dalam kasus ini, penggunaan *batch size* sebesar 16 memberikan hasil yang baik dalam mencapai akurasi yang tinggi. Jumlah *epoch*, yaitu 50, menandakan bahwa model CNN dilatih selama 50 iterasi atau putaran penuh dari seluruh dataset pelatihan. *Epoch* berperan dalam menentukan berapa kali model akan melihat dataset pelatihan dalam proses pembelajaran. Dalam hal ini, 50 *epoch* memberikan kesempatan yang cukup bagi model untuk mengenali dan menyesuaikan diri terhadap pola dan fitur pada data pelatihan. Ukuran citra yang digunakan adalah 224x224 dengan 3 channel RGB, yang menunjukkan bahwa citra memiliki resolusi tinggi dan memiliki tiga saluran warna (merah, hijau, dan biru). Dengan menggunakan citra dalam format ini, model dapat mengambil keuntungan dari informasi warna dan fitur yang lebih kaya untuk proses klasifikasi.

### 3.4 Hasil Klasifikasi Citra

Dari hasil pengujian yang telah dilakukan, terlihat bahwa pada pelatihan model CNN Xception dengan menggunakan citra berukuran 224x224, parameter *batch size* 16 dan *epoch* 50 menunjukkan performa kinerja yang cukup baik. Pada tahap klasifikasi citra ini dilakukan untuk melakukan analisis menyeluruh terhadap gambar dalam setiap kelas dataset. Proses ini menggunakan prediksi secara acak, yang merupakan salah satu subclass dari perpustakaan Keras. Tujuannya adalah untuk meningkatkan efisiensi analisis gambar[23]. Berikut ini adalah hasil klasifikasi yang diperoleh dari arsitektur CNN Xception tersebut.



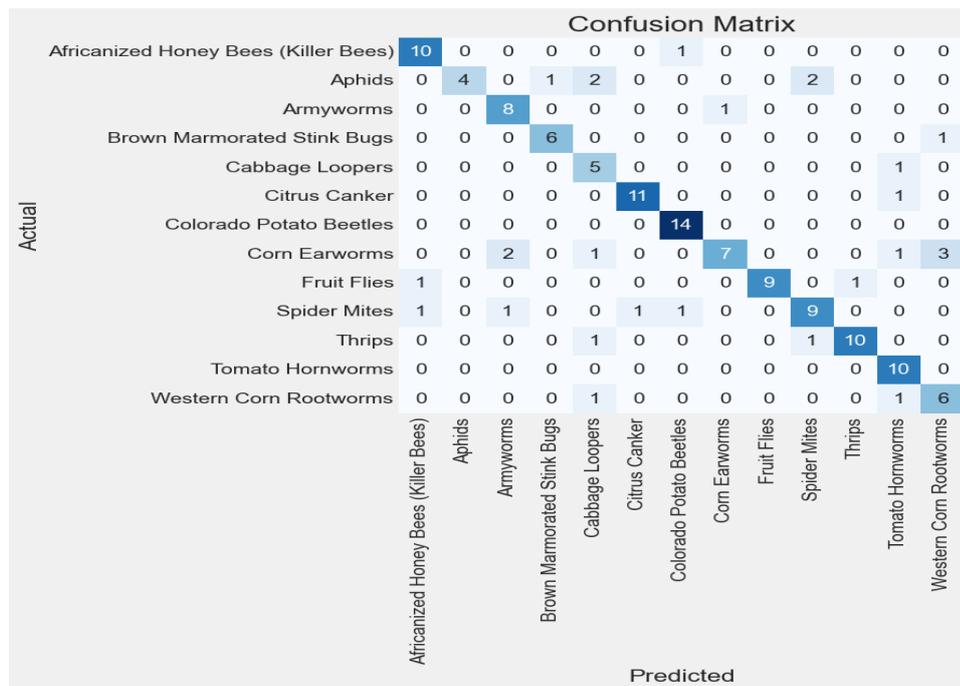
Gambar 5. Hasil Klasifikasi pada percobaan pelatihan 3

Pada Gambar 5 ini menampilkan hasil klasifikasi hama serangga yang diambil beberapa sampel secara acak menggunakan model CNN Xception. Pada citra yang diklasifikasikan dengan benar sebagai kelas hama serangga, bagian nama hama serangga akan ditampilkan dalam warna hijau. Namun, jika citra terdeteksi sebagai kelas yang salah, bagian nama hama serangga akan ditampilkan dalam warna merah. Hasil ini menunjukkan terdapat 2 citra yang terdeteksi salah, yaitu citra kelas *Africanized Honey Bees (Killer Bees)* yang seharusnya diklasifikasikan sebagai citra kelas *Fruit Flies*, dan citra kelas *Colorado Potato Beetles* yang seharusnya diklasifikasikan sebagai citra kelas *Africanized Honey Bees (Killer Bees)*.

### 3.5 Evaluasi Model

Pengujian dalam penelitian ini menggunakan *Confusion Matrix*, suatu pendekatan untuk mengukur performa dalam masalah klasifikasi yang sangat berguna untuk menghitung precision, recall, f1-score, akurasi, dan informasi mengenai jumlah kelas prediksi yang benar atau salah[21][24]. Pengujian ini dilakukan berdasarkan model terbaik yang dihasilkan selama proses pelatihan. Model yang diuji adalah hasil dari proses pelatihan dengan menggunakan

batch size 16 dan epoch 50, dengan akurasi training mencapai 93,81% dan akurasi validasi mencapai 81,75%. Hasil pengujian dari penelitian ini berdasarkan model yang cukup baik dapat dilihat pada Gambar 6 di bawah ini.



Gambar 6. Hasil dari confusion matrix

Pada Gambar 6 menunjukkan hasil dari pengujian confusion matrix yang berhasil dilakukan untuk mengklasifikasi hama serangga menggunakan model CNN Xception. Model yang telah dibuat berhasil memprediksi dengan tepat 109 data dari total 136 data uji. Dengan demikian, diperoleh tingkat akurasi rata-rata sebesar 80%. Dari hasil ini dapat disimpulkan bahwa model pelatihan yang telah diterapkan berfungsi dengan baik.

Tabel 3. Hasil Classification Report Hama Serangga

No.	Hama Serangga	Precision	Recall	F1 Score
1.	Africanized Honey Bees (Killer Bees)	0.83	0.91	0.87
2.	Aphids,	1.00	0.44	0.62
3.	Armyworms	0.73	0.89	0.80
4.	Brown Marmorated Stink Bugs	0.86	0.86	0.86
5.	Cabbage Loopers	0.50	0.83	0.62
6.	Citrus Canker	0.92	0.92	0.92
7.	Colorado Potato Beetles	0.88	1.00	0.93
8.	Corn Earworms	0.88	0.50	0.64
9.	Fruit Flies	1.00	0.82	0.90
10.	Spider Mites	0.75	0.69	0.72
11.	Thrips	0.91	0.83	0.87
12.	Tomato Hornworms	0.71	1.00	0.83
13.	Western Corn Rootworms	0.60	0.75	0.67

Berdasarkan tabel 3 diatas dari hasil classification report hama serangga bahwa classification report adalah laporan yang menyajikan metrik evaluasi performa model klasifikasi. Precision adalah rasio dari jumlah benar positif (true positive) terhadap total prediksi positif (true positive + false positive). Precision mengukur seberapa akurat model dalam mengklasifikasikan data ke dalam kelas positif. Precision yang tinggi menunjukkan bahwa model cenderung tidak memberikan banyak false positive, yaitu kasus-kasus di mana model memprediksi positif tetapi sebenarnya negatif. Precision tertinggi menunjukkan bahwa model memiliki kemampuan yang baik dalam mengidentifikasi hama serangga dengan sedikit false positive. Precision tertinggi pada tabel 3 diatas yaitu hama serangga dengan jenis Aphids dan Fruit Flies. Ini berarti model cenderung memberikan sedikit kesalahan dalam mengklasifikasikan data yang sebenarnya bukan hama serangga sebagai hama serangga. Precision terendah menunjukkan bahwa model memiliki masalah dalam mengklasifikasikan hama serangga dengan banyak false positive. Precision terendah pada tabel 3 tersebut yaitu Cabbage Loopers. Ini berarti model cenderung memberikan banyak kesalahan dalam mengklasifikasikan data yang sebenarnya bukan hama serangga sebagai hama serangga.

Recall, juga dikenal sebagai sensitivity atau true positive rate, adalah rasio dari jumlah benar positif (true positive) terhadap total kasus positif sebenarnya (true positive + false negative). Recall mengukur seberapa baik model

dapat menemukan dan mengidentifikasi semua kasus positif. *Recall* tertinggi terdapat pada hama serangga *Colorado Potato Beetles* dan *Tomato Hornworms* yang menunjukkan bahwa model memiliki kemampuan yang baik dalam menemukan dan mengidentifikasi hama serangga dengan sedikit *false negative*. *Recall* terendah terdapat pada hama serangga *Aphids* dan *Corn Earworms* yang menunjukkan bahwa model memiliki masalah dalam menemukan dan mengidentifikasi hama serangga dengan banyak *false negative*. *F1 score* adalah rata-rata harmonis dari *precision* dan *recall*. *F1 score* menggabungkan kedua metrik ini menjadi satu angka, dan sangat berguna ketika kita ingin mencari keseimbangan antara *precision* dan *recall*. *F1 score* tertinggi terdapat pada hama serangga *Colorado Potato Beetles* yang menunjukkan bahwa model memiliki keseimbangan yang baik antara *precision* dan *recall*. Model ini memiliki performa yang baik dalam mengklasifikasikan hama serangga dengan sedikit kesalahan *false positive* dan *false negative*. *F1 score* terendah terdapat pada hama serangga *Aphids* dan *Cabbage Loopers* yang menunjukkan bahwa model tidak memiliki keseimbangan yang baik antara *precision* dan *recall*. Model ini mungkin memiliki masalah dalam mengklasifikasikan hama serangga, baik dengan banyak *false positive* atau *false negative*.

#### 4. KESIMPULAN

Berdasarkan hasil dan pembahasan yang telah dijelaskan bahwa Klasifikasi Hama Serangga menggunakan metode CNN digunakan 1363 gambar citra dengan 13 kelas hama serangga. Data tersebut terbagi menjadi 1090 dataset untuk pelatihan (*training*), 136 dataset untuk pengujian (*testing*), dan 137 dataset untuk validasi (*validation*). Proses *preprocessing* dilakukan dengan *resizing* semua gambar diubah menjadi ukuran 224x224 pixel dengan input ukuran citra yaitu 224x224 pixel dengan 3 channel RGB. Data dilatih dengan menggunakan beberapa parameter, dan parameter terbaik yang ditemukan dengan Jumlah *epoch* sebanyak 50 menandakan bahwa model dilatih selama 50 iterasi atau putaran penuh dari seluruh dataset pelatihan, dengan menggunakan *batch size* 16, *dropout* dengan nilai 0.3 digunakan sebagai teknik regulasi untuk mengurangi *overfitting*. *Learning rate* sebesar 0.0001 adalah tingkat pembelajaran yang digunakan untuk mengoptimasi bobot model. Dan *optimizer* yang digunakan adalah Adam, yaitu salah satu algoritma optimasi yang efisien dalam mencari bobot yang optimal. Dengan demikian, kesimpulan dari hasil pembahasan klasifikasi hama serangga menggunakan metode CNN ini adalah model berhasil mencapai tingkat akurasi *training* sebesar 93,81% dan akurasi validasi sebesar 81,75%. Sedangkan pada tahap pengujian, model ini berhasil mencapai tingkat akurasi sebesar 80%. Model yang telah dibuat berhasil memprediksi dengan benar sebanyak 109 data *testing* dari total 136 data *testing*. Model ini memiliki performa yang baik dalam mengenali dan mengklasifikasikan hama serangga yang berpotensi berguna dalam upaya pengendalian hama pada pertanian. Meskipun demikian, penelitian ini masih memiliki kekurangan yang perlu diperbaiki pada penelitian berikutnya. Salah satu saran perbaikan adalah mempertimbangkan penggunaan dataset lain dengan jumlah kelas yang lebih banyak dan citra gambar yang lebih baik juga diharapkan dapat meningkatkan akurasi pengujian.

#### REFERENCES

- [1] Z. C. J. Kristiaga, Sutoyo, dan I. M. I. Agastya, "Kelimpahan Serangga Musuh Alami dan Serangga Hama pada Ekosistem Tanaman Cabai Merah (*Capsicum Annum* L.) pada Fase Vegetatif di Kecamatan Dau Kabupaten Malang," *J. Penelit. Pertan. Terap.*, vol. 20, no. 3, hal. 230–236, 2020.
- [2] M. F. N. Syahbani dan N. G. Ramadhan, "Klasifikasi Gerakan Yoga dengan Model Convolutional Neural Network Menggunakan Framework Streamlit," *J. Media Inform. Budidarma*, vol. 7, no. 1, hal. 509–519, 2023, doi: 10.30865/mib.v7i1.5520.
- [3] U. S. Rahmadhani dan N. L. Marpaung, "Klasifikasi Jamur Berdasarkan Genus dengan Menggunakan Metode CNN," *J. Inform. J. Pengemb. IT*, vol. 8, no. 2, hal. 169–173, 2023.
- [4] S. Yuliany, Aradea, dan A. N. Rachman, "Implementasi Deep Learning pada Sistem Klasifikasi Hama Tanaman Padi Menggunakan Metode Convolutional Neural Network (CNN)," *J. Buana Inform.*, vol. 13, no. 1, hal. 54–65, 2022, doi: 10.24002/jbi.v13i1.5022.
- [5] D. Alamsyah dan D. Pratama, "Implementasi Convolutional Neural Networks (CNN) untuk Klasifikasi Ekspresi Citra Wajah pada FER-2013 Dataset," *J. Teknol. Inf.*, vol. 4, no. 2, hal. 350–355, 2020, doi: 10.36294/jurti.v4i2.1714.
- [6] M. Sanjaya dan E. Nurraharjo, "Deteksi Jenis Rempah-Rempah Menggunakan Metode Convolutional Neural Network Secara Real Time," *J. Sains Komput. Inform. (J-SAKTI)*, vol. 7, no. 1, hal. 22–31, 2023.
- [7] R. Abdulhakim, Carudin, dan B. Arif Dermawan, "Analisis dan Penerapan Algoritma Convolutional Neural Network untuk Klasifikasi Kendaraan Prioritas," *J. Sains dan Inform.*, vol. 7, no. 2, hal. 135–144, 2021, doi: 10.34128/jsi.v7i2.335.
- [8] Samuel, K. R. Prilianti, H. Setiawan, P. Mimboro, dan P. Korespondensi, "Perkebunan Sawit Menggunakan Model Convolutional Neural Network (CNN) pada Perangkat Lunak Sistem Informasi the Method of Automatic Tree Detection on Oil Palm Plantation Images Using Convolutional Neural Network (Cnn) Model Using," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 9, no. 7, hal. 1689–1698, 2022, doi: 10.25126/jtiik.202296772.
- [9] A. E. Putra, M. F. Naufal, dan V. R. Prasetyo, "Klasifikasi Jenis Rempah Menggunakan Convolutional Neural Network dan Transfer Learning," *JEPIN (Jurnal Edukasi dan Penelit. Inform.)*, vol. 9, no. 1, hal. 12–18, 2023.
- [10] A. Jakaria, S. Mu'minah, D. Riana, dan S. Hadianti, "Klasifikasi Varietas Buah Kiwi dengan Metode Convolutional Neural Networks Menggunakan Keras," *J. Media Inform. Budidarma*, vol. 5, no. 4, hal. 1309, 2021, doi: 10.30865/mib.v5i4.3166.
- [11] H. Kurniawan dan K. Kusriani, "Klasifikasi Pengenalan Wajah Siswa pada Sistem Kehadiran dengan Menggunakan Metode Convolutional Neural Network," *J. Media Inform. Budidarma*, vol. 7, no. 2, hal. 846–856, 2023, doi: 10.30865/mib.v7i2.5958.
- [12] A. H. Nasrullah dan H. Annur, "Implementasi Metode Convolutional Neural Network Untuk Identifikasi Citra Digital Daun,"



- J. Media Inform. Budidarma*, vol. 7, no. 2, hal. 726–736, 2023, doi: 10.30865/mib.v7i2.5962.
- [13] T. Bariyah, M. Arif Rasyidi, dan Ngatini, “Convolutional Neural Network Untuk Metode Klasifikasi Multi-Label pada Motif Batik,” *Techno.COM*, vol. 20, no. 1, hal. 155–165, 2021.
- [14] M. F. Naufal, J. Siswanto, dan M. G. K. Wicaksono, “Klasifikasi Tulisan Tangan pada Resep Obat Menggunakan Convolutional Neural Network,” *Techno.Com*, vol. 22, no. 2, hal. 508–526, 2023, doi: 10.33633/tc.v22i2.8075.
- [15] U. Khultsum dan G. Taufik, “Komparasi Kinerja DenseNet 121 dan MobileNet untuk Klasifikasi Citra Penyakit Daun Kentang,” *JURIKOM (Jurnal Ris. Komputer)*, vol. 10, no. 2, hal. 558–565, 2023, doi: 10.30865/jurikom.v10i2.6047.
- [16] R. Kurniawan, P. B. Wintoro, Y. Mulyani, dan M. Komarudin, “Implementasi Arsitektur Xception Pada Model Machine Learning Klasifikasi Sampah Anorganik,” *JITET (Jurnal Inform. dan Tek. Elektro Ter.)*, vol. 11, no. 2, hal. 233–236, 2023, doi: 10.23960/jitet.v11i2.3034.
- [17] D. Husen, K. Kusriani, dan K. Kusnawi, “Deteksi Hama pada Daun Apel Menggunakan Algoritma Convolutional Neural Network,” *J. Media Inform. Budidarma*, vol. 6, no. 4, hal. 2103–2110, 2022, doi: 10.30865/mib.v6i4.4667.
- [18] R. Shinta, Jasril, M. Irsyad, F. Yanto, dan S. Sanjaya, “Klasifikasi Citra Penyakit Daun Tanaman Padi Menggunakan CNN dengan Arsitektur VGG-19,” *J. Sains dan Inform.*, vol. 9, no. 1, hal. 37–45, 2023, doi: 10.22216/jsi.v9i1.2175.
- [19] D. Irfansyah, M. Mustikasari, dan A. Suroso, “Arsitektur Convolutional Neural Network (CNN) Alexnet untuk Klasifikasi Hama pada Citra Daun Tanaman Kopi,” *J. Inform. J. Pengemb. IT*, vol. 6, no. 2, hal. 87–92, 2021, [Daring]. Tersedia pada: <http://ejournal.poltektegal.ac.id/index.php/informatika/article/view/2802>
- [20] M. A. Hasan, Y. Riyanto, dan D. Riana, “Grape Leaf Image Disease Classification Using CNN-VGG16 Model,” *J. Teknol. dan Sist. Komput.*, vol. 9, no. 4, hal. 218–223, 2021, doi: 10.14710/jtsiskom.2021.14013.
- [21] R. Magdalena, S. Saidah, N. K. C. Pratiwi, dan A. T. Putra, “Klasifikasi Tutupan Lahan Melalui Citra Satelit SPOT-6 dengan Metode Convolutional Neural Network (CNN),” *J. Edukasi dan Penelit. Inform.*, vol. 7, no. 3, hal. 335–339, 2021, doi: 10.26418/jp.v7i3.48195.
- [22] J. Gatc dan F. Maspriyanti, “Prediksi Parasit Plasmodium pada Citra Mikroskopis Sel Darah Merah dengan Convolutional Neural Networks,” *J. Buana Inform.*, vol. 13, no. 1, hal. 31–41, 2022, [Daring]. Tersedia pada: <https://ojs.uajy.ac.id/index.php/jbi/article/view/5007%0Ahttps://ojs.uajy.ac.id/index.php/jbi/article/view/5007/2603>
- [23] L. Arisandi dan B. Satya, “Sistem Klarifikasi Bahasa Isyarat Indonesia (Bisindo) dengan Menggunakan Algoritma Convolutional Neural Network,” *J. Sist. Cerdas*, vol. 5, no. 3, hal. 135–146, 2022, doi: 10.37396/jsc.v5i3.262.
- [24] F. N. Cahya, N. Hardi, D. Riana, dan S. Hadiyanti, “Klasifikasi Penyakit Mata Menggunakan Convolutional Neural Network (CNN),” *Sist. J. Sist. Inf.*, vol. 10, no. 3, hal. 618–626, 2021, doi: 10.32520/stmsi.v10i3.1248.