

## Deteksi Parasit Trypanosoma Evansi Pada Citra Mikroskopis Darah Kerbau Menggunakan Metode Convolutional Neural Network

Since Rolia Tamu Apu<sup>\*1</sup>, Fajar Hariadi<sup>2</sup>, Raynesta Mikaela Indri Malo<sup>3</sup>

<sup>1,2,3</sup>Universitas Kristen Wira Wacana Sumba

e-mail: [\\*1sincetamuapu@gmail.com](mailto:*1sincetamuapu@gmail.com), [2fajar@unkriswina.ac.id](mailto:2fajar@unkriswina.ac.id), [3raynesta@unkriwina.ac.id](mailto:3raynesta@unkriwina.ac.id)

**Abstract** – Trypanosoma evansi is a blood parasite that attacks livestock such as buffalo and can cause significant economic losses. In East Sumba Regency, the parasite detection process is currently still carried out manually through microscopic examination by veterinarians. The method takes a long time and risks producing misdiagnoses such as false positives and false negatives. This study aims to test the accuracy level of the Convolutional Neural Network (CNN) model in automatic image classification, which is one of the methods in deep learning, in detecting Trypanosoma evolution in microscopic images of buffalo blood. The dataset used consisted of 252 images, divided into two classes: normal blood and parasite-infected blood. The preprocessing stage includes RGB to grayscale conversion, image size change to 128×128 pixels, and normalization. The CNN architecture used consists of two convolutional layers, two max pooling, and one fully connected layer. The model was trained using 80% data and validated with 20%, using Adam optimization and EarlyTopping callbacks. Performance evaluation is carried out using confusion matrix and classification metrics. The results showed that the model achieved an accuracy of 98% on the validation data, with a precision of 0.96 and a recall of 1.00 for the Normal class, and a precision of 1.00 and a recall of 0.96 for the Infected class. The F1-score for both classes is 0.98, with the macro average and weighted average also 0.98. These results show that CNN is able to accurately classify microscopic images of buffalo blood and shows that CNN is effective as a diagnostic aid system for image-based parasite detection.

**Keywords** – Deep Learning, Trypanosoma evansi, Buffalo, CNN, Confusion Matrix.

**Abstrak** – Trypanosoma evansi merupakan parasit darah yang menyerang ternak seperti kerbau dan dapat menyebabkan kerugian ekonomi yang signifikan. Di Kabupaten Sumba Timur, proses deteksi parasit saat ini masih dilakukan secara manual melalui pemeriksaan mikroskopis oleh dokter hewan. Metode tersebut membutuhkan waktu yang lama dan berisiko menghasilkan kesalahan diagnosis seperti false positive dan false negative. Penelitian ini bertujuan untuk menguji tingkat akurasi model Convolutional Neural Network (CNN) dalam klasifikasi citra secara otomatis yang merupakan salah satu metode dalam deep learning, dalam mendeteksi Trypanosoma evansi pada citra mikroskopis darah kerbau. Dataset yang digunakan terdiri dari 252 citra, terbagi ke dalam dua kelas: darah normal dan darah yang terinfeksi parasit. Tahapan preprocessing mencakup konversi RGB ke grayscale, perubahan ukuran citra menjadi 128×128 piksel, serta normalisasi. Arsitektur CNN yang digunakan terdiri dari dua lapisan konvolusi, dua max pooling, dan satu lapisan fully connected. Model dilatih menggunakan 80% data dan divalidasi dengan 20%, menggunakan optimasi Adam dan callback EarlyStopping. Evaluasi performa dilakukan dengan confusion matrix dan metrik klasifikasi. Hasil menunjukkan bahwa model mencapai akurasi sebesar 98% pada data validasi, dengan precision 0,96 dan recall 1,00 untuk kelas Normal, serta precision 1,00 dan recall 0,96 untuk kelas Terinfeksi. Nilai F1-score untuk kedua kelas sebesar 0,98, dengan macro average dan weighted average juga sebesar 0,98. Hasil ini menunjukkan bahwa CNN mampu mengklasifikasikan citra mikroskopis darah kerbau secara akurat serta menunjukkan bahwa CNN efektif sebagai sistem alat bantu diagnosis untuk deteksi parasit berbasis citra.

**Kata Kunci** – Deep Learning, Trypanosoma Evansi, Kerbau, CNN, Confusion Matrix.

### I. PENDAHULUAN

Peternakan di Indonesia merupakan sektor yang menjadi pilar penting dalam mendukung pembangunan serta penguatan ekonomi masyarakat. Salah satu jenis ternak yang memiliki peran signifikan dalam kehidupan masyarakat adalah kerbau [1]. Di Kabupaten Sumba Timur, kerbau memegang peranan penting tidak hanya dalam aspek ekonomi, tetapi juga dalam kehidupan sosial. Selain itu, kerbau dalam kehidupan masyarakat di Sumba Timur juga tidak hanya sebagai sumber pangan, tetapi juga sebagai simbol budaya dan ekonomi. Di daerah ini, kerbau digunakan dalam berbagai upacara adat Marapu, seperti pemakaman dan pernikahan, serta dijadikan sebagai indikator status sosial. Dalam konteks ekonomi, peternakan kerbau merupakan salah satu sumber pendapatan utama masyarakat pedesaan, seiring dengan meningkatnya permintaan pasar lokal dan tradisional terhadap daging kerbau [2]. Kabupaten Sumba Timur merupakan salah satu daerah dengan populasi kerbau yang cukup tinggi. Menurut data Badan Pusat Statistik (BPS) tahun 2020, jumlah populasi kerbau mencapai 33.659 ekor dan pada tahun 2021 mencapai 30.905 ekor [3]. Data ini menunjukkan bahwa sektor peternakan memberikan kontribusi yang cukup signifikan terhadap perekonomian di Kabupaten Sumba Timur.

Namun, tingginya potensi ekonomi dan nilai budaya dari peternakan kerbau di Sumba Timur juga diiringi oleh tantangan serius, salah satunya adalah ancaman penyakit yang menyerang kesehatan ternak. Berdasarkan hasil wawancara yang peneliti lakukan pada 15 April 2025 di Dinas Peternakan Sumba Timur dengan Drh. Umbu Yabbu Anggung Praing, M.Si., salah satu penyakit yang sering terdeteksi adalah parasit *Trypanosoma evansi*. *Trypanosoma evansi* merupakan parasit yang ditularkan melalui gigitan lalat pengisap darah yang dapat menimbulkan berbagai gejala klinis seperti demam, anemia, penurunan berat badan, lesu, dan dalam kasus yang parah dapat menyebabkan kematian [4]. Penyakit ini tidak hanya menurunkan produktivitas ternak, tetapi juga menimbulkan kerugian ekonomi yang signifikan bagi para peternak. Menurut data Badan Pusat Statistik Kabupaten Sumba Timur, populasi ternak kerbau mengalami penurunan pada tahun 2024 sebesar 19.549 ekor dibandingkan dengan tahun 2022 sebesar 30.908 ekor [5]. Penurunan ini diduga disebabkan oleh adanya infeksi parasit *Trypanosoma*.

Saat ini, proses deteksi parasit *Trypanosoma* di Dinas Peternakan Sumba Timur masih dilakukan secara manual melalui pemeriksaan mikroskopis oleh dokter hewan. Proses ini memerlukan waktu dan tenaga, serta rentan terhadap kesalahan manusia. Parasit ini juga memiliki kecenderungan menghasilkan hasil pemeriksaan yang bersifat false positive (positif palsu), yaitu ketika darah normal terdeteksi mengandung parasit, maupun false negative (negatif palsu), yaitu ketika darah yang terinfeksi tidak terdeteksi mengandung parasit. Hal ini disebabkan karena parasit biasanya baru terlihat pada tingkat infeksi yang tinggi, sementara laboratorium yang tersedia masih tergolong tipe C dengan standar pemeriksaan yang terbatas hanya pada metode ulas darah.

Berbagai studi terdahulu telah menerapkan metode Convolutional Neural Network (CNN) dalam pengolahan citra mikroskopis maupun citra digital untuk mendeteksi berbagai penyakit. CNN telah digunakan secara luas dalam bidang pertanian, seperti untuk deteksi penyakit pada tanaman padi melalui analisis daun yang terinfeksi, dan dalam bidang medis seperti deteksi parasit malaria dari citra darah mikroskopis. Metode ini terbukti mampu meningkatkan akurasi deteksi, mempercepat proses diagnosis, dan mengurangi ketergantungan pada pemeriksaan manual. Namun demikian, hingga saat ini belum ditemukan studi yang secara khusus menerapkan CNN untuk mendeteksi parasit *Trypanosoma evansi* pada darah kerbau. Hal ini menunjukkan adanya keterbatasan dalam studi terdahulu, terutama dalam konteks spesifikasi objek deteksi dan jenis ternak, serta menjadi celah penelitian yang penting untuk dikembangkan.

Seiring dengan berkembangnya teknologi, khususnya di bidang kecerdasan buatan (Artificial Intelligence), pendekatan modern seperti Deep Learning mulai banyak digunakan dalam pengolahan dan analisis citra. Salah satu cabang Deep Learning yang populer adalah Convolutional Neural Network (CNN), yaitu jaringan saraf tiruan yang dirancang untuk mengenali pola visual dalam gambar[6]. CNN dikenal efektif dalam mendeteksi dan mengenali objek secara otomatis dengan tingkat akurasi yang tinggi [7].

Berdasarkan latar belakang tersebut, rumusan pertanyaan penelitian yang diajukan adalah Seberapa akurat metode Convolutional Neural Network (CNN) dalam mendeteksi parasit *Trypanosoma evansi* pada citra mikroskopis darah kerbau? dan untuk menjawab pertanyaan tersebut, penelitian ini mengajukan hipotesis sebagai berikut: Metode Convolutional Neural Network (CNN) memiliki tingkat akurasi yang tinggi dalam mendeteksi parasit *Trypanosoma evansi* pada citra mikroskopis darah kerbau, sehingga dapat menjadi alternatif metode deteksi yang lebih efisien dan akurat dibandingkan metode manual konvensional.

Penelitian ini diharapkan dapat memberikan alternatif metode deteksi yang efisien dan akurat guna mendukung proses diagnosis laboratorium dalam bidang kesehatan hewan. Selain itu, hasil dari penelitian ini juga diharapkan dapat menjadi dasar pengembangan sistem berbasis kecerdasan buatan yang mampu membantu dalam mendeteksi parasit secara lebih efektif, serta memberikan kontribusi dalam pengendalian penyakit ternak yang berpotensi menimbulkan kerugian ekonomi bagi masyarakat peternak.

## II. PENELITIAN YANG TERKAIT

Beberapa penelitian terdahulu telah menunjukkan keberhasilan penerapan metode Convolutional Neural Network (CNN) dalam klasifikasi citra, baik di bidang kesehatan maupun pertanian. Penelitian yang dilakukan oleh Zebua menunjukkan bahwa metode CNN efektif dalam mendeteksi gejala awal penyakit mulut dan kuku pada sapi berbasis citra, dengan akurasi tertinggi sebesar 99% pada data pelatihan dan 91% pada data pengujian. Hasil ini menunjukkan kemampuan model dalam mengenali pola visual dari gejala penyakit secara tepat [8]. Sementara itu, penelitian oleh Mintarsih et al. juga menerapkan metode CNN untuk mendiagnosis penyakit ayam (Manur), dan menunjukkan akurasi prediksi sebesar 99,94% dalam mengidentifikasi penyakit Coccidiosis. Hal ini menandakan bahwa CNN sangat efektif dalam mendeteksi penyakit berdasarkan citra hewan ternak [9]. Penelitian oleh Zakiyyah dan Kusno menunjukkan bahwa model CNN yang mereka kembangkan berhasil mencapai akurasi sebesar 98% dalam mendeteksi infeksi malaria pada citra darah[10]. Begitu pula dengan penelitian oleh Puteri Dwi dan Setiadi, yang menunjukkan bahwa model CNN mampu mencapai akurasi sebesar 99%, memperkuat bukti efektivitas CNN dalam klasifikasi citra medis [11]. Di bidang pertanian, penelitian oleh Dzaky menunjukkan bahwa CNN berhasil mendeteksi penyakit pada tanaman cabai dengan akurasi lebih dari 90% [12], sementara penelitian oleh Rijal et al. menunjukkan bahwa model CNN mampu mengklasifikasikan penyakit pada tanaman padi dengan tingkat akurasi mencapai 99% [13], dan penelitian oleh Karlos et al. Menunjukkan bahwa model CNN dalam klasifikasi warna tanpa ekstraksi ciri memberikan hasil yang sangat baik, dengan tingkat akurasi mencapai akurasi 89% [14]. namun demikian, hasil berbeda ditunjukkan oleh penelitian Azmi et al. Di mana model CNN yang dikembangkan memperoleh akurasi sebesar 98,75% pada data latih, tetapi hanya 62,5% pada data uji, yang menunjukkan adanya potensi

overfitting—yakni ketika model terlalu menyesuaikan diri terhadap data pelatihan dan kurang mampu melakukan generalisasi terhadap data baru [15]. Secara keseluruhan, berbagai penelitian tersebut menunjukkan bahwa CNN merupakan metode yang sangat potensial untuk menyelesaikan permasalahan klasifikasi citra pada berbagai bidang. Keberhasilan model sangat dipengaruhi oleh kualitas data, pemilihan arsitektur jaringan, serta strategi evaluasi yang tepat agar hasil yang diperoleh tidak hanya tinggi pada data pelatihan, tetapi juga akurat dan stabil pada data pengujian.

Penelitian ini memiliki unsur kebaruan dibandingkan dengan penelitian-penelitian sebelumnya. Meskipun sama-sama menggunakan metode Convolutional Neural Network (CNN), penelitian-penelitian sebelumnya lebih fokus pada deteksi penyakit pada manusia, seperti malaria dan tumor otak, serta pada tanaman, seperti cabai dan padi. Bahkan, terdapat penelitian yang menerapkan CNN untuk klasifikasi citra motif batik. Namun, hingga saat ini, belum ada penelitian yang menerapkan metode CNN untuk deteksi parasit pada ternak kerbau. Penelitian ini secara spesifik menggunakan CNN untuk menganalisis citra darah kerbau dalam mendeteksi keberadaan parasit, yang merupakan pendekatan baru di bidang kesehatan hewan. Fokus pada ternak kerbau sebagai objek utama juga menjadi nilai tambah, mengingat kerbau memiliki peran penting dalam sektor peternakan di Indonesia, tetapi masih jarang diteliti dari aspek teknologi deteksi penyakit berbasis citra.

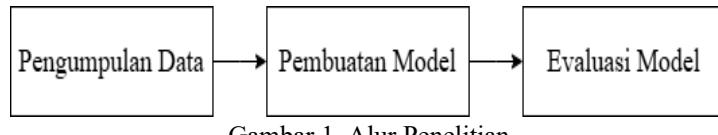
### III. METODE PENELITIAN

#### A. Objek Penelitian

Ternak kerbau merupakan objek utama dalam penelitian ini, peranannya yang sangat penting dalam kehidupan masyarakat Sumba Timur, baik sebagai aset ekonomi maupun simbol sosial dan budaya. Kerbau tidak hanya dimanfaatkan dalam kegiatan pertanian dan upacara adat, tetapi juga mencerminkan status sosial dan nilai tradisi yang kuat di tengah masyarakat. Penelitian ini dilakukan di Dinas Peternakan Kabupaten Sumba Timur, yang merupakan Organisasi Perangkat Daerah (OPD) yang bertanggung jawab atas pengelolaan sektor peternakan yang berlokasi di Jln. Soeharto No. 42. Waingapu, Sumba Timur. Pemilihan tempat didasarkan pada ketersediaan data terhadap sampel darah kerbau yang menjadi objek utama dalam penelitian ini.

#### B. Alur Penelitian

Penelitian ini menggunakan metode *Convolutional Neural Network* (CNN), yaitu algoritma deep learning yang efektif untuk pengolahan citra. CNN mampu mengenali pola visual dan melakukan klasifikasi secara otomatis. Metode ini dipilih karena sesuai untuk mendeteksi keberadaan parasit *Trypanosoma evansi* pada citra mikroskopis darah kerbau. Tahapan pada penelitian ini dimulai dari Pengumpulan Data, Pembuatan Model dan Analisis Akurasi yang dapat dilihat pada Gambar 1.



Gambar 1. Alur Penelitian

Di bawah ini adalah Penjelasan mengenai alur pada penelitian ini :

##### a. Pengumpulan Data

Dalam penelitian ini, data yang dikumpulkan melalui dua metode yaitu Wawancara dan dokumentasi, metode wawancara dilakukan dengan mengajukan pertanyaan-pertanyaan terkait bagaimana cara mengidentifikasi parasit *Trypanosoma evansi* pada saat pemeriksaan mikroskop dan apa saja tantangan yang dihadapi selama pemeriksaan serta berapa lama wakru yang dibutuhkan saat pemeriksaan. Sementara itu metode dokumentasi dilakukan dengan mengumpulkan citra darah kerbau yang sudah melalui proses pewarnaan di bawah mikroskop.

##### b. Pembuatan Model

Pembuatan model dalam penelitian ini dirancang untuk mengenali pola atau ciri khas dari citra darah kerbau yang menunjukkan keberadaan parasit *Trypanosoma evansi*. Tahap awal dalam proses ini adalah preprocessing data, yang bertujuan untuk menyiapkan citra agar lebih optimal saat digunakan oleh model. Citra darah yang semula dalam format RGB diubah menjadi grayscale untuk menyederhanakan informasi warna menjadi intensitas piksel, sehingga mempercepat proses komputasi tanpa menghilangkan informasi penting dari citra. Selanjutnya, citra diubah ukurannya menjadi  $128 \times 128$  piksel, untuk mempertahankan detail visual penting seperti bentuk dan tekstur parasit namun masih efisien untuk diproses oleh jaringan saraf convolutional. Selain itu, seluruh nilai piksel dinormalisasi ke skala 0–1 untuk membantu proses pembelajaran model menjadi lebih stabil dan cepat. Model CNN yang digunakan terdiri dari beberapa lapisan utama, yakni lapisan konvolusi, pooling, dan fully connected. Lapisan konvolusi bertugas mengekstraksi fitur-fitur penting dari citra masukan, seperti tepi, tekstur, dan pola bentuk, dengan menerapkan filter (kernel) yang bergerak melintasi citra. Hasil ekstraksi fitur ini kemudian diringkas oleh lapisan pooling, yang biasanya menggunakan metode max pooling untuk mempertahankan nilai fitur tertinggi dari area lokal, sehingga mengurangi dimensi data sambil mempertahankan informasi penting. Setelah fitur diekstraksi, data dilanjutkan ke lapisan fully connected yang berfungsi sebagai pengklasifikasi. Lapisan ini menerima representasi fitur dari lapisan sebelumnya dan menggunakannya untuk melakukan prediksi,

dalam hal ini membedakan antara citra darah normal dan yang terinfeksi.

Parameter-parameter dalam arsitektur CNN ini dipilih dengan mempertimbangkan keseimbangan antara performa dan efisiensi. Jumlah lapisan konvolusi ditetapkan sebanyak dua hingga tiga lapisan, cukup untuk menangkap fitur penting tanpa membuat model terlalu kompleks, mengingat jumlah data yang tersedia tidak terlalu besar. Ukuran gambar  $128 \times 128$  piksel dipertahankan karena ukuran ini mampu memuat detail parasit yang sangat kecil, sekaligus memberikan fleksibilitas saat membangun arsitektur jaringan. Fungsi aktivasi ReLU (Rectified Linear Unit) digunakan pada setiap lapisan konvolusi karena fungsinya yang efektif dalam mempercepat proses pelatihan. Sedangkan pada lapisan output digunakan fungsi aktivasi sigmoid karena klasifikasi yang dilakukan bersifat biner. Data dibagi menjadi 80% untuk pelatihan dan 20% untuk validasi. Pembagian ini dipilih berdasarkan praktik umum dalam pembelajaran mesin, di mana sebagian besar data dimanfaatkan untuk melatih model agar mampu belajar pola dengan baik, sementara sebagian kecil sisanya digunakan untuk menguji sejauh mana model mampu melakukan generalisasi terhadap data yang belum pernah dilihat sebelumnya.

c. Evaluasi Model

Setelah model CNN selesai dibangun dan dilatih, tahap selanjutnya adalah mengevaluasi performa model dalam mengklasifikasikan citra darah kerbau ke dalam dua kategori, yaitu normal dan terinfeksi. Evaluasi ini bertujuan untuk mengetahui sejauh mana model mampu membuat prediksi yang akurat terhadap data validasi. Salah satu metode yang digunakan adalah confusion matrix, yang menyajikan rincian jumlah prediksi yang benar dan salah pada masing-masing kelas. Dari confusion matrix, dapat diketahui nilai True Positive (citra normal yang diklasifikasikan dengan benar), True Negative (citra terinfeksi yang diklasifikasikan dengan benar), False Positive (citra terinfeksi yang salah diklasifikasikan sebagai normal), dan False Negative (citra normal yang salah diklasifikasikan sebagai terinfeksi). Untuk melengkapi evaluasi, digunakan juga metrik performa lainnya, yaitu akurasi, presisi, recall, dan F1-score. Akurasi menggambarkan proporsi keseluruhan prediksi yang benar dari seluruh data validasi. Presisi mengukur ketepatan model dalam mengidentifikasi citra terinfeksi, sementara recall menunjukkan seberapa baik model dalam menangkap semua citra yang benar-benar terinfeksi. F1-score digunakan sebagai metrik gabungan antara presisi dan recall untuk memberikan penilaian yang seimbang, terutama bila terjadi ketimpangan data antar kelas. Hasil evaluasi ini memberikan gambaran yang komprehensif terhadap kemampuan model dalam mendeteksi parasit Trypanosoma evansi, serta menjadi acuan untuk melakukan penyempurnaan model di tahap selanjutnya.

#### IV. HASIL DAN PEMBAHASAN

##### A. Pengumpulan Data

Pengumpulan data dilakukan di Dinas Peternakan Kabupaten Sumba Timur, dengan total data set sebanyak 252 citra mikroskopis darah kerbau. Data set ini terdiri dari 126 citra darah normal dan 126 citra darah yang terinfeksi parasit Trypanosoma evansi. Data ini bertujuan untuk dijadikan data set untuk membangun model CNN dalam mendeteksi parasit seperti Trypanosoma evansi secara otomatis. Setelah pengambilan data kemudian diolah di Google Colaboratory (Colab).

```
import os, cv2, zipfile
import numpy as np
import matplotlib.pyplot as plt
from google.colab import files
from glob import glob

# Upload dan ekstrak dataset
uploaded = files.upload()
with zipfile.ZipFile(list(uploaded.keys())[0], 'r') as zip_ref:
    zip_ref.extractall("dataset")
```

Gambar 2. Kode Upload File

Langkah awal pengolahan dimulai dengan mengunggah berkas dataset.zip yang berisi kumpulan citra melalui fungsi files.upload() dari pustaka google.colab. Setelah berhasil diunggah, berkas diekstrak menggunakan pustaka zipfile ke dalam folder dataset/. Proses ini memastikan bahwa folder kelas Normal dan Terinfeksi dapat diakses oleh sistem untuk tahap selanjutnya. Tahapan ini penting sebagai awal dalam menyiapkan data sebelum dilakukan *preprocessing* dan pelatihan model CNN.

##### B. Preprocessing Data

Pada tahap ini dilakukan proses *preprocessing* terhadap data set citra mikroskopis darah kerbau yang terbagi ke dalam dua kelas, yaitu Normal dan Terinfeksi. Dataset berada dalam folder dengan struktur dataset/Normal dan dataset/Terinfeksi. Setiap citra diambil dan diproses menjadi bentuk yang seragam sebelum dilatih menggunakan CNN.

```

import os, cv2, zipfile
import numpy as np
import matplotlib.pyplot as plt
from google.colab import files
from glob import glob

# Load dan preprocessing gambar
images, labels = [], []
for label_name, label_val in zip(["Normal", "Terinfeksi"], [0, 1]):
    for path in glob(os.path.join(paths[label_name], "*.*")):
        img = cv2.imread(path)
        if img is not None:
            gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
            resized = cv2.resize(gray, (128, 128))
            images.append(resized)
            labels.append(label_val)

# Ambil 5 gambar dari tiap kelas
normal_imgs = [img for img, lbl in zip(images, labels) if lbl == 0][:5]
terinfeksi_imgs = [img for img, lbl in zip(images, labels) if lbl == 1][:5]

# Tampilkan gambar
plt.figure(figsize=(10, 4))
for i in range(5):
    plt.subplot(2, 5, i + 1)
    plt.imshow(normal_imgs[i], cmap='gray')
    plt.title("Normal")
    plt.axis('off')

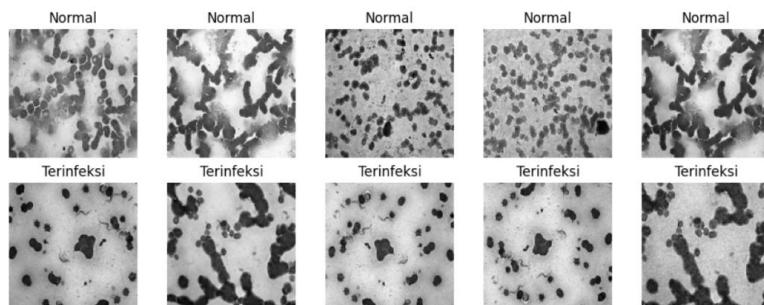
    plt.subplot(2, 5, i + 6)
    plt.imshow(terinfeksi_imgs[i], cmap='gray')
    plt.title("Terinfeksi")
    plt.axis('off')

plt.tight_layout()
plt.show()

```

Gambar 3. Kode Preprocessing

Langkah pertama yaitu membaca setiap file citra menggunakan library OpenCV. Citra kemudian dikonversi dari format RGB ke *grayscale* dengan tujuan menyederhanakan informasi warna yang tidak relevan terhadap deteksi parasit. Setelah itu, seluruh citra di *resize* menjadi ukuran  $128 \times 128$  piksel agar dimensi input ke CNN menjadi konsisten. Setiap gambar juga diberikan label numerik: 0 untuk kelas Normal dan 1 untuk kelas Terinfeksi. Setelah semua citra diproses, data disimpan dalam bentuk array menggunakan NumPy agar dapat digunakan dalam proses pelatihan model. Hasil preprocessing dapat dilihat pada gambar 4.



Gambar 4. Hasil Preprocessing

### C. Splitting Data

Setelah data citra selesai melalui tahap preprocessing, langkah selanjutnya adalah membagi data set menjadi dua bagian, yaitu data latih (training data) dan data validasi (validation data). Pembagian ini bertujuan untuk melatih model menggunakan sebagian besar data (80%), dan menguji performa model terhadap data yang belum pernah dilihat sebelumnya (20%). Pemilihan rasio 80:20 pada pembagian data dilakukan agar sebagian besar data dapat digunakan untuk melatih model, sedangkan sebagian kecil digunakan untuk menguji kemampuan model dalam mengenali data baru. Dengan 80% data latih, model dapat belajar pola dari citra secara optimal. Sementara 20% data validasi cukup untuk mengevaluasi seberapa baik model mengenali citra yang belum pernah dilihat. Rasio ini sering digunakan karena dianggap seimbang dan efektif, terutama ketika jumlah data tidak terlalu besar.

```

# Fungsi split
def split(d):
    random.shuffle(d)
    s = int(0.8 * len(d))
    return d[:s], d[s:]

# Split latih dan validasi
train_n, val_n = split(normal)
train_t, val_t = split(terinfeksi)

# Gabungkan & acak
train = train_n + train_t
val = val_n + val_t
random.shuffle(train)
random.shuffle(val)

# Siapkan data
X_train = np.array([x[0] for x in train]).reshape(-1, 128, 128, 1)
y_train = to_categorical([x[1] for x in train], num_classes=2)
X_val = np.array([x[0] for x in val]).reshape(-1, 128, 128, 1)
y_val = to_categorical([x[1] for x in val], num_classes=2)

# Ringkasan
df = pd.DataFrame({
    "Kategori": ["Normal", "Terinfeksi", "Total"],
    "Latih": [len(train_n), len(train_t), len(train)],
    "Validasi": [len(val_n), len(val_t), len(val)]
})

print("📊 Ringkasan Pembagian Data (80% latih - 20% validasi):")
print(df.to_string(index=False))

📊 Ringkasan Pembagian Data (80% latih - 20% validasi):
  Kategori Latih Validasi
  Normal     100      26
  Terinfeksi 100      26
  Total      200      52

```

Gambar 5. Kode *Splitting*

Dalam proses pelatihan model Convolutional Neural Network (CNN) data set yang digunakan terdiri dari dua kategori, yaitu gambar darah normal dan gambar darah terinfeksi. Proses pembagian dilakukan secara terpisah untuk setiap kategori. Data pada masing-masing kategori diacak terlebih dahulu menggunakan fungsi `random.shuffle()` guna menghindari bias urutan, lalu dibagi menggunakan fungsi `split_data()` yang membagi data berdasarkan indeks proporsional terhadap total data set. Berdasarkan gambar 5 Diperoleh hasil pembagian data set sebagai berikut :

Tabel 1. Hasil Spliting Data

Kategori	Data latih	Data Validasi
Nomal	100	26
Terinfeksi	100	26
Total	200	52

Tabel di atas menunjukkan bahwa dari masing-masing kelas (normal dan terinfeksi), sebanyak 100 data digunakan untuk pelatihan dan 26 data digunakan untuk validasi. Dengan demikian, total data latih adalah 200 gambar, dan total data validasi adalah 52 gambar. Pembagian ini dilakukan untuk memastikan bahwa model memiliki data yang cukup untuk belajar (train) sekaligus memiliki data terpisah untuk mengukur performa model (validasi).

#### D. Model CNN

Model yang dibangun menggunakan pendekatan Sequential dari library Keras dan TensorFlow yang memungkinkan penyusunan layer secara bertingkat.

```
# --- Bangun model CNN ---
model = Sequential([
    Conv2D(32, (3, 3), activation='relu', input_shape=(128, 128, 1)),
    MaxPooling2D(pool_size=(2, 2)),
    Conv2D(64, (3, 3), activation='relu'),
    MaxPooling2D(pool_size=(2, 2)),
    Flatten(),
    Dense(128, activation='relu'),
    Dense(2, activation='softmax')
])
```

Gambar 6. Kodel Model CNN

Adapun library yang digunakan antara lain adalah Sequential, Conv2D, MaxPooling2D, Flatten, dan Dense untuk membentuk struktur CNN. Model dibangun menggunakan pendekatan sequential, yang berarti setiap lapisan ditambahkan secara berurutan. Model diawali dengan lapisan Conv2D yang menggunakan 32 filter dengan ukuran kernel 3x3 dan fungsi aktivasi ReLU. Lapisan ini berfungsi untuk mengekstraksi fitur-fitur dasar dari citra input, seperti pola tepi dan bentuk struktur sel. Setelah fitur awal berhasil diidentifikasi, model dilanjutkan dengan MaxPooling2D berukuran 2x2 yang bertugas mereduksi dimensi spasial dari hasil ekstraksi. Kemudian, ditambahkan Conv2D kedua dengan 64 filter dan kernel 3x3 yang juga menggunakan aktivasi ReLU. Lapisan ini mampu mengenali fitur yang lebih kompleks berdasarkan hasil pooling sebelumnya. Untuk menyederhanakan dimensi hasil ekstraksi tersebut, digunakan kembali MaxPooling2D dengan ukuran yang sama, yaitu 2x2. Hasil dari proses convolution dan pooling kemudian diubah menjadi format satu dimensi menggunakan lapisan Flatten. Transformasi ini memungkinkan data untuk diteruskan ke lapisan fully connected. Selanjutnya, digunakan Dense layer dengan 128 unit dan fungsi aktivasi ReLU sebagai lapisan yang mengolah informasi fitur secara menyeluruh. Pada bagian akhir, terdapat Dense layer dengan 2 unit dan aktivasi Softmax yang menghasilkan dua nilai probabilitas.

#### E. Pelatihan Model

Tahap selanjutnya dalam adalah melakukan pelatihan (*training*) terhadap data latih yang telah disiapkan sebelumnya. Tujuan dari pelatihan ini adalah agar model dapat mempelajari pola-pola visual dari citra darah kerbau, sehingga mampu membedakan antara kondisi Normal dan Terinfeksi.

```
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.callbacks import EarlyStopping

# Kompliasi model
model.compile(optimizer=Adam(0.001), loss='categorical_crossentropy', metrics=['accuracy'])

# Callback untuk menghentikan training saat val_loss tidak membaik
early_stop = EarlyStopping(monitor='val_loss', patience=5, restore_best_weights=True)

# Training model
history = model.fit(
    X_train, y_train,
    validation_data=(X_val, y_val),
    epochs=50,
    batch_size=32,
    callbacks=[early_stop],
    verbose=1
)
```

Gambar 8. Kode Pelatihan Model

Pelatihan model dilakukan selama maksimal 50 epoch dengan ukuran batch sebesar 32. Artinya, dalam satu iterasi pelatihan, model memproses 32 citra sekaligus sebelum memperbarui bobotnya. Model dikompilasi menggunakan optimizer Adam dengan nilai learning rate sebesar 0.001. Optimizer Adam dipilih karena mampu melakukan pembaruan bobot secara adaptif. Untuk mencegah terjadinya overfitting, digunakan fitur callback EarlyStopping yang memantau nilai loss pada data validasi (val\_loss). Apabila dalam lima epoch berturut-turut tidak terjadi penurunan nilai validation loss, maka proses pelatihan akan dihentikan secara otomatis. Selain itu, parameter restore\_best\_weights=True digunakan agar model secara otomatis mengembalikan bobot terbaik yang diperoleh selama pelatihan, bukan bobot terakhir yang mungkin sudah mengalami penurunan performa. Seluruh riwayat pelatihan disimpan dalam variabel history, yang mencatat metrik loss dan accuracy pada data latih dan validasi untuk setiap epoch. Riwayat ini nantinya dapat digunakan untuk memvisualisasikan kinerja model selama proses pelatihan.

#### F. Evaluasi Epoch

Untuk mengevaluasi kinerja model selama proses pelatihan, dilakukan pemantauan terhadap dua metrik utama, yaitu akurasi dan loss pada data pelatihan (training) serta data validasi (validation) untuk setiap epoch.

```
import matplotlib.pyplot as plt

# Plot akurasi dan loss
plt.figure(figsize=(12, 5))

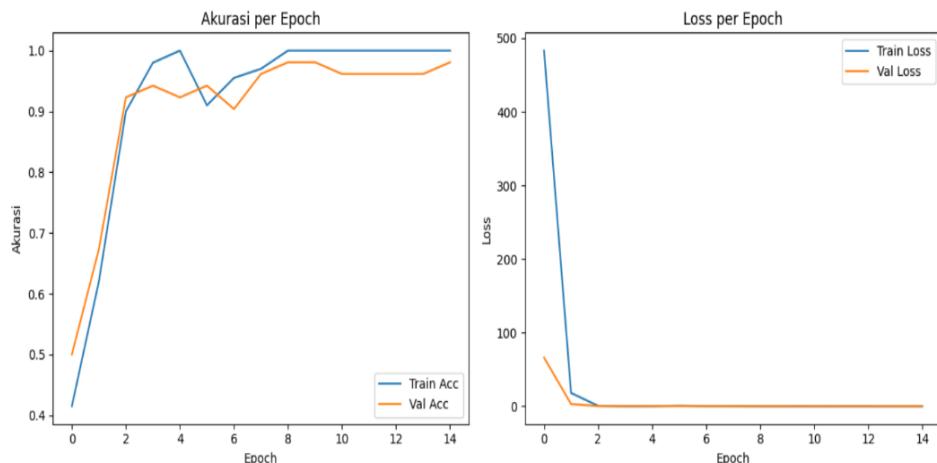
# Akurasi
plt.subplot(1, 2, 1)
plt.plot(history.history['accuracy'], label='Train Acc')
plt.plot(history.history['val_accuracy'], label='Val Acc')
plt.title('Akurasi per Epoch')
plt.xlabel('Epoch')
plt.ylabel('Akurasi')
plt.legend()

# Loss
plt.subplot(1, 2, 2)
plt.plot(history.history['loss'], label='Train Loss')
plt.plot(history.history['val_loss'], label='Val Loss')
plt.title('Loss per Epoch')
plt.xlabel('Epoch')
plt.ylabel('Loss')
plt.legend()

plt.tight_layout()
plt.show()
```

Gambar 9. Kode Evaluasi Epoch

Berdasarkan script di atas diperoleh visualisasi grafik seperti gambar di bawah ini:



Gambar 10. Visualisasi Grafik

Berdasarkan grafik hasil pelatihan model, terlihat bahwa akurasi pelatihan (Train Accuracy) mengalami peningkatan yang signifikan pada tiga epoch awal, hingga mendekati angka 100%. Setelah itu, akurasi tetap tinggi dan relatif stabil di atas 95%, menunjukkan bahwa model mampu mempelajari pola dari data latih dengan sangat baik. Sementara itu, akurasi validasi (Validation Accuracy) juga menunjukkan tren yang positif, dengan peningkatan yang cukup cepat hingga mencapai lebih dari 90% pada epoch ke-3. Meskipun terdapat sedikit fluktuasi pada epoch-epoch selanjutnya, akurasi validasi tetap berada pada kisaran tinggi dan tidak menunjukkan penurunan drastis, yang menandakan bahwa model masih mampu mempertahankan kinerjanya terhadap data yang belum pernah dilihat sebelumnya.

Pada grafik loss, training loss menunjukkan penurunan tajam pada epoch awal dan mencapai nilai mendekati nol sejak epoch ke-2, yang mengindikasikan kesalahan prediksi terhadap data latih sangat kecil. Validation loss juga menurun dengan pola yang serupa dan tetap rendah sepanjang proses pelatihan. Tidak terdapat peningkatan signifikan pada validation loss di epoch-epoch akhir, sehingga dapat disimpulkan bahwa tidak terdapat indikasi overfitting yang jelas pada

model ini. Secara keseluruhan, grafik menunjukkan bahwa model telah mencapai kestabilan dan mampu melakukan generalisasi dengan baik terhadap data validasi.

#### G. Evaluasi Model

Untuk mengevaluasi performa model secara menyeluruh, dilakukan pengujian terhadap data validasi menggunakan metrik evaluasi klasifikasi. Hasil prediksi model dibandingkan dengan label sebenarnya, kemudian dihitung nilai confusion matrix serta metrik evaluasi lainnya seperti *precision*, *recall*, dan *F1-score*. Berikut adalah kode untuk melakukan evaluasi model :

```
import seaborn as sns
from sklearn.metrics import confusion_matrix, classification_report

# Prediksi data validasi
y_pred = model.predict(X_val)
y_pred_classes = np.argmax(y_pred, axis=1)
y_true = np.argmax(y_val, axis=1)

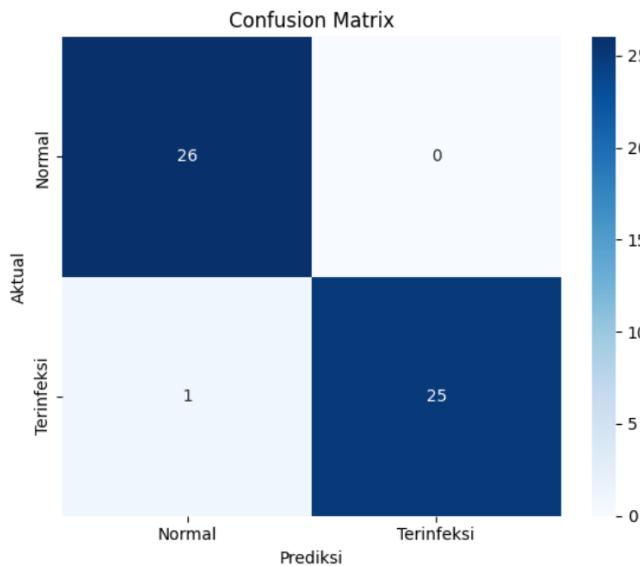
# Tampilkan classification report
print("\nClassification Report:")
print(classification_report(y_true, y_pred_classes, target_names=["Normal", "Terinfeksi"]))

# Confusion matrix
cm = confusion_matrix(y_true, y_pred_classes)

# Visualisasi confusion matrix
plt.figure(figsize=(6, 5))
sns.heatmap(cm, annot=True, fmt='d', cmap='Blues',
            xticklabels=["Normal", "Terinfeksi"],
            yticklabels=["Normal", "Terinfeksi"])
plt.xlabel("Prediksi")
plt.ylabel("Aktual")
plt.title("Confusion Matrix")
plt.tight_layout()
plt.show()
```

Gambar 11. Kode Evaluasi Model

Setelah proses evaluasi dijalankan, diperoleh grafik *confusion matrix* yang menggambarkan hasil prediksi model terhadap data uji.



Gambar 12. Confusion Matrix

Berdasarkan confusion matrix yang pada gambar 4.10, dapat dilihat bahwa model CNN mampu mengklasifikasikan citra darah dengan performa yang sangat baik. Dari total 52 citra pada data validasi, sebanyak 26 citra darah normal berhasil diprediksi dengan benar sebagai Normal, dan 25 citra darah terinfeksi berhasil diklasifikasikan dengan benar sebagai Terinfeksi. Hanya terdapat satu kasus kesalahan prediksi, yaitu ketika citra darah yang sebenarnya Terinfeksi tetapi diprediksi sebagai Normal. Tidak ditemukan kesalahan dalam prediksi citra Normal sebagai Terinfeksi. Hasil ini menunjukkan bahwa model memiliki tingkat akurasi dan keandalan yang tinggi dalam membedakan antara dua kelas

tersebut. Dengan tingkat kesalahan yang sangat rendah, dapat disimpulkan bahwa model telah berhasil mengenali pola visual pada citra darah secara efektif, baik untuk kondisi normal maupun yang menunjukkan terinfeksi Trypanosoma evansi.

Selain menggunakan confusion matrix, evaluasi performa model juga diperkuat melalui classification report yang menyajikan metrik-metrik seperti precision, recall, dan f1-score untuk masing-masing kelas. Berikut di sajikan gambar untuk metrik evaluasi :

Classification Report:				
	precision	recall	f1-score	support
Normal	0.96	1.00	0.98	26
Terinfeksi	1.00	0.96	0.98	26
accuracy			0.98	52
macro avg	0.98	0.98	0.98	52
weighted avg	0.98	0.98	0.98	52

Gambar 13. Hasil Metrik Evaluasi

Berdasarkan gambar 4.11 di atas, classification report menunjukkan bahwa model CNN yang dibangun memiliki performa klasifikasi yang sangat baik terhadap dua kelas, yaitu Normal dan Terinfeksi. Untuk kelas Normal, model memperoleh nilai precision sebesar 0.96, recall 1.00, dan f1-score 0.98. Ini menunjukkan bahwa seluruh citra normal berhasil teridentifikasi dengan benar tanpa adanya kesalahan klasifikasi, walaupun masih ada beberapa prediksi dari kelas lain yang keliru diklasifikasikan sebagai Normal. Sementara itu, pada kelas Terinfeksi, diperoleh nilai precision 1.00, recall 0.96, dan f1-score 0.98, yang berarti seluruh citra yang diprediksi sebagai Terinfeksi benar adanya, namun masih terdapat satu citra terinfeksi yang gagal dikenali dan justru diprediksi sebagai Normal.

Secara keseluruhan, model menunjukkan performa yang sangat optimal dengan nilai akurasi total sebesar 0.98. Selain itu, nilai macro average dan weighted average untuk precision, recall, dan f1-score juga sama-sama mencapai 0.98, yang mengindikasikan bahwa model bekerja secara seimbang dan konsisten terhadap kedua kelas. Hasil ini sejalan dengan confusion matrix sebelumnya, serta memperkuat kesimpulan bahwa model CNN yang dikembangkan memiliki kemampuan generalisasi yang sangat baik dalam mendeteksi keberadaan parasit Trypanosoma evansi pada citra mikroskopis darah kerbau.

## V. KESIMPULAN

Berdasarkan hasil evaluasi terhadap model Convolutional Neural Network (CNN) yang dikembangkan untuk klasifikasi citra mikroskopis darah kerbau, diperoleh performa yang sangat baik dengan tingkat akurasi sebesar 98% pada data validasi. Dari total 52 citra uji, sebanyak 51 citra berhasil diklasifikasikan dengan benar, dan hanya satu citra yang mengalami kesalahan klasifikasi. Confusion matrix menunjukkan bahwa seluruh citra darah Normal (26 citra) berhasil diklasifikasikan dengan benar, sementara pada kelas Terinfeksi, sebanyak 25 dari 26 citra berhasil dikenali dengan benar, dan hanya satu citra yang salah diklasifikasikan sebagai Normal. Pada kelas Normal, model memperoleh nilai precision sebesar 0,96, recall 1,00, dan F1-score 0,98, yang menunjukkan bahwa hampir semua citra normal berhasil dikenali dengan benar tanpa adanya kesalahan klasifikasi. Sedangkan pada kelas Terinfeksi, model memperoleh precision sebesar 1,00, recall 0,96, dan F1-score 0,98, yang berarti seluruh citra yang diprediksi sebagai terinfeksi memang benar adanya, meskipun masih terdapat satu kasus infeksi yang gagal terdeteksi. Secara keseluruhan, nilai macro average dan weighted average untuk precision, recall, dan F1-score semuanya berada di angka 0,98, yang menunjukkan bahwa model memiliki kinerja yang sangat konsisten dan seimbang antar kelas.

Dengan hasil tersebut, dapat disimpulkan bahwa model CNN yang dikembangkan dalam penelitian ini sangat efektif dan andal dalam mendeteksi keberadaan parasit Trypanosoma evansi pada citra mikroskopis darah kerbau. Tingkat kesalahan yang sangat rendah menjadikan model ini potensial untuk diimplementasikan sebagai sistem bantu diagnosis yang cepat dan akurat di bidang kesehatan hewan. Sebagai pengembangan lebih lanjut, sistem deteksi otomatis ini dapat diintegrasikan ke dalam praktik laboratorium di Dinas Peternakan dengan membangun sebuah antarmuka berbasis aplikasi desktop atau web. Aplikasi tersebut dapat dirancang agar petugas laboratorium cukup mengunggah citra hasil mikroskop, dan sistem akan secara otomatis memberikan hasil klasifikasi apakah darah tersebut normal atau terinfeksi. Integrasi ini tidak hanya akan mempercepat proses diagnosis, tetapi juga mengurangi ketergantungan pada pemeriksaan manual dan potensi kesalahan manusia. Dengan pelatihan teknis sederhana, petugas laboratorium dapat dengan mudah mengoperasikan sistem ini sebagai alat bantu, sehingga kualitas pelayanan dan pengawasan kesehatan ternak di lapangan dapat meningkat secara signifikan.

## DAFTAR PUSTAKA

- [1] A. Budiono, K. Muatip, and P. Yuwono, “Hubungan pendidikan dan pengetahuan peternak kerbau tentang pakan dengan keterampilan peternak dalam penyediaan pakan untuk pengembangan ternak kerbau di Kabupaten Pemalang,” *J. Anim. Sci. Technol.*, vol. 4, no. 3, pp. 328–335, 2022.
- [2] A. U. H. Pari, “Pemanfaatan recording untuk meningkatkan manajemen ternak kerbau di Kecamatan Matawai La Pawu Kabupaten Sumba Timur,” *J. Sains Peternak. Indones.*, vol. 13, no. 1, pp. 36–42, 2018.
- [3] Badan Pusat Statistik Nusa Tenggara, “Populasi ternak menurut kabupaten/kota dan jenis ternak di Provinsi Nusa Tenggara Timur (ekor),” BPS, 2020.
- [4] F. Athaillah, K. Amira, and Y. Fahrimal, “11. Diagnostic surra (*Trypanosoma evansi*) on buffalo slaughtered in Banda Aceh and Aceh Besar slaughterhouses based on Card Agglutination Test for *Trypanosoma evansi* (CATT),” *J. Med. Vet.*, vol. 15, no. 1, pp. 75–80, 2022.
- [5] Badan Pusat Statistik Nusa Tenggara Timur, “Populasi Ternak Menurut Kabupaten/Kota dan Jenis Ternak di Provinsi Nusa Tenggara Timur (ekor),” BPS, 2024.
- [6] A. Anhar and R. A. Putra, “Perancangan dan implementasi self-checkout system pada toko ritel menggunakan Convolutional Neural Network (CNN),” *ELKOMIKA J. Tek. Energi Elektr. Tek. Telekomun. Tek. Elektron.*, vol. 11, no. 2, p. 466, 2023.
- [7] Mada Sanjaya W. S., *Deep Learning Convolutional Neural Networks*. Bolabot, 2024.
- [8] H. Zebua, “Pembuatan model deteksi gejala awal penyakit mulut dan kuku pada sapi berbasis citra menggunakan metode Convolutional Neural Network, *Skripsi*, Universitas Jambi, 2023.
- [9] N. Mintarsih, D. D. MS, Y. M. Ariani, and A. M. Hilda, “Implementasi metode Convolutional Neural Network (Cnn) Densenet121 pada diagnosa penyakit ayam (Manur),” *Infotech J. Technol. Inf.*, vol. 10, no. 1, pp. 85–98, 2024.
- [10] A. F. Zakiyyah and M. Kusno, “Identifikasi infeksi penyakit malaria berdasarkan citra darah menggunakan Convolutional Neural Network,” *JSTIE (Jurnal Sarj. Tek. Inform.*, vol. 10, no. 2, p. 76, 2022.
- [11] B. S. E. Dwi and D. R. I. M. Setiadi, “Deteksi tumor otak dengan Metode Convolutional Neural Network,” *J. Eksplora Inform.*, vol. 13, no. 2, pp. 188–197, 2024.
- [12] A. T. R. Dzaky, “Deteksi penyakit tanaman cabai menggunakan metode Convolutional Neural Network,” *e-Proceeding Eng.*, vol. 8, no. 2, pp. 3039–3055, 2021.
- [13] M. Rijal, A. M. Yani, and A. Rahman, “Deteksi citra daun untuk klasifikasi penyakit padi menggunakan pendekatan deep learning dengan model CNN,” *J. Teknol. Terpadu*, vol. 10, no. 1, pp. 56–62, 2024.
- [14] D. Carlos, D. E. Herwindiati, and C. Lubis, “Implementasi algoritma Convolutional Neural Networks untuk klasifikasi jenis cat tembok menggunakan arsitektur MobileNet,” *Build. Informatics, Technol. Sci.*, vol. 6, no. 1, pp. 395–402, 2024.
- [15] K. Azmi, S. Defit, and S. Sumijan, “Implementasi Convolutional Neural Network (CNN) untuk klasifikasi batik tanah liat Sumatera Barat,” *J. Unitek*, vol. 16, no. 1, pp. 28–40, 2023.