

**IMPLEMENTASI CNN UNTUK IDENTIFIKASI SPESIES REPTIL DAN AMFIBI BERBASIS CITRA DIGITAL****Aldo Romulus Napitupulu<sup>1</sup>, Herri Suba L. Tobing<sup>2</sup>, Ng Poi Wong<sup>3</sup>, Kelvin<sup>4\*</sup>**<sup>1,2,3,4</sup>*Fakultas Informatika, Universitas Mikroskil, Medan, Sumatera Utara, 20214, Indonesia***Info Artikel**

Riwayat Artikel :

Diterima : 20 November 2025

Direvisi : 28 November 2025

Diterima : 07 Desember 2025

Diterbitkan : 20 Desember 2025

**Kata Kunci : Identifikasi Citra; CNN; Reptil; Amfibi****Penulis Korespondensi: Kelvin**

Email:

[kelvin.chen@mikroskil.ac.id](mailto:kelvin.chen@mikroskil.ac.id)**Abstrak**

Reptil dan amfibi merupakan dua kelompok hewan vertebrata yang memiliki peran penting dalam ekosistem, namun kerap sulit dibedakan hanya melalui pengamatan visual. Kemiripan warna kulit, bentuk tubuh, dan pola permukaan sering menimbulkan kesalahan identifikasi, terlebih ketika faktor pencahayaan, latar belakang, dan sudut pengambilan gambar bervariasi. Kondisi ini menimbulkan tantangan signifikan bagi proses identifikasi otomatis berbasis citra. Penelitian ini bertujuan mengembangkan model identifikasi berbasis citra dengan memanfaatkan algoritma Convolutional Neural Network (CNN), yang dikenal handal dalam mengekstraksi dan mempelajari fitur visual dari data citra kompleks. Dataset penelitian terdiri dari 6.045 gambar yang mencakup 10 spesies, dikategorikan menjadi dua kelas utama, yaitu reptil dan amfibi. Arsitektur CNN yang dirancang terdiri dari tiga lapisan konvolusi, pooling, dan fully connected, disertai mekanisme callback seperti EarlyStopping dan ModelCheckpoint untuk mencegah overfitting dan mempertahankan bobot terbaik. Hasil pengujian pada data uji menunjukkan akurasi 86,60%, dengan nilai F1-score 0,91 untuk kelas reptil dan 0,73 untuk kelas amfibi. Perbedaan kinerja ini menunjukkan model lebih optimal dalam mengenali reptil dibandingkan amfibi, yang kemungkinan disebabkan oleh distribusi data tidak seimbang dan kompleksitas visual pada amfibi. Secara keseluruhan, penelitian ini membuktikan bahwa CNN dapat digunakan sebagai alat bantu identifikasi otomatis yang efektif untuk membedakan spesies reptil dan amfibi berdasarkan citra visual.

**Jurnal Mahajana Informasi****e-ISSN : 2527-8290****Vol. 10 No. 2 , Desember, 2025 (P135-144)****Homepage :** <https://e-journal.sari-mutiara.ac.id/index.php/7>**DOI :** <https://doi.org/10.51544/jurnalmi.v10i2.6306>

**How To Cite :** Aldo Romulus Napitupulu, Herri Suba L. Tobing, & Kelvin, K. IMPLEMENTASI CNN UNTUK IDENTIFIKASI SPESIES REPTIL DAN AMFIBI BERBASIS CITRA DIGITAL . *JURNAL MAHAJANA INFORMASI*, 10(2), 135–144. Retrieved from <https://e-journal.sari-mutiara.ac.id/index.php/7/article/view/6306>



Copyright © 2025 by the Authors, Published by Program Studi: Sistem Informasi Fakultas Sain dan Teknologi Informasi Universitas Sari Mutiara Indonesia. This is an open access article under the CC BY-SA Licence ([Creative Commons Attribution-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-sa/4.0/)).

**1. Pendahuluan**

Reptil dan amfibi merupakan dua kelompok hewan vertebrata yang berperan penting dalam menjaga keseimbangan ekosistem, terutama sebagai pengendali populasi serangga dan

organisme kecil. Reptil memiliki ciri khas kulit bersisik dan kering, hidup dominan di darat, serta bereproduksi dengan bertelur. Kelompok ini mencakup spesies seperti buaya, kadal, ular, dan kura-kura [1]. Sebaliknya, amfibi mampu hidup di dua habitat—air dan darat—dengan siklus hidup yang dimulai sebagai larva di air sebelum bermetamorfosis menjadi dewasa di darat [2]. Kulitnya yang lembap mendukung pertukaran oksigen, sehingga sangat bergantung pada lingkungan basah.

Meskipun berbeda secara biologis, reptil dan amfibi sering sulit dibedakan pada citra visual, terutama pada jenis-jenis tertentu yang memiliki kemiripan warna, bentuk tubuh, dan pola kulit. Kesulitan ini semakin besar ketika citra diambil dalam kondisi pencahayaan beragam, pose yang bervariasi, latar belakang berantakan, atau terdapat efek blur akibat pergerakan [3]. Habitat keduanya yang kerap tumpang tindih—seperti di daerah perairan, rawa, dan hutan—juga memicu kesalahpahaman dalam identifikasi, terutama tanpa bantuan pengetahuan khusus atau teknologi pendukung.

Dalam ranah pengolahan citra, Convolutional Neural Network (CNN) dikenal sebagai salah satu algoritma deep learning yang efektif meniru mekanisme persepsi visual makhluk hidup. CNN mampu mengekstraksi fitur citra secara otomatis dan telah digunakan dalam berbagai studi identifikasi spesies yang memiliki tingkat kemiripan visual tinggi [4], [5]. Misalnya, penelitian S. K. Dirjen, dkk., berhasil mengidentifikasi 14 jenis reptil dengan akurasi 64,3% berdasarkan ciri bentuk kepala, pola tubuh, warna kulit, dan bentuk mata [1]. Studi Schneider et al. juga membuktikan kemampuan CNN membedakan individu dalam spesies yang sama meskipun terdapat variasi pose, pencahayaan, dan latar belakang [6].

Struktur CNN terdiri dari beberapa lapisan—konvolusi, pooling, dan fully connected—yang secara bertahap mengekstraksi dan menggabungkan fitur penting citra. Keunggulan ini membuat CNN lebih adaptif dibandingkan metode tradisional seperti SVM, karena tidak memerlukan ekstraksi fitur manual dan lebih andal untuk citra kompleks [7], [8].

Berdasarkan potensi tersebut, penelitian ini bertujuan mengembangkan model identifikasi citra yang mampu membedakan spesies reptil dan amfibi secara akurat. Pemanfaatan CNN diharapkan dapat mengatasi tantangan identifikasi akibat kemiripan visual dan variasi kondisi citra, sekaligus memberikan kontribusi terhadap upaya konservasi, edukasi, dan pengembangan teknologi pengenalan spesies di masa mendatang.

## 2. Metode

Pelaksanaan penelitian dibagi ke dalam lima tahap utama:

### 1. Pengumpulan Data

Dataset citra spesies reptil dan amfibi diambil dari Kaggle (“Reptiles and Amphibians Image Dataset”), berjumlah 6.045 citra yang mencakup 10 spesies: bunglon, buaya, katak, tokek, kodok, kadal, salamander, ular, kura-kura, dan iguana.

### 2. Analisis

**Analisis Kebutuhan** — Terdiri dari kebutuhan fungsional (digambarkan dengan *use case diagram*) dan kebutuhan non-fungsional yang dianalisis menggunakan pendekatan PIECES. seperti pada Gambar 1.

**Analisis Proses** — Memodelkan alur pembangunan sistem identifikasi berbasis CNN mulai dari pengumpulan data hingga evaluasi model, divisualisasikan menggunakan *flowchart* seperti pada Gambar 2.

### 3. Perancangan

Membuat rancangan tampilan (*mockup*) antarmuka identifikasi spesies reptil dan amfibi

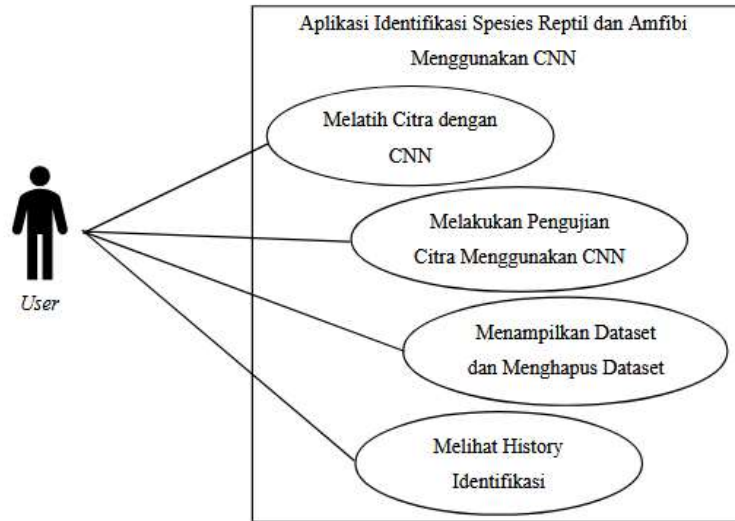
menggunakan Microsoft Word, untuk dokumentasi metodologi tanpa pengembangan aplikasi nyata.

4. **Pengembangan**

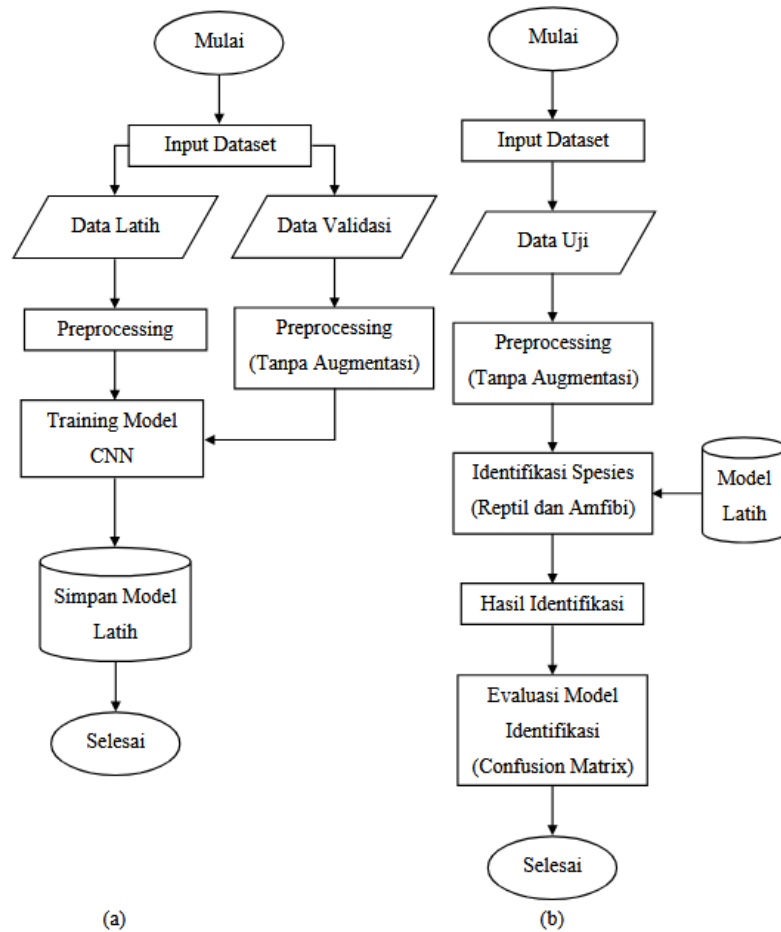
Membangun model CNN menggunakan bahasa pemrograman Python berdasarkan hasil rancangan. Pengembangan fokus pada model identifikasi citra, sementara rancangan tampilan hanya digunakan untuk dokumentasi.

5. **Pengujian**

Menguji kinerja model CNN menggunakan *confusion matrix* dengan metrik akurasi, presisi, *recall*, dan F1-score.



**Gambar 1.** Use Case Sistem Identifikasi Spesies Reptil dan Amfibi



**Gambar 2.** *Flow Chart* Sistem Identifikasi data latih (a) dan data uji (b)

### 3. Hasil

Penelitian ini dilakukan menggunakan bahasa pemrograman Python pada platform Google Colab yang mendukung eksekusi dengan GPU (Tesla T4). Framework yang digunakan adalah TensorFlow dan Keras, dengan bantuan library tambahan seperti NumPy, Matplotlib. Model CNN yang dikembangkan terdiri atas beberapa lapisan konvolusi, batch normalization, max pooling, dan fully connected layer. Gambar 3 memperlihatkan arsitektur model CNN yang digunakan dalam penelitian ini. Model dikompilasi menggunakan fungsi loss `categorical_crossentropy` dan optimizer Adam dengan learning rate sebesar 0.00005. Model dilatih menggunakan data pelatihan dan validasi selama maksimal 15 epoch dengan batch size sebesar 32. Untuk meningkatkan efisiensi pelatihan dan mencegah overfitting, digunakan callback berupa `EarlyStopping` dan `ModelCheckpoint`. Kode konfigurasi pelatihan ditunjukkan pada Gambar 4.

```

from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Flatten, Dense, Dropout, BatchNormalization
from tensorflow.keras.optimizers import Adam

model = Sequential()

# Block 1
model.add(Conv2D(32, (3, 3), activation='relu', input_shape=(224, 224, 3)))
model.add(BatchNormalization())
model.add(MaxPooling2D(pool_size=(2, 2)))

# Block 2
model.add(Conv2D(64, (3, 3), activation='relu'))
model.add(BatchNormalization())
model.add(MaxPooling2D(pool_size=(2, 2)))

# Block 3
model.add(Conv2D(128, (3, 3), activation='relu'))
model.add(BatchNormalization())
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.5))

# Fully connected
model.add(Flatten())
model.add(Dense(1000, activation='relu'))
model.add(Dropout(0.5))

model.add(Dense(1000, activation='relu'))
model.add(Dropout(0.5))

# output layer (10 nodes)
model.add(Dense(10, activation='softmax'))

# compile model
optimizer=Adam(lr=0.0001)
model.compile(optimizer=optimizer,
              loss='categorical_crossentropy',
              metrics=['accuracy'])

```

Gambar 3 Arsitektur CNN

```

[] from tensorflow.keras.callbacks import EarlyStopping, ModelCheckpoint

checkpoint_filepath = '/content/drive/MyDrive/TA/model_checkpoint.keras'

model_checkpoint_callback = ModelCheckpoint(
    filepath=checkpoint_filepath,
    save_best_only=True, # hanya simpan model terbaik berdasarkan monitor
    monitor='val_accuracy',
    mode='max',
    verbose=1
),

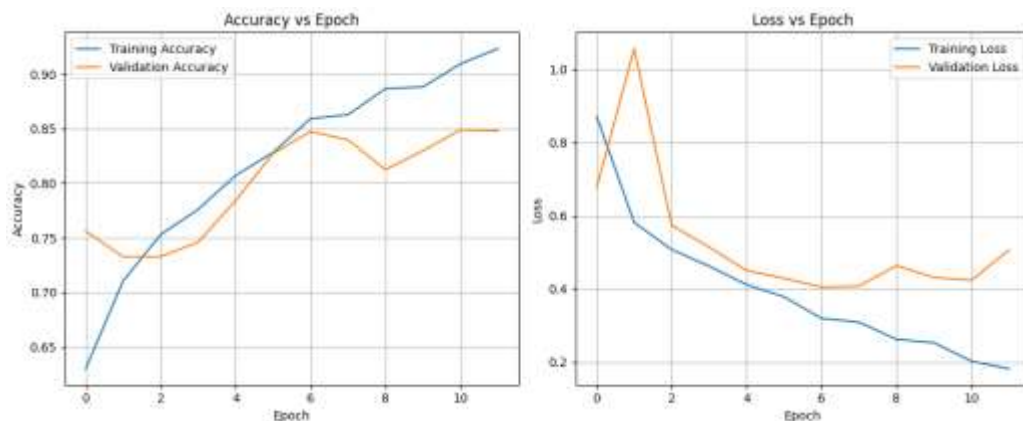
early_stop = EarlyStopping(
    monitor='val_loss',
    patience=5,
    restore_best_weights=True,
    verbose=1
)

[] # training
history = model.fit(
    train_generator,
    epochs=15,
    validation_data=val_generator,
    callbacks=[model_checkpoint_callback, early_stop]
)

```

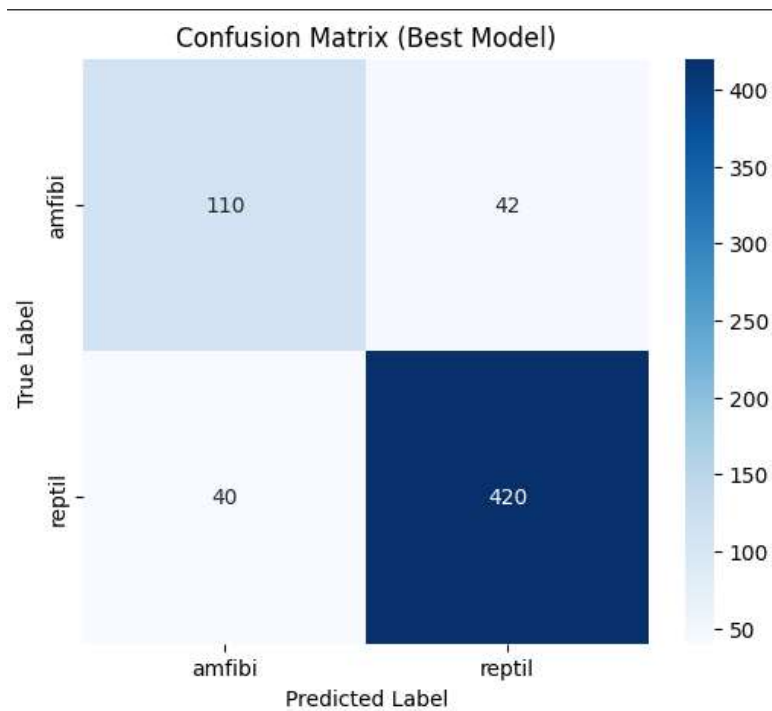
Gambar 4 Kode konfigurasi *callback* dan *training*

Selama proses pelatihan, model mencatat perkembangan nilai akurasi dan loss pada data pelatihan dan validasi setiap epoch. Visualisasi grafik hasil pelatihan ditampilkan pada Gambar 5 yang menunjukkan tren akurasi dan loss terhadap jumlah epoch yang dilalui.



Gambar 5 Grafik akurasi dan *loss* selama pelatihan

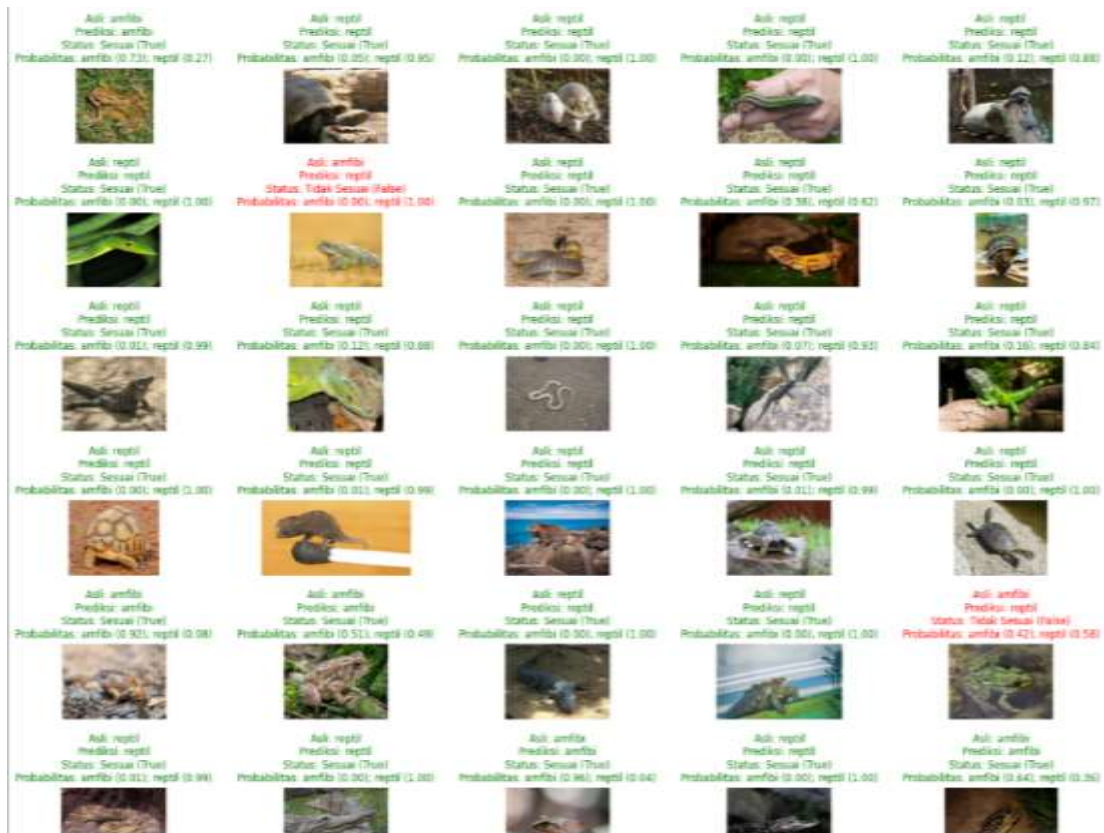
Model terbaik yang tersimpan melalui mekanisme ModelCheckpoint kemudian dievaluasi menggunakan data uji yang tidak digunakan selama pelatihan. Hasil evaluasi menunjukkan nilai akurasi sebesar 86.60% dan nilai loss sebesar 0.4144. Untuk mengetahui kinerja identifikasi pada masing-masing kelas, dibuat *confusion matrix* yang menggambarkan jumlah prediksi benar dan salah dari model terhadap label sebenarnya. Gambar 6 memperlihatkan visualisasi dari *confusion matrix*. Hasil klasifikasi dapat dilihat pada Tabel 1. Visualisasi hasil prediksi model terhadap data uji. Visualisasi ini menampilkan gambar uji bersama dengan label asli, label hasil prediksi, status kecocokan, dan probabilitas dari masing-masing kelas. Contoh visualisasi ditampilkan pada Gambar 7.



Gambar 1 *Confusion Matrix*

**Tabel 1.** Hasil Klasifikasi

Kelas	Precision	Recall	F1Score	Support
Amfibi	73	72	73	152
Reptil	91	91	91	460
Accuracy			87	612
Macro avg	82	82	82	612
Weighted avg	87	87	87	612



Gambar 2 Visualisasi Identifikasi

## 4. Pembahasan

Pada bagian ini dibahas secara mendalam kinerja model utama yang dikembangkan untuk mengidentifikasi spesies reptil dan amfibi. Analisis dilakukan berdasarkan arsitektur model, proses pelatihan, evaluasi performa, serta pengaruh distribusi data dan teknik preprocessing terhadap kemampuan generalisasi model.

### 4.1. Analisis Model Utama

#### Deskripsi Model

Model *Convolutional Neural Network* (CNN) dibangun menggunakan bahasa pemrograman Python dengan *framework* TensorFlow dan Keras. Arsitektur terdiri atas tiga lapisan konvolusi (*Convolutional Layer*) yang masing-masing diikuti oleh *Batch Normalization*, *Max Pooling*, dan *Dropout*. Setelahnya, model diakhiri dengan dua lapisan *fully connected* (*Dense Layer*) untuk klasifikasi dua kelas utama, yakni reptil dan amfibi. Pemilihan tiga lapisan konvolusi bertujuan mencapai keseimbangan antara kedalaman ekstraksi fitur dan kompleksitas model, sehingga risiko *overfitting* dapat diminimalkan mengingat keterbatasan jumlah data pada beberapa kelas. Kombinasi *MaxPooling* dan *Dropout* digunakan untuk mengurangi *overfitting* sekaligus mempercepat pelatihan, sedangkan *Batch Normalization* membantu menstabilkan distribusi aktivasi.

### Analisis Pelatihan

Proses pelatihan dilakukan dengan *batch size* 32 dan jumlah *epoch* maksimal 15, menggunakan fungsi *loss categorical\_crossentropy* serta *optimizer Adam* dengan *learning rate* 0,00005. Mekanisme *EarlyStopping* dengan *patience* 5 digunakan untuk menghentikan pelatihan ketika *loss* validasi tidak mengalami penurunan signifikan, dan *ModelCheckpoint* menyimpan bobot terbaik berdasarkan akurasi validasi tertinggi. Hasil pelatihan menunjukkan akurasi latih meningkat hingga 91% pada *epoch* ke-12, sementara akurasi validasi berada di kisaran 80–85% dengan fluktuasi kecil. *Loss* pelatihan menurun stabil hingga mendekati 0,15, sedangkan *loss* validasi stagnan setelah *epoch* ke-6. Hal ini mengindikasikan tanda awal *overfitting* yang berhasil dikendalikan oleh *EarlyStopping*. Selisih akurasi pelatihan dan validasi relatif kecil, menunjukkan kemampuan generalisasi yang baik.

### Analisis Evaluasi

Model terbaik dari *ModelCheckpoint* diuji pada data uji yang benar-benar baru, menghasilkan akurasi 86,60% dan *loss* 0,4144. Nilai ini menunjukkan kinerja model yang solid untuk data dengan distribusi kelas yang tidak seimbang.

### Analisis Confusion Matrix

Dari 612 citra uji, 420 citra reptil dan 110 citra amfibi terklasifikasi dengan benar. Namun, terdapat 42 citra amfibi yang salah dikenali sebagai reptil, dan 40 citra reptil yang salah dikenali sebagai amfibi. Kesalahan ini umumnya terjadi pada spesies dengan kemiripan visual tinggi, di mana model lebih optimal mengenali reptil karena jumlah data latihnya lebih banyak dan variasi visualnya lebih konsisten.

### Analisis Hasil Klasifikasi

*Classification report* menunjukkan nilai *precision*, *recall*, dan *F1-score* sebesar 0,91 untuk kelas reptil, sedangkan kelas amfibi memperoleh 0,73 untuk ketiga metrik tersebut. Rata-rata *macro avg* tercatat 0,82 dan *weighted avg* sebesar 0,87. Hal ini mengindikasikan kinerja keseluruhan model baik, namun terdapat ketimpangan antar kelas yang disebabkan oleh distribusi data dan kompleksitas visual amfibi.

### Visualisasi Identifikasi

Visualisasi prediksi menunjukkan sebagian besar citra dikenali dengan benar. Kesalahan terutama terjadi pada citra amfibi dengan latar belakang kompleks atau fitur visual yang mirip reptil. Probabilitas prediksi diperoleh dari keluaran *softmax* pada lapisan akhir, sedangkan label asli berasal dari anotasi dataset.

### Simpulan Analisis Model

Model CNN yang dibangun mampu mengidentifikasi reptil dan amfibi dengan akurasi tinggi, khususnya pada kelas reptil. Implementasi tiga lapisan konvolusi, *callback*, dan *preprocessing* data yang tepat berkontribusi pada kemampuan generalisasi yang baik. Meski begitu, performa pada kelas amfibi masih perlu ditingkatkan melalui *data augmentation* tambahan atau modifikasi arsitektur.

## 4.2. Dataset dan Preprocessing

*Dataset* yang digunakan dalam penelitian ini terdiri dari sepuluh kelas spesies reptil dan amfibi, dengan distribusi jumlah data yang bervariasi. Kelas reptil terdiri dari *Chameleon* sebanyak

210 gambar, *Crocodile\_Alligator* 692, *Gecko* 302, *Iguana* 499, *Lizard* 500, *Snake* 500, dan *Turtle\_Tortoise* sebanyak 1.862 gambar, dengan total 4.565 gambar. Sementara itu, kelas amfibi terdiri dari *Frog* sebanyak 499 gambar, *Salamander* 484, dan *Toad* 497, dengan total 1.480 gambar. Jika digabungkan, total *dataset* yang digunakan adalah sebanyak 6.045 gambar.

Distribusi yang tidak seimbang ini berpotensi menyebabkan model menjadi lebih terlatih pada kelas-kelas dominan (reptil) dan kurang akurat dalam mengenali kelas dengan jumlah sampel lebih sedikit (amfibi). Oleh karena itu, sebelum proses pelatihan dimulai, *dataset* terlebih dahulu dibagi menjadi tiga subset, yaitu *training* (70%), *validasi* (20%), dan *pengujian* (10%), menggunakan *script Python* yang melakukan pembagian berdasarkan struktur folder. Pembagian ini dilakukan secara proporsional terhadap masing-masing kelas, dan dikelompokkan ke dalam dua kategori utama, yaitu reptil dan amfibi. Hasil dari pembagian ini adalah, pada data *training* terdapat sebanyak 3.194 gambar reptil dan 1.034 gambar amfibi, pada data *validasi* terdapat 911 gambar reptil dan 294 gambar amfibi, serta pada data *pengujian* terdapat 460 gambar reptil dan 152 gambar amfibi.

Setelah proses pembagian *dataset*, dilakukan augmentasi data di luar proses pelatihan, yang hanya diterapkan pada kelas-kelas amfibi, yaitu *Frog*, *Salamander*, dan *Toad*. Teknik augmentasi yang digunakan meliputi *horizontal flip*, rotasi, dan *scaling*, dengan tujuan menambah variasi visual dan memperkaya representasi fitur pada gambar amfibi. Augmentasi ini diharapkan dapat membantu model mengenali pola-pola yang lebih beragam dan meningkatkan kemampuan generalisasi, khususnya terhadap kelas minoritas.

## 5. Kesimpulan

Berdasarkan hasil penelitian, dapat disimpulkan bahwa model CNN yang dikembangkan menunjukkan performa yang baik dengan akurasi *pengujian* sebesar **86,60%** dan nilai *loss* sebesar 0,4144, yang mengindikasikan kemampuan generalisasi yang cukup baik terhadap data uji. Performa identifikasi terhadap spesies reptil lebih tinggi dibandingkan amfibi, ditunjukkan dengan **nilai F1-score sebesar 0,91 untuk kelas reptil dan 0,73 untuk kelas amfibi**. Hal ini menunjukkan bahwa model lebih mampu mengenali ciri-ciri visual spesies reptil secara konsisten, sedangkan penurunan performa pada kelas amfibi disebabkan oleh jumlah data yang lebih sedikit serta kemiripan visual dan fitur citra yang lebih halus antarspesies, sehingga menyulitkan model dalam membedakan fitur khas masing-masing spesies. Secara keseluruhan, model CNN yang dibangun telah berhasil mengenali dan mengidentifikasi spesies reptil maupun amfibi berdasarkan citra visual.

## 6. Referensi

- [1] S. K. Dirjen, dkk., 2020, Identifikasi Spesies Reptil Menggunakan Convolutional Neural Network (CNN), *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, No. 3, Vol. 1, 899–906, <https://doi.org/10.29207/resti.v4i5.2282>.
- [2] M. Rangkuti, 2024, Pengertian Amfibi: Hewan yang Memiliki Peran dalam Ekosistem, <https://fahum.umsu.ac.id/blog/pengertian-amfibi-hewan-yang-memiliki-peran-dalam-ekosistem/>, diakses tanggal 18 November 2024.
- [3] B. Islam, dkk., 2023, Animal species recognition with deep convolutional neural networks from ecological camera trap images, *Animals*, No. 9, Vol. 13, 1526, DOI: 10.3390/ani13091526.
- [4] L. G. C. Vithakshana dan W. G. D. M. Samankula, 2020, IoT based animal classification system using convolutional neural network, *2020 International Research Conference on Smart Computing and Systems Engineering (SCSE)*, Sri Lanka, Sep. 2020.
- [5] P. Zeng, 2021, Research on Similar Animal Classification Based on CNN Algorithm, *Journal of Physics: Conference Series*, No. 1, Vol. 2132, 012001, <https://doi.org/10.1088/1742->

6596/2132/1/012001.

- [6] E. Kabuga, dkk., 2024, Similarity learning networks uniquely identify individuals of four marine and terrestrial species, *Esa Journal*, No. 10, Vol. 15, <https://doi.org/10.1002/ecs2.70012>
- [7] P. Alkhairi dan A. P. Windarto, 2023, Classification Analysis of Back Propagation-Optimized CNN Performance in Image Processing, *Jurnal Sistem Teknik dan Teknologi Informasi (JOSEIT)*, No. 1, Vol. 2, 1–9, <https://doi.org/10.29207/joseit.v2i1.5015>.
- [8] S. Y. Chaganti, dkk., 2020, Image Classification using SVM and CNN, *2020 International Conference on Computer Science, Engineering and Applications (ICCSEA)*, University of Southern Queensland, March 2020
- [9] R. C. Gonzalez dan R. E. Woods, 2018, *Digital Image Processing*, Ed. 4, Pearson, New York.
- [10] R. Munir, Operasi-operasi Dasar Pengolahan Citra, <https://informatika.stei.itb.ac.id/~rinaldi.munir/Citra/2020-2021/05-Operasi-dasar-pengolahan-citra-2021.pdf>, diakses tanggal 18 Desember 2024.
- [11] R. Hao, dkk., 2021, A Comprehensive Study of Data Augmentation Strategies for Prostate Cancer Detection in Diffusion-Weighted MRI Using Convolutional Neural Networks, *Journal of Digital Imaging*, No. 4, Vol. 34, 862–876, <https://doi.org/10.1007/s10278-021-00478-7>.
- [12] N. Ketkar dan J. Moolayil, 2021, *Deep Learning with Python: Learn Best Practices of Deep Learning Models with PyTorch*, Ed. 2, Apress, New York.
- [13] A. Escontrela, 2018, Convolutional neural networks from the ground up, <https://towardsdatascience.com/convolutional-neural-networks-from-the-ground-up-c67bb41454e1>, diakses tanggal 21 November 2024.
- [14] I. Goodfellow, Y. Bengio, dan A. Courville, 2016, *Deep Learning*, MIT Press, Cambridge.
- [15] C. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, 2018, Activation Functions: Comparison of Trends in Practice and Research for Deep Learning, *arXiv preprint*, <https://doi.org/10.48550/arXiv.1811.03378>.
- [16] K. Karunakaran, 2024, Master the compile() Method in Keras: Your Ultimate Guide to Building AI Models, <https://medium.com/@iitkarthik/master-the-compile-method-in-keras-your-ultimate-guide-to-building-ai-models-053d23165c5a>, diakses tanggal 12 Juni 2025
- [17] K. Kobs, dkk., 2020, SimLoss: Class Similarities in Cross Entropy, <https://arxiv.org/pdf/2003.03182>, , diakses tanggal 11 Juni 2025
- [18] N. A. Ahmed, 2024, What is a confusion matrix in machine learning?, <https://www.datacamp.com/tutorial/what-is-a-confusion-matrix-in-machine-learning>, diakses tanggal 21 November 2024.
- [19] M. D. Kusri, 2020, *Amfibi dan Reptil Sumatera Selatan: Areal Sembilang-Dangu dan Sekitarnya*, Pustaka Media Konservasi, Bogor.
- [20] L. J. Vitt dan J. P. Caldwell, 2013, *Herpetology: An Introductory Biology of Amphibians and Reptiles*, Ed. 4, Academic Press, London.
- [21] D. B. Wake dan M. S. Koo, 2018, Amphibians, *Current Biology Magazine*, No. 21, Vol. 28, R1237–R1241