



ARTICLE

## Segmentasi Berbasis Warna Untuk Pengelompokan Kualitas Cacing ANC Menggunakan YOLOv8

### Color Based Segmentation For ANC Worm Quality Grouping Using YOLOv8

Firman Nurdiyansyah,<sup>\*1</sup> Ismail Akbar,<sup>2</sup> dan Lionardi Ursaputra<sup>2</sup>

<sup>1</sup>Bisnis Digital, Fakultas Ekonomi dan Bisnis, Universitas Widya Gama, Malang, Indonesia

<sup>2</sup>Teknik Informatika, Fakultas Teknik, Universitas Widya Gama, Malang, Indonesia

\*Penulis Korespondensi: firmannurdiyansyah7@gmail.com

(Disubmit 05-10-24; Diterima 02-01-25; Dipublikasikan online pada 05-02-25)

#### Abstrak

Mengembangkan metode otomatisasi dalam pengelompokan kualitas cacing African Night Crawler (ANC) menggunakan model YOLOv8 yang didukung oleh segmentasi berbasis warna. Metode manual yang selama ini digunakan dalam menilai kualitas cacing sering kali memakan waktu dan cenderung tidak konsisten, sehingga pendekatan berbasis teknologi diperlukan untuk meningkatkan efisiensi dan akurasi. Penelitian ini dimulai dengan pengumpulan dataset gambar cacing ANC, yang kemudian dianotasi berdasarkan tiga kelas utama, yaitu merah, putih, dan putih-biru. Dataset ini diproses melalui langkah-langkah preprocessing untuk memastikan kualitas data yang konsisten, kemudian dibagi menjadi training set (80%), validation set (15%), dan test set (5%). Model YOLOv8 diterapkan untuk mendeteksi dan mengklasifikasikan objek, dengan arsitektur yang terdiri dari Backbone, Neck, dan Head yang dirancang untuk mengoptimalkan deteksi multi-skala. Hasil penelitian menunjukkan bahwa model YOLOv8 memiliki performa yang sangat baik, dengan mean Average Precision (mAP) rata-rata sebesar 93.8% pada training set, 94.0% pada validation set, dan 95.0% pada test set. Nilai precision mencapai 95.6% pada training set, sementara recall mencatat 85.4%, menghasilkan F1-Score sebesar 90.3%. Hasil ini menunjukkan bahwa model tidak hanya akurat dalam mendeteksi objek yang relevan tetapi juga memiliki sensitivitas yang baik terhadap variasi data. Kesimpulannya, penggunaan YOLOv8 untuk segmentasi berbasis warna pada cacing ANC memberikan solusi yang efisien dan akurat dalam mengklasifikasikan kualitas cacing. Selain mendukung otomatisasi dalam industri peternakan, penelitian ini juga membuka peluang penerapan lebih lanjut dalam sektor agrikultur lainnya, dengan rekomendasi untuk meningkatkan generalisasi model pada variasi data yang lebih kompleks di kondisi dunia nyata.

**Kata kunci:** YOLOv8; Segmentasi Berbasis Warna; African Night Crawler (ANC); Deteksi Objek; Klasifikasi Kualitas Cacing

#### Abstract

Develops an automated method for classifying the quality of African Night Crawler (ANC) worms using the YOLOv8 model supported by color-based segmentation. Traditional manual methods for assessing worm quality are time-consuming and inconsistent, highlighting the need for a technology-based solution to improve efficiency and accuracy. The research involves collecting ANC worm image datasets, annotating them into three main classes—red, white, and white-blue—and preprocessing the data to ensure consistency before dividing it into training (80%), validation (15%), and test (5%) sets. The YOLOv8 model, with its Backbone, Neck, and Head architecture, was employed to optimize multi-scale object detection. The results demonstrate that YOLOv8 achieved outstanding performance, with mean Average Precision (mAP) of 93.8%, 94.0%, and 95.0% on the training, validation, and test sets, respectively. Precision reached 95.6% on the training set, while recall was 85.4%, yielding an F1-Score of 90.3%. These findings indicate that the model is both accurate in detecting relevant ob-

jects and sensitive to variations in the data. This research not only offers an efficient and accurate solution for classifying ANC worm quality but also supports automation in the livestock industry and lays the groundwork for further applications in other agricultural sectors. Future research should focus on improving the model's generalization to handle complex real-world data variations effectively.

**KeyWords:** YOLOv8; Color-Based Segmentation; African Night Crawler (ANC); Object Detection; Worm Quality Classification

## 1. Pendahuluan

Kualitas produk peternakan menjadi perhatian utama di era modern, seiring meningkatnya permintaan global terhadap hasil ternak yang tidak hanya bernilai ekonomis tinggi, tetapi juga berperan dalam mendukung keberlanjutan lingkungan [1]. Salah satu komoditas yang semakin mendapatkan perhatian adalah cacing *African Night Crawler* (ANC). Cacing ANC memiliki potensi luar biasa sebagai penghasil kompos berkualitas tinggi [2], yang mendukung praktik pertanian organik, serta sebagai sumber protein alternatif untuk pakan ternak [3]. Dalam industri peternakan cacing, memastikan kualitas cacing ANC secara konsisten menjadi faktor kunci untuk mempertahankan nilai pasarnya.

Proses evaluasi kualitas cacing ANC saat ini umumnya dilakukan secara manual, yang tidak hanya memakan waktu tetapi juga cenderung bersifat subjektif karena bergantung pada kemampuan individu dalam mengidentifikasi karakteristik cacing. Metode manual ini sering kali menghasilkan penilaian yang tidak konsisten, terutama ketika diterapkan pada skala produksi besar. Oleh karena itu, diperlukan pendekatan otomatis berbasis teknologi untuk meningkatkan akurasi dan efisiensi dalam pengelompokan kualitas cacing ANC [4].

Teknologi berbasis kecerdasan buatan (*Artificial Intelligence*) telah menjadi solusi inovatif dalam berbagai sektor [5], termasuk peternakan cacing. Salah satu metode yang menjanjikan adalah segmentasi berbasis warna, yang mampu mengidentifikasi dan mengelompokkan objek berdasarkan karakteristik visual tertentu [6]. Cacing ANC memiliki pola warna yang dapat digunakan sebagai indikator kualitasnya. Dengan integrasi *Artificial Intelligence*, analisis visual ini dapat dilakukan dengan lebih cepat dan akurat, menggantikan proses manual yang kurang efisien [7].

Metode segmentasi berbasis warna dalam berbagai aplikasi agrikultur dan peternakan telah banyak diterapkan, salah satunya penggunaan algoritma YOLO (*You Only Look Once*) telah terbukti menjadi solusi yang efektif khususnya dalam deteksi dan pengelompokan objek berbasis citra. Penelitian terdahulu telah menunjukkan keunggulan YOLO dalam mendeteksi hama tanaman dengan akurasi tinggi, seperti yang dilakukan oleh [8], yang menggunakan YOLO untuk mendeteksi hama pertanian dengan mAP mencapai 92.42% dan kecepatan deteksi yang tinggi [8]. Selain itu, Ahmad *et al.* [9] menggunakan YOLOv5 untuk mendeteksi dan mengklasifikasikan hama tanaman dengan mAP hingga 98.3%, menunjukkan potensi tinggi algoritma ini dalam mengenali objek dalam kondisi lapangan nyata.

Penelitian [10] mengembangkan model "*YOLO-pest*" yang memanfaatkan YOLOv4 untuk mendeteksi hama kecil dan multi-kelas dengan peningkatan akurasi yang signifikan melalui teknik fusi fitur. Penelitian lain oleh [11] menggunakan YOLO untuk mendeteksi penyakit tanaman secara *real-time*, memfasilitasi identifikasi dini dengan tingkat akurasi dan efisiensi yang tinggi. Selain itu, [12] mengembangkan "*Pest-YOLO*" untuk mendeteksi dan menghitung hama kecil dengan kepadatan tinggi, yang menunjukkan peningkatan signifikan dalam metrik deteksi, menjadikannya alat yang relevan untuk aplikasi agrikultur skala besar.

Penelitian terdahulu juga mencatat keberhasilan penggunaan YOLOv3 dalam mengenali dan mengklasifikasikan buah-buahan berdasarkan citra digital, sebagaimana dilaporkan oleh [13], yang menyoroti potensi besar YOLO dalam segmen agrikultur. [14] juga menggunakan YOLO untuk deteksi penyakit tanaman dengan kecepatan pemrosesan *real-time*, menunjukkan keunggulan YOLO dibandingkan metode lain. [15] memadukan YOLOv4 dengan CNN untuk mendeteksi penyakit pada daun apel dan tomat, menghasilkan efisiensi tinggi dalam klasifikasi berbasis warna.

Penelitian-penelitian ini menunjukkan perkembangan signifikan dalam penggunaan YOLO dan algoritma berbasis *deep learning* lainnya untuk mendukung solusi otomatisasi di bidang agrikultur, yang menjadi dasar kuat untuk implementasi YOLOv8 dalam aplikasi yang lebih spesifik seperti pengelompokan kualitas cacing ANC. Penelitian ini bertujuan untuk mengembangkan metode pengelompokan kualitas cacing ANC

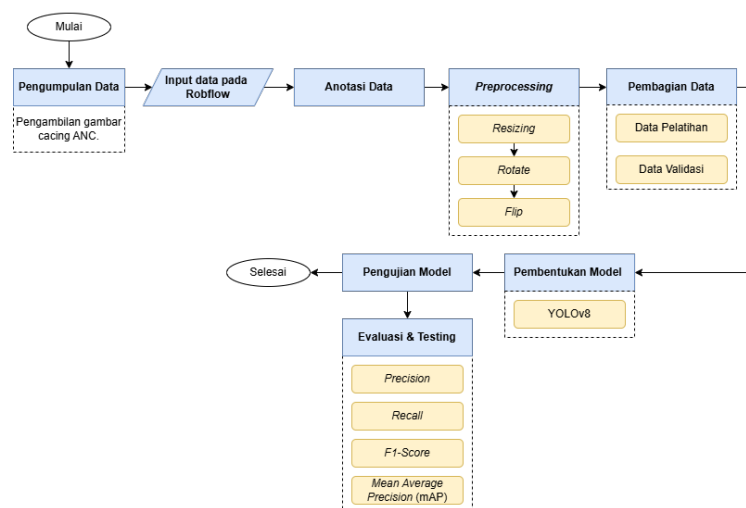
yang didukung oleh segmentasi berbasis warna dan algoritma YOLOv8. Dengan pendekatan ini, penelitian diharapkan tidak hanya meningkatkan efisiensi evaluasi kualitas cacing ANC tetapi juga memperkenalkan solusi otomatis yang relevan bagi peternak cacing modern. Kebaruan penelitian ini terletak pada integrasi teknologi segmentasi berbasis warna dengan algoritma YOLOv8 yang dioptimalkan untuk kebutuhan spesifik analisis visual cacing ANC. Hasilnya diharapkan dapat memberikan kontribusi signifikan dalam mendorong inovasi peternakan berbasis teknologi modern.

## 2. Metode

Penelitian ini dilakukan dengan tahapan yang sistematis untuk memperoleh hasil yang diharapkan, yaitu membangun sistem deteksi cacing *African Night Crawler* (ANC) menggunakan model YOLOv8. Gambar alur tahapan penelitian disajikan pada Gambar 1.

### 2.1 Pengumpulan Data

Data gambar cacing *African Night Crawler* (ANC) dikumpulkan menggunakan kamera *smartphone* di lokasi penelitian CV. Messi Putra Jaya. Gambar diambil dalam berbagai kondisi pencahayaan (terang, redup, gelap) untuk memastikan variasi dataset. Variasi warna cacing yang diperoleh mencakup tiga kategori utama: putih kebiru-biruan, putih, dan merah. Setiap kategori diwakili oleh minimal 500 gambar berkualitas tinggi.



Gambar 1. Tahapan Penelitian

### 2.2 Input Data ke Roboflow dan Anotasi Data

Setelah data terkumpul, langkah berikutnya adalah mengunggahnya ke *Roboflow*, sebuah platform *online* yang membantu mengelola dataset untuk proyek pembelajaran mesin [16]. *Roboflow* menyediakan berbagai alat yang memudahkan proses seperti memberi label (anotasi) yang disajikan pada Gambar 2, meningkatkan variasi data (*augmentasi*), dan mempersiapkan data untuk digunakan pada model [17]. Pada tahap anotasi, setiap gambar diberi tanda warna untuk menunjukkan objek penting yaitu cacing ANC. Setiap kelas diberi label spesifik, dengan garis anotasi berwarna merah untuk kelas "merah", ungu untuk "putih", dan hijau cerah untuk "putih-biru". Hal ini dilakukan agar model dapat mempelajari karakteristik utama dari dataset yang digunakan.

### 2.3 Preprocessing

Pada tahap *preprocessing*, data yang telah dibagi ke dalam subset mengalami serangkaian proses untuk memastikan kualitas dan konsistensinya sesuai dengan kebutuhan model pembelajaran mesin [18]. Langkah pertama adalah *resizing*, yaitu mengubah ukuran gambar agar sesuai dengan format *input* yang dibutuhkan oleh model, sehingga data dapat diproses dengan efisien [19]. Selanjutnya, dilakukan *rotation*, yaitu memutar gambar ke berbagai sudut untuk meningkatkan variasi data, sehingga model dapat mengenali

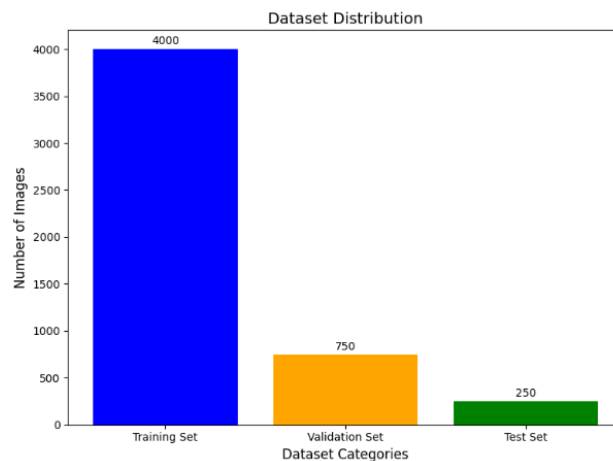


Gambar 2. Proses Anotasi Data

pola yang lebih beragam dari sudut pandang yang berbeda [20]. Proses terakhir adalah *flip*, yaitu membalik gambar secara horizontal atau vertikal, yang bertujuan untuk menambah variasi data tanpa mengubah makna atau informasi dalam gambar [21]. Tahapan *preprocessing* ini dirancang untuk meningkatkan performa model dengan memastikan data lebih representatif dan mencakup berbagai kemungkinan skenario di dunia nyata.

#### 2.4 Pembagian Data

Setelah melalui proses *preprocessing*, dataset mentah yang awalnya berjumlah 140 gambar diolah dan ditingkatkan melalui berbagai teknik augmentasi hingga menghasilkan total 5.000 gambar. Dataset ini kemudian dibagi menjadi tiga kelompok utama untuk memastikan hasil yang akurat dan andal, seperti yang disajikan pada Gambar 3. Pembagian dataset meliputi *training set*, *validation set*, dan *test set*, yang masing-masing digunakan untuk melatih, mengevaluasi, dan menguji performa model secara terpisah [22]. Sebanyak 4.000 gambar (80% dari total dataset) digunakan sebagai *training set* untuk melatih model dalam mengenali pola dan fitur penting pada gambar. Dari sisa data, 750 gambar (15% dari total dataset) dialokasikan sebagai *validation set* untuk mengevaluasi performa model selama pelatihan, sehingga risiko *overfitting* dapat diminimalkan. Sisanya, sebanyak 250 gambar (5% dari total dataset), digunakan sebagai *test set* untuk menguji kemampuan akhir model pada data yang belum pernah dilihat sebelumnya. Pendekatan ini memastikan setiap tahap pelatihan, validasi, dan pengujian dilakukan secara terpisah untuk memperoleh hasil yang optimal.



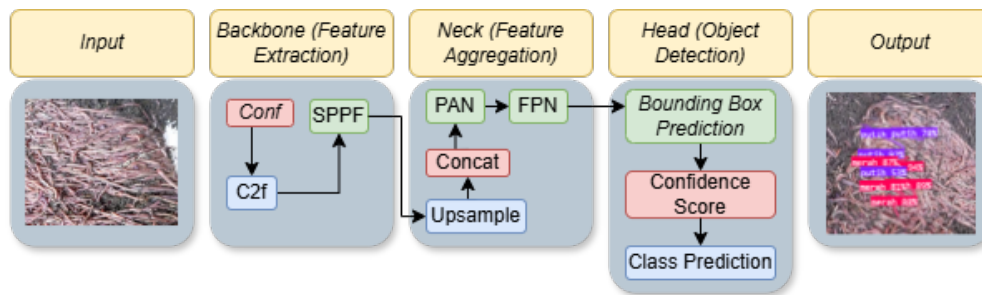
Gambar 3. Distribusi Pembagian Data

#### 2.5 Pembentukan Model

Arsitektur YOLOv8 dalam penelitian ini terdiri dari tiga komponen utama: *Backbone*, *Neck*, dan *Head*, yang saling terintegrasi untuk mendukung proses deteksi objek [23] yang disajikan pada Gambar 4. Pada tahap pertama, *Backbone (Feature Extraction)*, alur dimulai dengan *layer convolution* awal (*Conf*) yang bertugas mengekstrak fitur sederhana seperti tepi dan tekstur dari gambar [24]. Selanjutnya, modul *C2f (Cross Stage*

*Partial with Focused Connections*) digunakan untuk mengurangi redundansi komputasi dengan memanfaatkan koneksi lintas lapisan, yang membantu model mempelajari fitur yang lebih kompleks [24]. Tahap terakhir dari *backbone* adalah SPPF (*Spatial Pyramid Pooling-Fast*), yang menggabungkan informasi spasial dari berbagai skala secara efisien, sehingga model mampu mendeteksi objek dengan berbagai ukuran. Proses ini menghasilkan peta fitur (*feature map*) yang akan diteruskan ke tahap berikutnya [25].

Pada tahap kedua, *Neck (Feature Aggregation)*, peta fitur dari *backbone* diproses lebih lanjut untuk menghasilkan representasi *multi-skala* [26]. Alur dimulai dari SPPF, yang diteruskan ke *Upsample* untuk memperbesar peta fitur agar dapat digabungkan dengan fitur lain. Selanjutnya, proses *Concat (Concatenation)* menggabungkan peta fitur dari berbagai skala untuk menangkap informasi lebih kaya. PAN (*Path Aggregation Network*) digunakan untuk memperkuat koneksi antara lapisan rendah dan tinggi, menjaga informasi konteks dan detail kecil [27]. Akhirnya, FPN (*Feature Pyramid Network*) mengintegrasikan peta fitur dari berbagai level untuk memastikan deteksi objek, baik yang kecil maupun besar. Proses ini memastikan bahwa fitur relevan dari *backbone* dapat dimanfaatkan secara maksimal oleh *head*. Pada tahap terakhir, *Head (Object Detection)*,



Gambar 4. Arsitektur YoloV8 untuk segmentasi warna cacing ANC

*ad (Object Detection)*, *output* dari FPN diteruskan untuk menghasilkan prediksi akhir[28]. Proses dimulai dengan *Bounding Box Prediction*, di mana model memprediksi koordinat *bounding box* untuk setiap objek dalam gambar. Selanjutnya, *Confidence Score* dihitung untuk menentukan tingkat kepercayaan bahwa *bounding box* tersebut mengandung objek. Akhirnya, *Class Prediction* dilakukan untuk menentukan kelas dari objek yang terdeteksi, seperti cacing ANC dalam studi kasus ini. *Head* bertugas menghasilkan prediksi *bounding box*, *confidence score*, dan klasifikasi kelas yang akurat berdasarkan fitur yang telah diproses.

Tahap akhir dari model adalah *Output*, yang memberikan hasil berupa *bounding box* yang divisualisasikan pada gambar *input*, lengkap dengan *confidence score* dan label kelas objek. Dalam konteks penelitian ini, *output* mencakup deteksi dan klasifikasi cacing ANC. Proses ini dirancang sesuai dengan *pipeline* YOLOv8, memastikan model mampu mendeteksi objek secara akurat dan efisien pada berbagai kondisi dan ukuran. Penjelasan terperinci ini menggambarkan setiap tahap dalam arsitektur model, memberikan landasan teknis yang jelas untuk penelitian ini.

### 2.6 Pengujian Model dan Evaluasi

Setelah model selesai dilatih, proses pengujian dilakukan menggunakan dataset validasi untuk mengevaluasi kinerja deteksi. Dataset validasi digunakan untuk mengukur kemampuan model dalam mendeteksi objek yang relevan secara akurat [29]. Model diuji dengan berbagai metrik evaluasi untuk memastikan performanya sesuai dengan tujuan penelitian.

Proses evaluasi dan pengujian model melibatkan beberapa metrik utama. *Precision* digunakan untuk mengukur ketepatan model dalam mendeteksi objek yang relevan [29] dan dihitung dengan rumus (1) :

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{1}$$

Sementara itu, *Recall* mengukur sensitivitas model, yaitu kemampuannya untuk mendeteksi semua objek yang relevan [30], dengan rumus (2):

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \tag{2}$$

Selanjutnya, metrik *F1-Score* digunakan untuk memberikan rata-rata harmonik antara *precision* dan *recall* [22], memberikan keseimbangan antara keduanya. rumus (3) :

$$F1-Score = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \quad (3)$$

Terakhir, performa keseluruhan model dievaluasi menggunakan *Mean Average Precision* (mAP), yang merupakan rata-rata dari nilai *precision* pada berbagai *threshold Intersection over Union* (IoU) [31]. mAP memberikan gambaran lengkap mengenai keakuratan deteksi model untuk berbagai tingkat toleransi kesalahan [32]. Dengan menggunakan metrik ini, model dapat dinilai secara objektif untuk mengetahui kemampuan deteksinya dalam mengenali objek dengan akurat, serta memastikan performanya sesuai dengan kebutuhan studi kasus. Evaluasi ini memberikan landasan kuat dalam memastikan keandalan model pada data nyata.

### 3. Hasil

Hasil evaluasi menggunakan *confusion matrix* yang disajikan pada Gambar 5 menunjukkan performa model dalam mendeteksi tiga kelas utama, yaitu merah, putih, dan putih-biru, serta *false negatives*. Untuk kelas "merah," model berhasil memprediksi dengan benar sebanyak 40 objek (*true positives*), tetapi masih terdapat 10 objek yang tidak terdeteksi dengan benar (*false negatives*). Pada kelas "putih," model menunjukkan kinerja yang sangat baik dengan 97 objek terdeteksi dengan benar, meskipun terdapat 1 objek salah diprediksi sebagai "merah," 2 objek diprediksi sebagai "putih-biru," dan 38 objek tidak terdeteksi (*false negatives*). Kelas "putih-biru" juga menunjukkan performa yang cukup baik dengan 39 objek terdeteksi dengan benar, tetapi masih terdapat 10 *false negatives*.

Selain itu, model menghasilkan beberapa *false positives*, yaitu 2 prediksi salah untuk kelas "merah," 3 untuk kelas "putih," dan 1 untuk kelas "putih-biru." Secara keseluruhan, model menunjukkan performa yang baik, terutama pada kelas "putih," yang memiliki tingkat deteksi tertinggi. Namun, kelemahan masih terlihat pada sensitivitas deteksi untuk kelas "merah" dan "putih-biru," yang dapat ditingkatkan lebih lanjut untuk mengurangi kesalahan *false negatives*. Hasil ini memberikan gambaran jelas mengenai keandalan model dalam mendeteksi dan mengklasifikasikan objek, meskipun ada ruang untuk perbaikan pada beberapa kategori.

	merah	putih	putih-biru	False Negative
merah	40	0	0	10
putih	1	97	2	38
putih-biru	0	0	39	10
False Positive	2	3	1	0

Gambar 5. Hasil Confusion Matrix.

Hasil evaluasi Model YOLOv8 yang telah dilatih menggunakan dataset cacing ANC menunjukkan performa yang sangat baik pada Tabel 1. Rata-rata nilai *mean Average Precision* (mAP) pada *training set* adalah 93.8%, yang menunjukkan kemampuan model dalam mendeteksi objek secara konsisten pada berbagai nilai *Intersection over Union* (IoU) [32]. Selain itu, model menghasilkan nilai *precision* sebesar 95.6%, yang mengindikasikan tingkat keakuratan model dalam mengidentifikasi cacing yang relevan tanpa menghasilkan terlalu banyak prediksi yang salah. Sedangkan nilai *recall* mencapai 85.4%, menunjukkan sensitivitas model dalam mendeteksi sebagian besar objek yang ada dalam dataset. Berdasarkan nilai *precision*

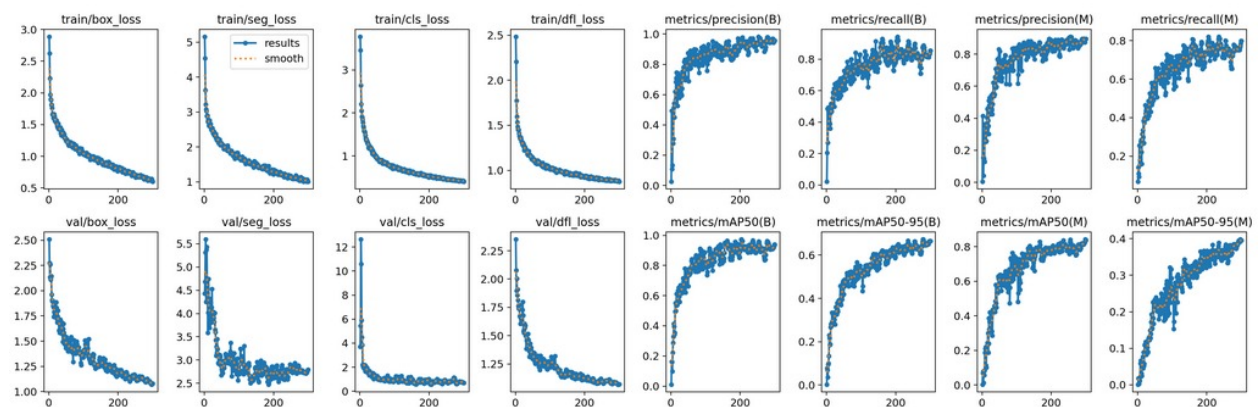
dan *recall*, model memiliki *F1-Score* sebesar 90.3%, yang merupakan rata-rata harmonik dari *precision* dan *recall*. Hal ini menunjukkan bahwa model YOLOv8 cukup andal untuk mendeteksi dan mengklasifikasikan cacing ANC dengan tingkat akurasi yang tinggi.

Tabel 1. Hasil Evaluasi

Metric	Training Set	Validation Set	Test Set
mAP (%)	93.8	94	95
Precision (%)	95.6	94.5	94.8
Recall (%)	85.4	84.5	85
F1-Score (%)	90.3	89.3	89.9

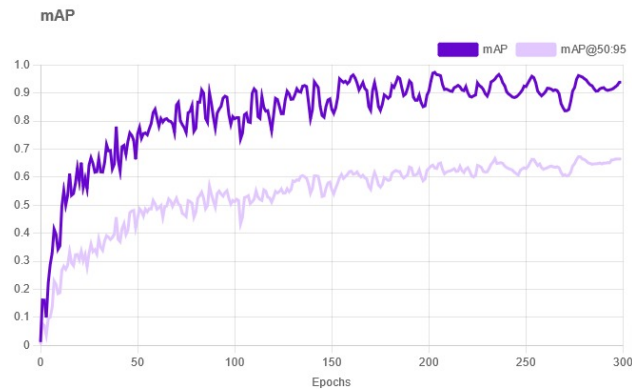
Evaluasi pada *validation set* menunjukkan rata-rata nilai mAP sebesar 94.0%, sedikit lebih tinggi dibandingkan *training set*. Model menghasilkan nilai *precision* sebesar 94.5%, *recall* sebesar 84.5%, dan *F1-Score* sebesar 89.3%. Hal ini menunjukkan bahwa model mampu mempertahankan performa deteksi yang sangat baik pada data validasi, dengan keseimbangan yang cukup antara *precision* dan *recall*. Pada *test set*, model menunjukkan performa yang serupa, dengan rata-rata nilai mAP mencapai 95.0%. *Precision* pada *test set* mencapai 94.8%, *recall* sebesar 85.0%, dan *F1-Score* sebesar 89.9%. Hasil ini mengindikasikan bahwa model mampu mempertahankan kinerjanya pada data yang belum pernah dilihat sebelumnya. Secara keseluruhan, model menunjukkan performa yang andal, dengan tingkat deteksi yang konsisten pada *training*, *validation*, dan *test set*. Penilaian ini menunjukkan bahwa model YOLOv8 dapat digunakan secara efektif untuk mendeteksi dan mengklasifikasikan cacing ANC dengan tingkat akurasi tinggi di berbagai skenario dataset. Namun, beberapa perbaikan pada *recall* dapat lebih meningkatkan sensitivitas model terhadap deteksi objek.

Hasil evaluasi performa model YOLOv8 berdasarkan pelatihan menunjukkan tren peningkatan yang konsisten, sebagaimana ditunjukkan dalam Gambar 6 hingga Gambar 8. Gambar 6 menunjukkan perubahan nilai *loss* (*train/box\_loss*, *train/seg\_loss*, *train/cls\_loss*, *train/dfl\_loss*) selama proses pelatihan, baik pada *training set* maupun *validation set*. Nilai *loss* untuk *box* (kesalahan prediksi *bounding box*), *seg* (*segmentation loss*), *cls* (*class loss*), dan *dfl* (*distribution focal loss*) mengalami penurunan yang signifikan selama pelatihan, yang mengindikasikan bahwa model semakin teroptimasi dan mampu memahami pola data dengan baik. Penurunan nilai pada *validation loss* juga menunjukkan bahwa model memiliki kemampuan generalisasi yang baik, dengan tidak menunjukkan tanda-tanda *overfitting*.



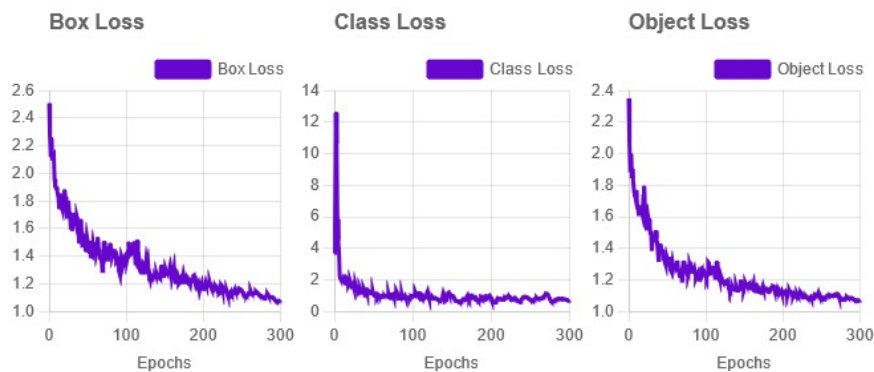
Gambar 6. Visualisasi Nilai Loss.

Gambar 7 memvisualisasikan nilai rata-rata *precision*, *recall*, dan *mean Average Precision* (mAP) selama pelatihan. Pada *threshold* IoU 50% (mAP50) dan berbagai *threshold* lainnya (mAP50-95), performa model terus meningkat seiring bertambahnya jumlah *epoch*, hingga mencapai stabilitas pada nilai mendekati maksimum setelah sekitar 200 *epoch*. Hal ini mengindikasikan kemampuan model untuk mendeteksi objek dengan tingkat akurasi tinggi, baik untuk objek kecil maupun besar, pada berbagai tingkat toleransi kesalahan.



Gambar 7. Visualisasi fase pelatihan.

Gambar 8 menggambarkan tren penurunan *box loss*, *class loss*, dan *object loss* hingga mencapai nilai yang sangat kecil setelah sejumlah *epoch*. *Box loss* menurun hingga mendekati 1.0, menunjukkan prediksi *bounding box* yang semakin akurat. *Class loss* yang awalnya berada di angka tinggi (>14) menurun drastis ke angka stabil, mengindikasikan kemampuan model untuk mengenali kelas objek dengan baik. *Object loss* juga menunjukkan tren yang sama, di mana model semakin presisi dalam memprediksi keberadaan objek pada gambar.



Gambar 8. Visualisasi Tren Penurunan Loss.

Secara keseluruhan, grafik-grafik ini menunjukkan bahwa model YOLOv8 telah dilatih secara efektif dan memiliki performa generalisasi yang sangat baik. Penurunan nilai *loss* dan peningkatan nilai mAP menegaskan bahwa model mampu mendeteksi dan mengklasifikasikan objek dengan tingkat akurasi yang tinggi, sehingga dapat digunakan secara andal untuk mendeteksi cacing ANC pada dataset yang digunakan.

#### 4. Pembahasan

Model YOLOv8 yang digunakan dalam penelitian ini menunjukkan performa yang sangat baik dalam mendeteksi dan mengklasifikasikan cacing ANC berdasarkan evaluasi dataset. Rata-rata nilai mAP (*mean Average Precision*) mencapai 93.8% pada *training set*, 94.0% pada *validation set*, dan 95.0% pada *test set*, mengindikasikan kemampuan model yang konsisten dalam mendeteksi objek pada berbagai tingkat *Intersection over Union* (IoU). Selain itu, nilai *precision* yang tinggi menunjukkan akurasi model dalam mengenali objek tanpa menghasilkan prediksi yang salah secara berlebihan, dengan *precision* mencapai 95.6% pada *training set*, 94.5% pada *validation set*, dan 94.8% pada *test set*. Di sisi lain, *recall* menunjukkan sensitivitas model terhadap pendeteksian semua objek, dengan nilai 85.4% pada *training set*, 84.5% pada *validation set*, dan 85.0% pada *test set*. Rata-rata harmonik dari *precision* dan *recall*, yaitu *F1-Score*, menunjukkan nilai yang baik, dengan 90.3%, 89.3%, dan 89.9% masing-masing untuk *training*, *validation*, dan *test set*, yang menegaskan keseimbangan performa model dalam deteksi objek.



Penelitian ini dibandingkan dengan penelitian terdahulu, seperti [8] dan [9], yang juga menggunakan algoritma YOLO untuk aplikasi agrikultur. Hasil penelitian ini menunjukkan peningkatan performa pada deteksi spesifik cacing ANC dibandingkan dengan aplikasi serupa pada deteksi hama dan penyakit tanaman. Namun, ada beberapa kelemahan yang diidentifikasi, seperti sensitivitas model yang sedikit lebih rendah pada kelas tertentu (misalnya, kelas merah pada *confusion matrix*). Hal ini dapat dikaitkan dengan distribusi dataset yang tidak seimbang atau karakteristik visual objek yang lebih sulit dideteksi. Dalam penelitian sebelumnya oleh [10], penggunaan teknik fusi fitur pada YOLO menghasilkan peningkatan deteksi pada hama kecil. Teknik serupa dapat diterapkan untuk meningkatkan sensitivitas model ini terhadap kelas tertentu yang menunjukkan *false negatives* lebih tinggi.

Penelitian ini memberikan kontribusi signifikan dalam mendukung inovasi peternakan berbasis teknologi. Dengan mengintegrasikan YOLOv8 untuk deteksi otomatis cacing ANC, penelitian ini tidak hanya memperkenalkan pendekatan yang lebih efisien dibandingkan metode manual tetapi juga menciptakan solusi yang dapat diadopsi oleh peternak modern. Selain itu, implementasi segmentasi berbasis warna membuka peluang pengembangan aplikasi serupa untuk klasifikasi objek dalam agrikultur dan peternakan lainnya. Penelitian ini juga memperluas cakupan penerapan algoritma YOLO dalam deteksi spesifik berbasis visual, memberikan landasan kuat untuk penelitian dan pengembangan lebih lanjut di bidang ini. Sehingga penelitian ini berhasil membuktikan keandalan YOLOv8 sebagai metode untuk mendeteksi dan mengklasifikasikan cacing ANC berdasarkan segmentasi warna, memberikan solusi otomatis yang efisien dan akurat bagi peternak cacing. Namun, masih ada ruang untuk perbaikan, seperti memperluas dataset dengan variasi lebih besar atau menerapkan teknik peningkatan fitur untuk mengurangi *false negatives*. Penelitian di masa depan dapat fokus pada pengembangan model yang lebih *robust* terhadap variasi dataset dan meningkatkan generalisasi pada kondisi dunia nyata, sehingga menghasilkan sistem yang lebih tangguh untuk aplikasi agrikultur otomatis.

## 5. Simpulan

Penelitian ini mengungkapkan bahwa penerapan model YOLOv8 yang didukung oleh segmentasi berbasis warna mampu mendeteksi dan mengklasifikasikan cacing *African Night Crawler* (ANC) dengan tingkat akurasi tinggi. Penelitian ini penting karena memperkenalkan solusi otomatis yang dapat meningkatkan efisiensi dan akurasi dalam pengelompokan kualitas cacing ANC, menggantikan metode manual yang subjektif dan kurang efisien.

Hasil penelitian menunjukkan rata-rata nilai *mean Average Precision* (mAP) sebesar 93.8% pada *training set*, 94.0% pada *validation set*, dan 95.0% pada *test set*. *Precision model* mencapai 95.6% pada *training set*, sementara *recall* masing-masing tercatat 85.4%, 84.5%, dan 85.0% untuk dataset yang digunakan, dengan *F1-Score* mencapai 90.3% pada *training set*. Keseimbangan antara *precision* dan *recall* ini menegaskan keandalan model dalam mendeteksi objek yang relevan sekaligus sensitif terhadap semua objek dalam dataset.

Studi ini memberikan kontribusi penting bagi inovasi di sektor peternakan berbasis teknologi, khususnya dalam klasifikasi objek agrikultur. Pendekatan ini dapat diterapkan pada skenario lain seperti deteksi hama atau penyakit tanaman, memberikan solusi teknologi yang relevan bagi pelaku agribisnis. Dengan memperluas cakupan dataset atau menerapkan teknik peningkatan fitur, hasil yang lebih baik dapat dicapai. Penelitian di masa depan dapat fokus pada generalisasi model dalam menghadapi variasi kondisi lapangan nyata, sehingga menghasilkan sistem yang lebih tangguh dan aplikatif.

## Sumber dana

Penelitian ini didanai melalui Program Penelitian Dosen Dana Internal Perintis (PERINTIS) Universitas Widyagama Malang. Dana yang digunakan untuk mendukung kegiatan yang mencakup berbagai komponen biaya seperti pengumpulan data, analisis, serta publikasi hasil penelitian. Penelitian ini sepenuhnya didanai oleh sumber dana internal yang disetujui pada tahun 2024.

## Ucapan Terima kasih

Penulis mengucapkan terima kasih kepada CV. Messi Putra Jaya atas dukungan data dan bantuan praktis selama pengumpulan data, serta kepada rekan-rekan peneliti. Ucapan terima kasih juga ditujukan kepada LPPM Universitas Widyagama Malang atas dukungan administrasi dan pendampingan selama proses penelitian. Penelitian ini didanai oleh Program Penelitian Dosen Dana Internal Perintis (PERINTIS) Universitas Widyagama Malang, yang memungkinkan penelitian ini terlaksana dengan baik.

## Pustaka

- [1] N. Baylon and I. Catian, "Population growth rate of african night (eudrilus eugeniae) crawler fed with different leguminous leaves as supplement," *Philippine Journal of Agricultural Economics*, vol. 3, no. 1, pp. 52–64, Jan. 2019.
- [2] F. Kabi, D. Kayima, A. Kigozi, E. Z. Mpingirika, R. Kayiwa, and D. Okello, "Effect of different organic substrates on reproductive biology, growth rate and offtake of the african night crawler earthworm (eudrilus eugeniae)," *Organic Agriculture*, vol. 10, no. 3, pp. 395–407, Sep. 2020.
- [3] I. R. Dini, R. Saputra, D. Salbiah, and S. Yoseva, "Development of integrated organic agricultural agribusiness at d'jamuran pekanbaru student business," *ABDIMAS Universitas Muhammadiyah Tasikmalaya*, vol. 4, Jan. 2022.
- [4] T. M. T. LE *et al.*, "Using cassava waste of the cassava starch processing as food for raising african nightcrawler (eudrilus eugeniae) to obtain vermicomposting and earthworm biomass," *Journal of Vietnamese Environment*, vol. 12, no. 2, pp. 169–176, Nov. 2020.
- [5] N. Stasenko, E. Chernova, D. G. Shadrin, G. V. Ovchinnikov, I. Krivolapov, and M. Pukalchik, "Deep learning for improving the storage process: Accurate and automatic segmentation of spoiled areas on apples," in *2021 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, 2021, pp. 1–6. [Online]. Available: <https://api.semanticscholar.org/CorpusID:235748556>
- [6] E. J. Sadgrove, G. Falzon, D. Miron, and D. W. Lamb, "The segmented colour feature extreme learning machine: Applications in agricultural robotics," *Agronomy*, vol. 11, no. 11, Nov. 2021.
- [7] A. F. Leite *et al.*, "Artificial intelligence-driven novel tool for tooth detection and segmentation on panoramic radiographs," *Clin Oral Investig*, vol. 25, no. 4, pp. 2257–2267, Apr. 2021.
- [8] Y. Luo *et al.*, "Detection of agricultural pests based on yolo," in *Journal of Physics: Conference Series*. Institute of Physics, 2023.
- [9] I. Ahmad *et al.*, "Deep learning based detector yolov5 for identifying insect pests," *Applied Sciences (Switzerland)*, vol. 12, no. 19, Oct. 2022.
- [10] S. Dong, J. Zhang, F. Wang, and X. Wang, "Yolo-pest: a real-time multi-class crop pest detection model," in *International Conference on Computer Application and Information Security*. SPIE-Intl Soc Optical Eng, May 2022, p. 41.
- [11] A. Morbekar, A. Parihar, and R. Jadhav, "Crop disease detection using yolo," Jun. 2020.
- [12] C. Wen *et al.*, "Pest-yolo: A model for large-scale multi-class dense and tiny pest detection and counting," *Front Plant Sci*, vol. 13, Oct. 2022.
- [13] A. Marin and I. E. Radoi, "Image-based fruit recognition and classification," in *2022 21st RoEduNet Conference: Networking in Education and Research (RoEduNet)*, 2022, pp. 1–4. [Online]. Available: <https://api.semanticscholar.org/CorpusID:253047423>
- [14] L. K. R. Chairma, B. Praveena, G. Sahaana, J. J. Nithya, T. Gnanasekaran, and M. Hashim, "Yolo for detecting plant diseases," in *2023 Third International Conference on Artificial Intelligence and Smart Energy (ICAIS)*, 2023, pp. 1029–1034. [Online]. Available: <https://api.semanticscholar.org/CorpusID:257792406>

- [15] R. Suguna, V. K. C. N. S, S. Deepa, and A. M. S., "Apple and tomato leaves disease detection using emperor penguins optimizer based cnn," in *2023 9th International Conference on Advanced Computing and Communication Systems (ICACCS)*, vol. 1, 2023, pp. 1803–1808. [Online]. Available: <https://api.semanticscholar.org/CorpusID:258510809>
- [16] D. Y. K. Ramalingam, P. Ramani, and R. M. Deepak, "Machine learning in the detection of oral lesions with clinical intraoral images," *Cureus*, Aug. 2023.
- [17] P. Kakani and S. Vyas, "Automated catalog generation using deep learning," *International Research Journal of Modernization in Engineering Technology and Science*, Aug. 2023.
- [18] A. S. Nurhikam, W. P. Agung, S. Rohman, and I. M. Saputra, "Klasifikasi tanaman obat berdasarkan citra daun menggunakan backpropagation neural networks," *JIKO (Jurnal Informatika dan Komputer)*, vol. 8, no. 1, p. 1, Feb. 2024.
- [19] P. Sengupta, A. Mehta, and P. S. Rana, "Enhancing performance of deep learning models with a novel data augmentation approach," in *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, 2023, pp. 1–7. [Online]. Available: <https://api.semanticscholar.org/CorpusID:265407195>
- [20] M. K. Nanthini, D. D. Sivabalaselvamani, M. K. Chitra, M. P. Gokul, M. S. KavinKumar, and M. S. Kishore, "A survey on data augmentation techniques," in *2023 7th International Conference on Computing Methodologies and Communication (ICCMC)*, 2023, pp. 913–920. [Online]. Available: <https://api.semanticscholar.org/CorpusID:257959880>
- [21] Y. Wang, G. Huang, S. Song, X. Pan, Y. Xia, and C. Wu, "Regularizing deep networks with semantic data augmentation," *IEEE Trans Pattern Anal Mach Intell*, vol. 44, pp. 3733–3748, 2020. [Online]. Available: <https://api.semanticscholar.org/CorpusID:220665675>
- [22] I. Akbar, M. Faisal, and T. Chamidy, "Multi-label classification of indonesian qur'an translation using long short-term memory model," *Computer Network, Computing, Electronics, and Control Journal*, vol. 4, no. 3, pp. 119–128, 2019. [Online]. Available: <https://kinetik.umm.ac.id/index.php/kinetik/article/view/1901>
- [23] J. Zhang, J. Zhang, K. Zhou, Y. Zhang, H. Chen, and X. Yan, "An improved yolov5-based underwater object-detection framework," in *Sensors (Basel)*, vol. 23, 2023. [Online]. Available: <https://api.semanticscholar.org/CorpusID:257943960>
- [24] L. Shen, B. Lang, and Z. Song, "Ds-yolov8-based object detection method for remote sensing images," *IEEE Access*, vol. 11, pp. 125 122–125 137, 2023. [Online]. Available: <https://api.semanticscholar.org/CorpusID:265091454>
- [25] M. Ma and H. Pang, "Sp-yolov8s: An improved yolov8s model for remote sensing image tiny object detection," *Applied Sciences (Switzerland)*, vol. 13, no. 14, Jul. 2023.
- [26] K. Liang, W. Zhang, F. Li, Z. Jiang, and D. Zhang, "Improved yolov7 for small and overlapping objects detection," in *2023 IEEE 6th International Conference on Pattern Recognition and Artificial Intelligence (PRAI)*, 2023, pp. 153–157. [Online]. Available: <https://api.semanticscholar.org/CorpusID:265877090>
- [27] Z. Tian, J. Huang, Y. Yang, and W. Nie, "Kcfs-yolov5: A high-precision detection method for object detection in aerial remote sensing images," *Applied Sciences (Switzerland)*, vol. 13, no. 1, Jan. 2023.
- [28] J. Liang, R. Kong, R. Ma, J. Zhang, and X. Bian, "Aluminum surface defect detection algorithm based on improved yolov5," in *Adv Theory Simul*, vol. 7, 2023. [Online]. Available: <https://api.semanticscholar.org/CorpusID:266294408>
- [29] F. Nurdiansyah *et al.*, "Penerapan convolutional neural network untuk deteksi kualitas telur ayam ras berdasarkan warna cangkang," 2024.

- [30] A. R. Muslikh, I. Akbar, D. R. I. M. Setiadi, and H. M. M. Islam, “Multi-label classification of indonesian al-quran translation based cnn, bilstm, and fasttext,” 2023. [Online]. Available: <https://quran.kemenag.go.id>
- [31] M. Otani, R. Togashi, Y. Nakashima, E. Rahtu, J. Heikkila, and S. Satoh, “Optimal correction cost for object detection evaluation,” in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 21 075–21 083. [Online]. Available: <https://api.semanticscholar.org/CorpusID:247763091>
- [32] B. Cheng, R. B. Girshick, P. Doll’ar, A. C. Berg, and A. Kirillov, “Boundary iou: Improving object-centric image segmentation evaluation,” in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 15 329–15 337. [Online]. Available: <https://api.semanticscholar.org/CorpusID:232417671>