

# Sistem Deteksi Jenis Burung Menggunakan Implementasi Yolov11

**<sup>1</sup>Shania Dila Vanesa, <sup>2</sup>Made Ayu Dusea Widya Dara, <sup>3</sup>Umi Mahdiyah**

<sup>1,2,3</sup> Teknik Informatika, Universitas Nusantara PGRI Kediri

E-mail: <sup>1</sup>[shaniashania1717@gmail.com](mailto:shaniashania1717@gmail.com), <sup>2</sup>[madedara@unpkediri.ac.id](mailto:madedara@unpkediri.ac.id), <sup>3</sup>[umimahdiyah@gmail.com](mailto:umimahdiyah@gmail.com)

*Penulis Korespondensi : Shania Dila Vanesa*

**Abstrak**—Identifikasi jenis burung secara manual memerlukan keahlian khusus dan rentan terhadap kesalahan, terutama pada spesies yang memiliki kemiripan morfologi tinggi seperti burung merpati. Penelitian ini bertujuan mengembangkan sistem deteksi otomatis untuk mengidentifikasi burung merpati menggunakan algoritma *YOLOv11* yang dapat menerima input melalui unggahan gambar maupun kamera webcam. Metode penelitian meliputi pengumpulan dataset berisi pengumpulan data, pra-pemrosesan data, pelabelan data, pelatihan model, dan implementasi sistem deteksi , evaluasi sistem menggunakan 60 gambar uji yang terdiri dari 30 gambar merpati dan 30 gambar bukan merpati hasil pengujian menunjukkan sistem mencapai akurasi 93.3%, *Precision* 90,6%, *Recall* 96.7%, dan *F1-Score* 93.6 %. Confusion matrix menunjukkan 29 true positive, 27 true negative, 3 *false positive*, dan 1 *false negative*. Sistem ini memberikan solusi efisien untuk digitalisasi identifikasi burung dan dapat diimplementasikan sebagai alat bantu praktis dalam penelitian ornitologi dan monitoring ekosistem.

**Kata Kunci**—deteksi otomatis, identifikasi burung, merpati, sistem klasifikasi, *YOLOv11*

**Abstract**—*Manual bird species identification requires specialized expertise and is prone to errors, particularly for species with high morphological similarity such as pigeons. This research aims to develop an automatic detection system to identify pigeons using the YOLOv11 algorithm that can accept input through image uploads or webcam camera. The research methodology includes dataset collection, data preprocessing, data labeling, model training, and detection system implementation. System evaluation was conducted using 60 test images consisting of 30 pigeon images and 30 non-pigeon images. Test results show the system achieved 93.3% accuracy, 90.6% precision, 96.7% recall, and 93.6% F1-Score. The confusion matrix shows 29 true positives, 27 true negatives, 3 false positives, and 1 false negative. This system provides an efficient solution for digitizing bird identification and can be implemented as a practical tool in ornithological research and ecosystem monitoring.*

**Keywords**—automatic detection, bird identification, classification system, pigeon, *YOLOv11*



This is an open access article under the CC BY-SA License.

## I. PENDAHULUAN

Indonesia menjadi salah satu negara yang memiliki banyak keanekaragaman spesies burung. Tahap pra-pemrosesan data bertujuan untuk menyiapkan citra masukan agar sesuai dengan kebutuhan pelatihan model *YOLOv11*. [1]. Identifikasi jenis burung secara

manual hingga saat ini masih umum dilakukan di berbagai toko hewan, peternakan, maupun konservasi. Namun, proses ini membutuhkan keahlian khusus dan cukup memakan waktu, terutama karena adanya kemiripan bentuk dan warna pada beberapa spesies burung. Kelemahan identifikasi visual manual dapat berdampak pada kesalahan *Klasifikasi* yang signifikan[2].

Seiring dengan perkembangan teknologi kecerdasan buatan, khususnya dalam bidang computer vision, pendekatan otomatis berbasis deteksi objek mulai digunakan untuk menyelesaikan permasalahan ini [3]. Salah satu metode yang banyak digunakan adalah *You Only Look Once (YOLO)*, yang memungkinkan deteksi objek secara *Realtime* dengan tingkat *Akurasi* yang tinggi .*YOLO* telah diterapkan dalam berbagai bidang seperti pertanian, lalu lintas, dan kesehatan [4].

Beberapa penelitian sebelumnya telah mencoba mendeteksi jenis burung menggunakan metode YOLOv8 mencapai Presisi 96,5% , Recall 98,5%, mAP 99% dalam mengidentifikasi 5 jenis burung berbasis web streamlit [5], kemudian YOLO, kemudian YOLOv5 mencapai Precision sebesar 0,95 dan Recall sebesar 0,93 dalam mendeteksi penggunaan safety helmet [6], pada YOLOv7 mencapai Precision konsisten berada pada angka 0,4 –0,8. Sedangkan nilai Recall maksimum pada angka 0,6 dalam mendekripsi masker wajah [7],

sementara YOLOv3 mencapai Nilai mAP tertinggi diperoleh pada model dengan konfigurasi ke-3, yaitu sebesar 77,92%. Pada konfigurasi ini, kelas good\_mask mencapai nilai AP sebesar 86,15%, sedangkan kelas bad\_mask memperoleh AP sebesar 69,75% dalam mendeteksi pelanggaran penggunaan masker [8], dan pada YOLOv9 mencapai hasil 97.2% untuk Precision, 97.4% untuk Recall, dan 99.4% untuk mAP dalam mendeteksi rambutan matang dan busuk [9].

Berdasarkan penelitian terdahulu dari berbagai versi YOLO tersebut, penelitian ini bertujuan untuk merancang dan mengimplementasikan sistem deteksi otomatis jenis burung menggunakan metode YOLOv11 guna mencapai tingkat akurasi yang akurat. Sistem ini dibangun dalam *platform* web berbasis Flask yang mendukung *input* gambar melalui *upload* maupun kamera *webcam*, guna memberikan solusi yang cepat, praktis, dan efisien bagi pengguna umum .

## II. METODE

Penelitian ini dilakukan secara eksperimental menggunakan pendekatan *deep learning* untuk mendeteksi burung merpati dalam citra digital[10]. Proses penelitian terdiri dari beberapa tahap utama, yaitu pengumpulan data, pra-pemrosesan data, pelabelan data, pelatihan model, evaluasi model, dan implementasi sistem

### A. Pengumpulan Data

Tahap awal dari penelitian ini adalah mengumpulkan dataset gambar burung merpati. Data dikumpulkan dari berbagai sumber terbuka seperti: Situs Kaggle, *Google Images*, dan Dokumentasi pribadi menggunakan kamera hp. Jumlah total data yang diperoleh adalah 150 gambar burung merpati, dengan variasi:

1. Sudut pandang yang berbeda (*frontal*, samping, diagonal)
2. Kondisi pencahayaan berbeda (terang, redup)
3. Latar belakang bervariasi (dalam sangar, *background* pagar, dan alam)
4. Dataset disimpan dalam format .jpg dan .png, dengan struktur direktori dataset/*images*/



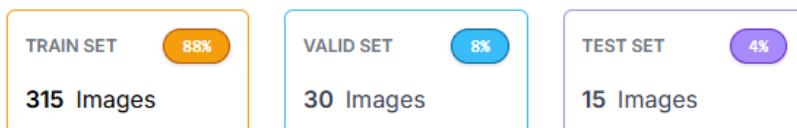
Gambar 1 burung merpati



Gambar 2 Bukan Burung Merpati

#### B. Pra-Premosesan Data

Pra-pemrosesan dilakukan untuk memastikan data yang digunakan sesuai dengan kebutuhan pelatihan model *YOLOv11*. Tahap pra-pemrosesan data bertujuan untuk menyiapkan citra masukan agar sesuai dengan kebutuhan pelatihan model *YOLOv11*. Dataset yang digunakan terdiri dari 360 citra burung yang telah di *anotasi* terbagi menjadi *training* set (315 citra/88%), *Validation* set (30 citra/8%), dan *test* set (15 citra/4%) sebagaimana ditunjukkan pada gambar 1 proses ini dilakukan secara otomatis menggunakan pipeline dari *training* environment berbasis Ultralytics.

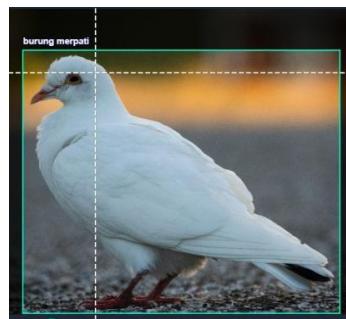


Gambar 3 Dataset Split

Setiap gambar disesuaikan orientasinya (*auto-orient*) dan diubah ukurannya menjadi  $640 \times 640$  piksel. Untuk mencegah *overfitting*, dilakukan *augmentasi* berupa *flipping horizontal*, *cropping*, penyesuaian *brightness*, *blur*, dan *noise*, dengan tiga versi *augmentasi* per gambar.

#### C. Pelabelan Data

Proses pelabelan dilakukan menggunakan *Roboflow*, dengan *bounding box* dibuat manual untuk kelas burung merpati dan bukan burung merpati. Gambar diresize ke  $640 \times 640$  piksel dan diekspor dalam format *YOLO* (.txt) berisi *class* dan koordinat rasio. Proses pelabelan ditunjukkan pada Gambar 3, yang dimana *bounding box* pada objek burung merpati beserta label kelas.



Gambar 4 Proses *Labeling*

#### D. Pelatihan Model

Model *YOLOv11* dilatih menggunakan layanan *cloud Google Colab* dengan pustaka Ultralytics *YOLOv11*. Dataset terdiri dari gambar burung merpati yang telah dianotasi dengan *bounding box*. Pelatihan dilakukan dengan konfigurasi sebagai berikut:

1. jumlah *epoch* sebanyak 100,
2. *batch size* sebesar 16,
3. *learning rate* sebesar 0.001,
4. ukuran gambar  $640 \times 640$  piksel,
5. data dibagi menjadi 80% untuk pelatihan dan 20% untuk *Validasi*,

#### E. Evaluasi Model

Model *YOLOv11* yang telah dilatih kemudian dievaluasi menggunakan beberapa metrik penting, yaitu *Precision*, *Recall*, dan mean Average *Precision* (*mAP*). Evaluasi dilakukan pada 20% data *Validasi* dari total dataset. Dengan rumus perhitungan *confusion matrix*:

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

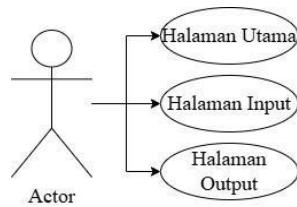
$$Precision = \frac{TP}{TP+FP} \quad (2)$$

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

$$F1 - Score = 2x \frac{Precision \times Recall}{Precision + Recall} \quad (4)$$

#### F. Implementasi Sistem Deteksi

Model *YOLOv11* yang telah dilatih kemudian diimplementasikan ke dalam sebuah sistem berbasis web menggunakan *framework Flask*. Sistem ini dirancang agar mudah digunakan oleh pengguna akhir dan menyediakan dua metode utama untuk melakukan deteksi burung.



Gambar 5 Usecase

Pada gambar 5 menggambarkan alur sistem deteksi penelitian ini yang dimana terdapat 3 bagian utama yaitu :

1. Halaman Utama = Pada halaman Utama terdapat 2 pilihan , pertama ada tentang kami untuk menjelaskan tujuan dibuatnya sistem ini , kedua menuju ke halaman mulai deteksi
2. Halaman *Input* = Pada halaman *input* terdapat 2 pilihan untuk melakukan sistem deteksi, yang pertama dalam bentuk *upload file* dan kedua dalam bentuk *webcam (Realtime)*
3. Halaman *Output* = Pada halaman *Output* menjelaskan hasil dari deteksi dalam bentuk *upload* yang terdapat *bounding box*, nama burung , dan hasil *akurasi*.

### III. HASIL DAN PEMBAHASAN

Bab III ini menyajikan hasil dan pembahasan implementasi sistem deteksi burung, Dataset pengujian disiapkan dengan total 60 gambar yang terdiri dari dua kategori utama. Kategori pertama adalah gambar positif yang berisi 30 gambar burung merpati dalam berbagai kondisi seperti merpati dengan pose dan background yang berbeda. Kategori kedua adalah gambar negatif yang berisi 30 gambar yang tidak mengandung burung merpati, meliputi 10 gambar burung lain seperti gagak dan elang, 10 gambar objek yang mirip burung seperti pesawat terbang, dan 10 gambar background tanpa burung .



Gambar 6 Hasil terdeteksi burung merpati



Gambar 7 hasil yang tidak terdeteksi burung Merpati

Pada gambar 6 dan 7 menunjukkan bahwa sistem sudah bekerja dengan baik, selanjutnya perhitungan performa *confusion matrix* yang ditunjukkan pada tabel 1.

Tabel 1 Pembentukan *Confusion Matrix*

<b>Kondisi</b> <b>Aktual//Prediksi</b>	<b>Prediksi: merpati</b>	<b>Prediksi :Bukan Merpati</b>	<b>Total Aktual</b>
	<b>Merpati</b>		
Aktual: Merpati	29 (TP)	1 (FN)	30
Bukan Merpati	3 (FP)	27 (TN)	30
Total Prediksi	30	30	60

Keterangan:

1. *True Positive* (TP) = 29 → Burung merpati yang diprediksi benar sebagai merpati.
2. *False Negative* (FN) = 3 → Burung merpati yang salah diprediksi sebagai bukan merpati.
3. *False positive* (FP) = 1 → Bukan burung merpati yang salah diprediksi sebagai merpati.
4. *True Negative* (TN) = 27 → Bukan burung merpati yang diprediksi benar sebagai bukan merpati (tidak relevan dalam evaluasi satu kelas vs semua).

Langkah-langkah perhitungan *Confusion Matrix*:

Tabel 2 Perhitungan Metrik Performa

<b>Metrik</b>	<b>Rumus</b>	<b>Perhitungan</b>	<b>Hasil</b>
<i>Accuracy</i>	(TP + TN ) / Total	(29 + 27) / 60	93.3%
<i>Precision</i>	TP / (TP + FP)	29 / (29 + 3)	90.6%
<i>Recall</i>	TP / (TP + FN)	29 / (29 + 1)	96.7%
<i>F1-Score</i>	$2 \times (Precision \times Recall) / (Precision + Recall)$	$2 \times (0.906 \times 0.967) / (0.906 + 0.967)$	93.6 %

Berdasarkan tabel 2 perhitungan metrik di atas, sistem deteksi burung menggunakan model klasifikasi menunjukkan kinerja yang sangat baik. Nilai *accuracy* sebesar 93.3% menunjukkan bahwa secara keseluruhan, sistem mampu mengklasifikasikan gambar dengan benar dalam 93.3% dari 60 kasus. Nilai *Precision* sebesar 90.6% menunjukkan bahwa dari seluruh prediksi positif benar merpati. Hal ini berarti sistem memiliki tingkat kesalahan *false positive* yang relatif rendah, dengan hanya 9.4% prediksi merpati yang sebenarnya bukan merpati. Sedangkan nilai *Recall* sebesar 96.7% menunjukkan bahwa dari seluruh gambar merpati yang sebenarnya, 96.7% berhasil dideteksi dengan benar oleh sistem.

Nilai *Recall* yang tinggi ini mengindikasikan bahwa sistem sangat baik dalam menangkap atau mendeteksi objek merpati dengan tingkat *false negative* yang sangat rendah (hanya 3.3%). Terakhir, nilai *F1-Score* sebesar 93.6% mencerminkan keseimbangan yang sangat baik antara *Precision* dan *Recall*. Nilai *F1-Score* yang mendekati nilai *Recall* menunjukkan bahwa model memiliki kecenderungan sedikit lebih baik dalam mendeteksi merpati (*Recall* tinggi) dibandingkan dengan akurasi prediksi positifnya (*Precision*), namun secara keseluruhan menunjukkan kinerja model yang optimal dalam deteksi burung merpati.

Tabel 3 Analisis Kesalahan

Jenis Kesalahan	Jumlah	Kemungkinan Penyebab	Dampak
<i>False Negative</i>	1 kasus	Pose tidak biasa, oklusi objek, pencahayaan buruk	<i>Missed</i>
<i>False positive</i>	3 kasus	Burung atau benda mirip merpati	<i>False</i>

Berdasarkan pada tabel 3 Sistem deteksi merpati yang dikembangkan menunjukkan performa yang sangat memuaskan berdasarkan evaluasi komprehensif terhadap 60 sampel data. Dengan akurasi mencapai 93.3%, sistem mampu mengklasifikasikan gambar dengan benar dalam lebih dari 9 dari 10 kasus, menunjukkan tingkat reliabilitas yang tinggi untuk implementasi praktis. Tingkat kesalahan yang sangat rendah sebesar 6.7% dengan total hanya 4 kesalahan membuktikan robustnya sistem yang dibangun analisis kesalahan menunjukkan 1 kasus false negative dimana sistem gagal mendeteksi,

merpati yang sebenarnya ada, dan 3 kasus *false positive* dimana sistem salah mengklasifikasikan objek non-merpati sebagai merpati. Kesalahan-kesalahan ini umumnya disebabkan oleh faktor-faktor seperti pose merpati yang tidak biasa, adanya oklusi pada objek, kondisi pencahayaan yang buruk, resolusi gambar yang rendah, adanya objek mirip burung, bayangan, noise pada gambar, atau kemungkinan overfitting model. Kekuatan utama sistem terletak pada nilai *Recall* yang sangat tinggi sebesar 96.7%,

mengindikasikan kemampuan deteksi yang superior dengan tingkat missed detection yang minimal. Hal ini sangat krusial dalam aplikasi praktis dimana pentingnya tidak melewatkannya objek merpati yang ada. Nilai *Precision* sebesar 90.6% juga menunjukkan bahwa mayoritas prediksi positif sistem adalah akurat meskipun masih terdapat beberapa *false positive* yang dapat ditoleransi. *F1-Score* sebesar 93.6% mencerminkan keseimbangan optimal antara *Precision* dan *Recall*, menunjukkan bahwa sistem tidak mengalami trade-off signifikan antara kedua metrik tersebut. Dengan performa yang telah dicapai, sistem deteksi merpati ini siap untuk diimplementasikan dengan tingkat akurasi dan reliabilitas yang tinggi.

#### IV. KESIMPULAN

Tujuan penelitian untuk mengembangkan sistem deteksi jenis burung menggunakan implementasi YOLOv11 di toko burung telah berhasil dicapai. Sistem yang dikembangkan mampu menjawab kebutuhan otomatisasi identifikasi burung merpati di lingkungan komersial toko, mengurangi ketergantungan pada proses manual yang memakan waktu dan rentan terhadap kesalahan manusia. Implementasi algoritma YOLOv11 terbukti efektif dalam mengidentifikasi jenis burung secara *real-time* dengan tingkat akurasi yang memadai untuk aplikasi praktis di toko burung. Penelitian ini berkontribusi pada kemajuan bidang teknik industri melalui pengembangan.

#### DAFTAR PUSTAKA

- [1] DHIKA LAKSONO, “Seluk Beluk Hutan Tropis,” in *Paru - Paru Dunia*, Cv Media Edukasi Creative, 2022, p. 64. [Online]. Available: [https://books.google.co.id/books?Id=eojqceaaqbaj&dq=Indonesia+menjadi+salah+satu+negara++yang+memiliki+banyak+akan+keanekaragaman+spesies+hewan+dan+tumbuhan+peringkat+keatas+di+dunia+&lr=&hl=id&source=gbs\\_navlinks\\_s](https://books.google.co.id/books?Id=eojqceaaqbaj&dq=Indonesia+menjadi+salah+satu+negara++yang+memiliki+banyak+akan+keanekaragaman+spesies+hewan+dan+tumbuhan+peringkat+keatas+di+dunia+&lr=&hl=id&source=gbs_navlinks_s)
- [2] M. Yusuf, D. Kurniawan, and T. Agustin, “BUATAN BERBASIS CNN,” no.

- November, pp. 355–368, 2024.
- [3] I. Fatahna, P. Desi, K. Sari, A. N. Kamilah, R. Wulanningrum, and C. Utomo, “Implementasi Computer Vision Terhadap Jenis Kualitas Pisang Susu Menggunakan Metode yolov8n Berbasis webapps,” *STAINS*, vol. 4, pp. 1–10, 2025.
  - [4] Q. Aini, N. Lutfiani, H. Kusumah, and M. S. Zahran, “DETEKSI DAN PENGENALAN OBJEK DENGAN MODEL MACHINE LEARNING : MODEL YOLO,” vol. 6, no. 2, pp. 192–199, 2021.
  - [5] Y. P. Iswoyo, R. Wulanningrum, and A. Bagus, “Identifikasi Jenis Burung Menggunakan Yolo8 Berbasis Web Streamlit,” vol. 8, pp. 8–15, 2024.
  - [6] N. Ainun, K. Dira, A. Khairunnisa, S. Aras, and A. P. Data, “Deteksi Penggunaan Safety Helmet Menggunakan yolov5,” vol. 07, pp. 74–77, 2023.
  - [7] V. No, J. Hal, and R. G. Guntara, “Pemanfaatan Google Colab Untuk Aplikasi Pendekripsi Masker Wajah Menggunakan Algoritma Deep Learning yolov7 Jurnal Teknologi Dan Sistem Informasi Bisnis,” vol. 5, no. 1, pp. 55–60, 2023.
  - [8] M. Christopher, “Aplikasi Framework YOLO V . 3 untuk Deteksi Pelanggaran Penggunaan Masker,” vol. 9, no. 4, 2023.
  - [9] F. Akbar, N. Muhammad, and Z. Amrulloh, “Deteksi Rambutan Matang dan Busuk Menggunakan Algoritma yolov9 Detection of Ripe and Rotten Rambutan Using the yolov9 Algorithm,” no. 1, pp. 53–59, 2025, doi: 10.26798/jiko.v9i1.1382.
  - [10] A. N. Inaya, A. U. Rahma, M. Jannah, L. R. K. Arafah, L. Ishak, and M. R. Edy, “Klasifikasi Citra Dengan Pendekatan Transfer Learning Pada Gambar Fauna Terbang,” vol. 7, no. 1, pp. 85–89, 2024.