



## Perbandingan Kinerja YOLO vs Faster R-CNN untuk Deteksi & Estimasi Berat Ikan

Justam<sup>1</sup>, Abdul Malik<sup>2</sup>, Erlita<sup>3</sup>, Deo Mangellak<sup>4</sup>, Yuyun<sup>5</sup>

Program Studi Informatika, Universitas Mega Buana Palopo<sup>1</sup>, Program Studi Sistem Informasi, Universitas Mega Buana Palopo<sup>2</sup>, Program Studi Sistem Informasi, Institut Teknologi Pendidikan dan Bisnis (ITPB) Qana'ah Sidenreng Rappang<sup>3</sup>, Program Studi Informatika, Universitas Mega Buana Palopo<sup>4</sup>, Program Studi Informatika, Universitas Mega Buana Palopo<sup>5</sup>

Jl. Andi Ahmad No. 25, Wara Utara, 91913, Kota Palopo, Sulawesi Selatan, Indonesia<sup>1245</sup>, Jl.

Planduk No. 6, Maritenggar, 91611, Sidenreng Rappang<sup>3</sup>

justam@umegabuana.ac.id\*<sup>1</sup>, malik@umegabuana.ac.id<sup>2</sup>, erlitazainddin3@gmail.com<sup>3</sup>,

deoganz223@gmail.com<sup>4</sup>, yuyunmairi515@gmail.com<sup>5</sup>

### Kata Kunci :

Kerapu;  
Kakap;  
YOLO;  
Faster R-CNN;  
Deteksi Jenis;  
Estimasi Berat.

### ABSTRAK

Ikan kerapu dan ikan kakap memiliki nilai ekonomi tinggi di pasar global, sehingga identifikasi jenis dan estimasi beratnya menjadi aspek penting dalam perdagangan. Metode manual yang umum digunakan memerlukan waktu lama dan tenaga kerja besar. Oleh karena itu, penelitian ini membandingkan performa dua model deep learning, yaitu YOLO dan Faster R-CNN, dalam mendeteksi jenis dan mengestimasi berat ikan. Dataset terdiri dari 2.991 citra yang terbagi dalam 18 kelas dan diperluas melalui augmentasi menjadi 6.843 citra. Proses deteksi menggunakan detection threshold 0,8, dengan evaluasi berdasarkan precision, recall, accuracy, serta Mean Absolute Percentage Error (MAPE) untuk estimasi berat. Hasil menunjukkan bahwa model YOLO memiliki precision, recall, dan accuracy masing-masing sebesar 0,98, 0,98, dan 0,96, sedangkan Faster R-CNN mencapai 0,97, 0,98, dan 0,95. Untuk estimasi berat, MAPE YOLO pada citra sebesar 2,42% dan pada video 3,66%, sementara Faster R-CNN memiliki MAPE 14,62% pada citra dan 13,59% pada video. Dengan demikian, model YOLO menunjukkan kinerja lebih baik dibandingkan Faster R-CNN dalam mendeteksi jenis dan mengestimasi berat ikan.

### Keywords

Grouper;  
Snapper; YOLO;  
Faster R-CNN;  
Species  
Detection;  
Weight  
Estimation;

### ABSTRACT

*Grouper and snapper fish have high economic value in the global market, making species identification and weight estimation crucial aspects of trade. Traditional manual methods are time-consuming and labor-intensive. Therefore, this study compares the performance of two deep learning models, YOLO and Faster R-CNN, in detecting fish species and estimating their weight.*

*The dataset consists of 2,991 images divided into 18 classes, augmented to 6,843 images using rotation and flipping techniques. The detection process employs a threshold of 0.8, with performance evaluated using precision, recall, and accuracy, while Mean Absolute Percentage Error (MAPE) is used for weight estimation. Results show that YOLO achieved precision, recall, and accuracy of 0.98, 0.98, and 0.96, respectively, while Faster R-CNN achieved 0.97, 0.98, and 0.95. For weight estimation, YOLO's MAPE was 2.42% for images and 3.66% for videos, whereas Faster R-CNN had a MAPE of 14.62% for images and 13.59% for videos. Thus, YOLO outperforms Faster R-CNN in both species detection and weight estimation.*

---Jurnal JISTI @2024---



---

## PENDAHULUAN

Ikan merupakan sumber protein yang penting bagi manusia dan menjadi salah satu komoditas utama dalam industri pangan global. Diperkirakan lebih dari 30% konsumsi ikan untuk manusia berasal dari sektor perikanan dan budidaya (Bashir dkk., 2019). Dalam tiga dekade terakhir, budidaya ikan telah berkembang pesat menjadi sektor penghasil ikan terbesar di Indonesia. Sebagian besar produksi ikan berasal dari usaha skala kecil di negara-negara berkembang, yang menghadapi tantangan dalam meningkatkan efisiensi dan kualitas hasil tangkapan mereka (Bashir dkk., 2019).

Di antara berbagai jenis ikan yang dikonsumsi secara luas, ikan kerapu dan ikan kakap memiliki nilai ekonomi tinggi di pasar global. Identifikasi jenis dan estimasi berat ikan menjadi aspek penting dalam industri perikanan, karena berpengaruh terhadap penentuan harga dan kualitas ikan yang diperdagangkan. Secara konvensional, proses ini dilakukan secara manual oleh para ahli. Namun, metode ini memiliki beberapa kelemahan, seperti membutuhkan waktu lama serta memerlukan tenaga kerja yang banyak.

Metode berbasis komputer, khususnya pengolahan citra digital, dapat menjadi solusi yang lebih efektif dan efisien dalam proses identifikasi jenis dan estimasi berat ikan. Pendekatan ini menawarkan kecepatan dan akurasi tinggi dalam pendeteksian objek. Penelitian sebelumnya oleh Fernandes dkk. (2020) telah mengaplikasikan metode pengolahan citra pada ikan nila, dengan evaluasi menggunakan metrik Intersection over Union (IoU). Hasil penelitian menunjukkan nilai IoU sebesar 99% untuk latar belakang, 90% untuk area badan ikan, dan 64% untuk sirip ikan (Fernandes dkk., 2020). Temuan ini menunjukkan bahwa metode berbasis pengolahan citra memiliki potensi besar dalam industri perikanan.

Berdasarkan urgensi peningkatan efisiensi dalam industri perikanan, penelitian ini bertujuan untuk membandingkan kinerja dua algoritma deep learning yang sering digunakan dalam deteksi objek, yaitu You Only Look Once (YOLO) dan Faster Region-Based Convolutional Neural Network (Faster R-CNN). Model ini akan diterapkan pada deteksi jenis serta estimasi berat ikan kerapu dan ikan kakap. Hasil dari penelitian ini diharapkan dapat memberikan wawasan baru dalam penerapan kecerdasan buatan di industri perikanan, serta berkontribusi pada peningkatan produktivitas dan keberlanjutan sektor ini.

## KAJIAN PUSTAKA

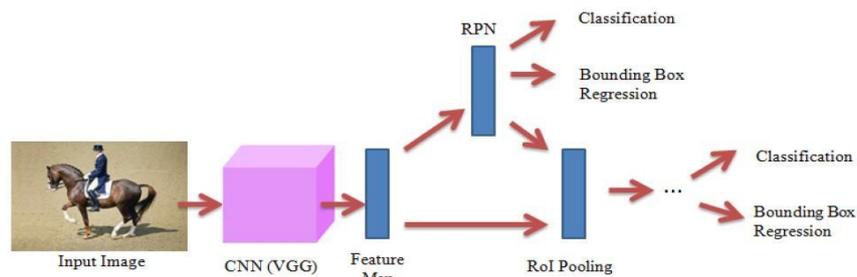
### 1. Ikan Kerapu

Ikan kerapu (*grouper*) termasuk dalam famili Serranidae, yaitu kelompok ikan laut karnivora yang tersebar di perairan tropis dan subtropis di seluruh dunia (Craig & Hastings, 2007). Famili ini terdiri dari 203 spesies yang ditemukan di berbagai wilayah (Chatla dkk., 2019). Ikan kerapu merupakan predator yang memangsa ikan kecil, plankton hewani, udang-udangan, dan invertebrata laut lainnya. Habitatnya meliputi terumbu karang, daerah berbatu, serta padang lamun (Darwin dkk., 2020). Secara morfologi, ikan kerapu memiliki tubuh kokoh dengan kepala besar, mulut lebar, dan sisik kecil yang rapat. Ciri khas lainnya adalah duri tajam pada operculum dan sirip punggung pertama (Ningsih, 2015). Ukurannya bervariasi, mulai dari 30 cm hingga mencapai 3 meter. Perbedaan antarspesies sering kali sulit diidentifikasi karena kemiripan bentuk dan pola tubuhnya, sehingga masyarakat sering kali menggunakan nama yang sama untuk beberapa spesies berbeda (Ding dkk., 2006).



### 1. *Faster Region-proposed Convolutional Neural Network (Faster R- CNN)*

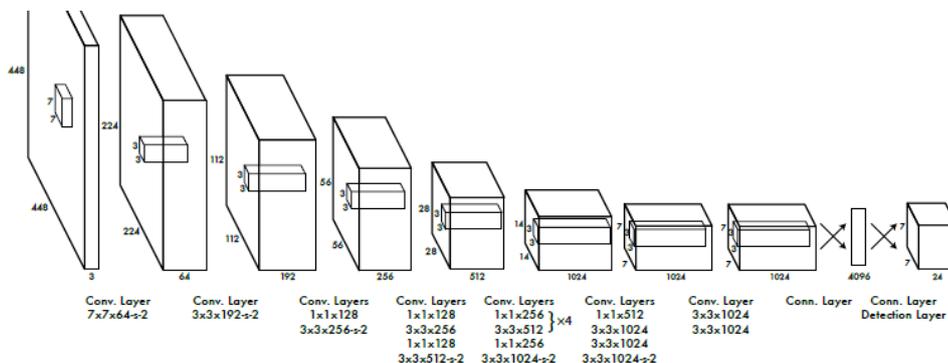
Faster R-CNN merupakan pengembangan dari R-CNN yang diperkenalkan pada tahun 2015. Algoritma ini menggabungkan Fast R-CNN dengan Region Proposal Network (RPN) dalam arsitekturnya. Perbedaannya terletak pada penggantian fitur tertentu di Fast R-CNN untuk meningkatkan efisiensi dan akurasi deteksi objek (Shianto dkk., 2019). Salah satu komponen utama dalam Faster R-CNN adalah Region Proposal Network (RPN), yang berfungsi mengidentifikasi lokasi objek dalam citra. RPN menerima gambar sebagai input dan menghasilkan kotak prediksi yang menunjukkan kemungkinan lokasi objek (Ren dkk., 2017). Prosesnya dimulai dengan memasukkan gambar ke dalam Convolutional Neural Network (CNN), seperti VGG atau ResNet, untuk menghasilkan peta fitur. Kemudian, RPN menggunakan sliding window untuk menganalisis berbagai posisi dalam peta fitur tersebut. Pada setiap posisi, RPN mengevaluasi beberapa kotak jangkar (anchor boxes) dengan skala dan rasio aspek yang berbeda guna mendeteksi berbagai ukuran dan bentuk objek dalam citra. Untuk arsitektur Faster R-CNN ditampilkan pada gambar 1.



Gambar 1. Srsitektur Faster R-CNN

### 2. *You Only Look Once (YOLO)*

You Only Look Once (YOLO) adalah algoritma yang banyak digunakan untuk object detection. Algoritma ini pertama kali diperkenalkan oleh Redmon dkk. (2015) dan dikenal karena ukuran modelnya yang kecil serta kecepatan kalkulasinya yang tinggi. Struktur YOLO memungkinkan prediksi kelas dan posisi objek dengan bounding box langsung melalui jaringan saraf tanpa memerlukan tahap pemrosesan terpisah. Keunggulan utama YOLO adalah kecepatan deteksi yang tinggi, karena cukup dengan satu kali pemrosesan gambar untuk menghasilkan hasil deteksi akhir. Hal ini membuatnya cocok untuk deteksi objek dalam video secara real-time. Selain itu, YOLO menggunakan pendekatan global image detection, yang membantu menyandikan informasi global dan mengurangi kesalahan dalam mendeteksi latar belakang sebagai objek. Dengan kemampuan generalisasi yang kuat, YOLO dapat belajar fitur umum dan diterapkan dalam berbagai bidang (Jiang dkk., 2022). Untuk arsitektur dari YOLO ditunjukkan oleh gambar 2.



Gambar 2. Arsitektur YOLO



## METODE PENELITIAN

Selanjutnya ini adalah bagian dari metode penelitian yang digunakan pada penelitian yang dilakukan.

### 1. Instrument penelitian

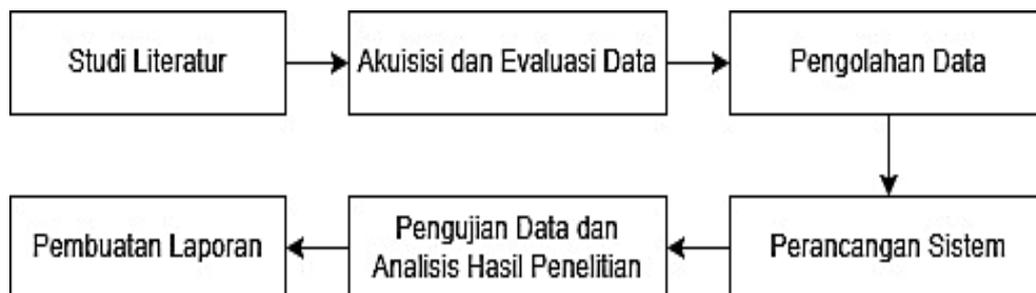
Pada penelitian ini digunakan dua jenis instrumen yaitu instrumen perangkat lunak dan instrumen perangkat keras. Berikut rincian mengenai setiap instrumen penelitian yang digunakan yang ditunjukkan pada table 1.

Tabel 1. Instrument penelitian

| Kategori        | Instrument  |
|-----------------|---|
| Perangkat Lunak | PyCharm<br>Google Colaboratory<br>Kaggle Notebook<br>Jupyter Notebook<br>Roboflow<br>Microsoft Office (Word, Excel, dan PowerPoint)<br>Windows 10 22H2<br>Python 3.10<br>Beberapa library pendukung seperti OpenCV dan PyTorch  |
| Perangkat Keras | Laptop Acer Aspire 5 A514-51G-52C3 Intel® Core™ i5-8265U 1.6 GHz NVIDIA® GeForce® MX130, RAM 20 GB serta menggunakan penyimpanan SSD 512 GB dan HDD 1 TB.<br>Komputer Asus ROG GT51CH Intel® Core™ i7-7700K NVIDIA® GeForce® GTX 1080, RAM 16 GB serta menggunakan penyimpanan SSD 512 GB dan HDD 1 TB<br>Xiaomi Redmi Note 11 Pro 5G<br>Xiaomi Redmi Note 5A<br>Tripod dan timbangan sebagai alat penunjang penelitian |

### 2. Tahapan Penelitian

Tahapan penelitian yang dilaksanakan secara garis besarnya dapat dilihat pada 3.



**Gambar 3. Tahapan Penelitian**

Berdasarkan Gambar 10, dapat dilihat bahwa penelitian ini memiliki enam tahap, dimana tahapan penelitian yang dilaksanakan adalah sebagai berikut.

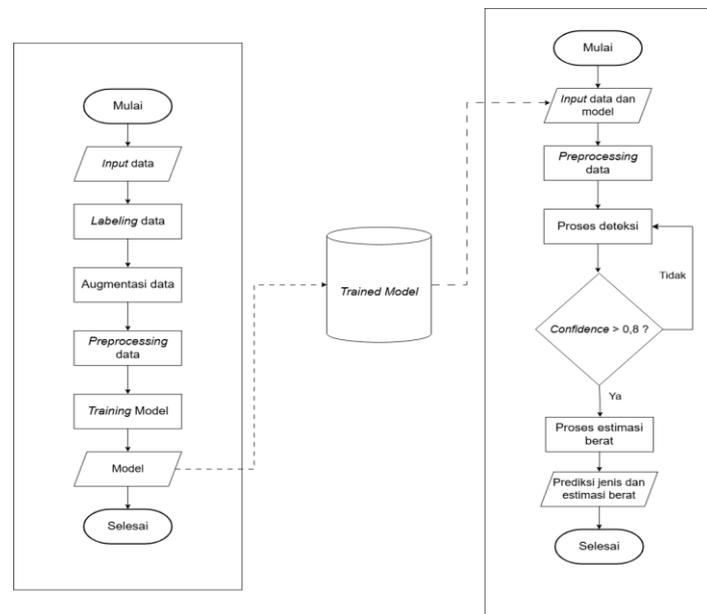
- Studi Literatur, Tahapan studi literatur melibatkan pencarian sumber-sumber literatur mengenai teori-teori atau aplikasi lain yang relevan dalam bidang visi komputer dan pembelajaran mesin, dengan penekanan khusus pada aplikasi yang terkait dengan data perikanan.



- Akuisisi dan Evaluasi Data, Proses ini mengandalkan dua jenis data, yaitu data primer yang dikumpulkan langsung oleh peneliti dan data sekunder. Evaluasi kualitas data dilakukan dengan memeriksa relevansinya terhadap tujuan penelitian serta memahami keterbatasan data yang ada.
- Pengolahan Data, Melakukan proses *labeling* atau anotasi data, terutama jika data yang diperoleh belum sepenuhnya terlabel. Selanjutnya, dilakukan tahap *preprocessing* yang meliputi pengubahan ukuran data dan prosedur pemrosesan lainnya untuk menstandarisasi data sesuai dengan persyaratan *input* model.
- Perancangan Sistem, Data yang telah dipreproses dibagi menjadi set pelatihan, pengujian, dan validasi. Data ini kemudian digunakan untuk melatih model. Model yang telah terlatih digunakan untuk mendeteksi kelas objek menggunakan data sekunder yang telah diproses.
- Pengujian Data dan Analisis Hasil Penelitian, Integrasi metode estimasi berat dengan model deteksi. Proses deteksi dan estimasi berat dilakukan berdasarkan karakteristik objek yang terdeteksi, yang membutuhkan analisis terpisah. Proses estimasi berat ini menggunakan data primer yang dikumpulkan oleh peneliti sebagai bentuk pengujian model terhadap data baru. Evaluasi dilakukan menggunakan *confusion matrix* dan MAPE.
- Pembuatan Laporan, Penyusunan laporan hasil penelitian yang menyoroti pentingnya kualitas dan pengolahan data sekunder dalam penelitian visi komputer dan pembelajaran mesin.

### 3. Perancangan Sistem

Rancangan sistem ini bertujuan untuk memberikan gambaran terhadap proses sistem yang akan dibuat dan dikembangkan serta untuk memperjelas detail dari alur proses pada sistem tersebut. Diagram rancangan sistem dapat dilihat pada Gambar 4



Gsmbsr 4. Rancangan sistem

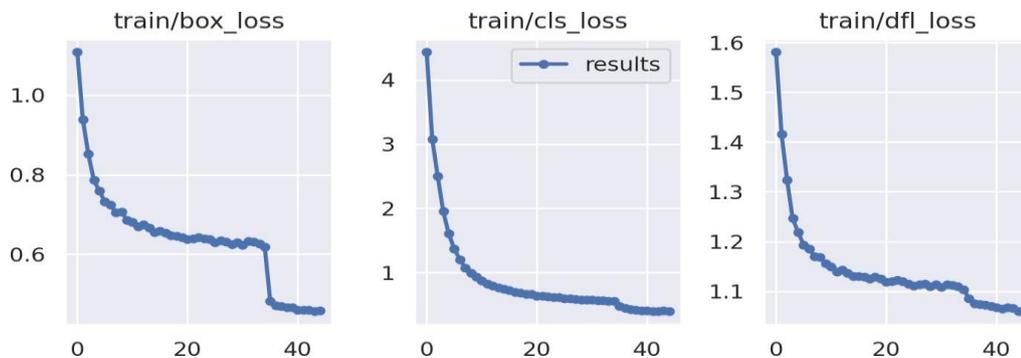
## HASIL DAN PEMBAHASAN

Bagian ini membahas hasil training dan testing model menggunakan algoritma YOLO dan Faster R-CNN. Analisis dilakukan untuk menilai kinerja masing-masing model dalam mendeteksi dan mengenali objek berdasarkan metrik seperti precision, recall dan accuracy. Hasil dari kedua algoritma ini akan dibahas secara terpisah, dimulai dengan analisis pada model YOLO kemudian diikuti oleh model Faster R-CNN.



## 1. Analisis Hasil Training dan Testing Model YOLO

Dari hasil *training* yang dilakukan, peneliti berfokus menganalisis tren nilai *loss* yang dihasilkan selama proses *train* berjalan dari awal hingga selesai. Selain itu, proses *train* dapat pula dianalisis dari grafik *loss* yang dihasilkan selama proses *train* berlangsung. Adapun grafik tersebut dapat dilihat pada Gambar 20.



**Gambar 5. Grafik *loss* pada YOLO**

Berdasarkan grafik nilai *loss* pada Gambar 20, terlihat adanya penurunan nilai pada model YOLO selama proses *training* dijalankan. Tren penurunan ini menunjukkan bahwa proses *training* pada model YOLO berjalan dengan baik dan efektif. *Box loss* turun dari 1,10 ke sekitar 0,4556, sementara *classification loss* turun dari 4,436 ke sekitar 0,3922. Nilai *box loss* mengalami penurunan signifikan mulai dari *epoch* 35, sedangkan *classification loss* menunjukkan penurunan signifikan mulai dari *epoch* 36. Dengan tren penurunan tersebut maka proses *training* pada model YOLO disimpulkan telah berjalan dengan baik. Adapun deteksi pada model YOLO menggunakan data uji yang telah dikelompokkan sebelumnya ditunjukkan pada Gambar 6.

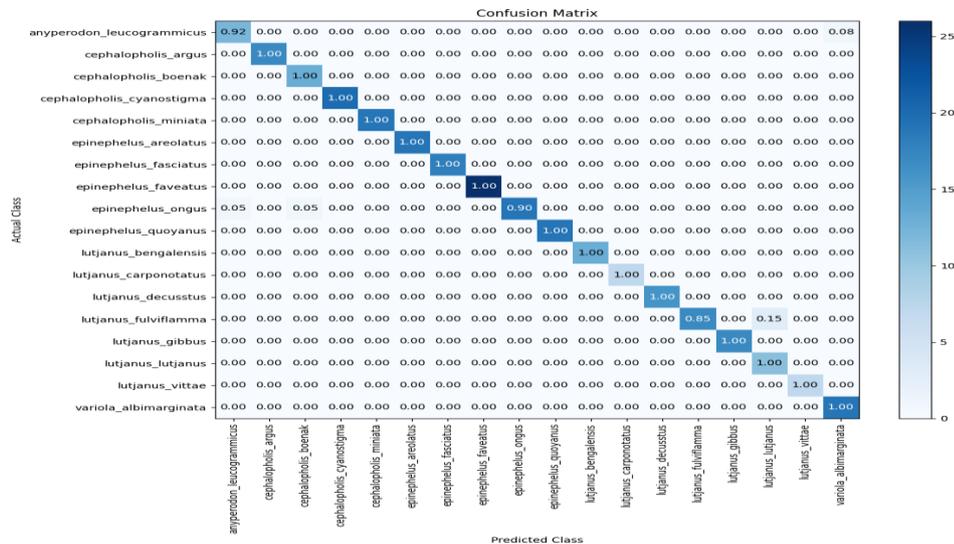


**Gambar 6. Hasil deteksi YOLO pada sampel**

Berdasarkan hasil deteksi pada Gambar 6 yang memperlihatkan beberapa sampel data uji tersebut, terlihat model YOLO berhasil mendeteksi dan memprediksi kelas atau jenis ikan dengan



benar. Selain itu, nilai *confidence* yang diberikan model dalam mendeteksi ikan berada di rentang 80% hingga 95% ke atas pada keseluruhan data uji.

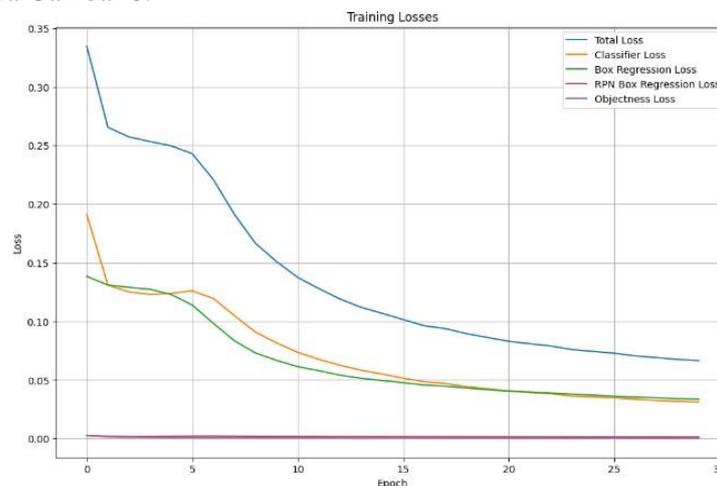


Gambar 7 Confusion matrix pada model YOLO

Berdasarkan grafik pada Gambar 22, model YOLO dapat dikatakan mampu mendeteksi jenis ikan yang ada. Hal tersebut dapat dilihat pada grafik yang menggambarkan mayoritas kelas pada model yang terdeteksi (*predicted class*) sesuai dengan kelas yang sebenarnya (*actual class*) serta tingkat akurasi pada tiap pertemuan *actual class* dan *predicted class* berada pada rentang 0,85 hingga 0,90 ke atas. Terakhir, nilai penentu dalam mengevaluasi kinerja model YOLO adalah dengan mengetahui nilai *precision*, *recall* dan *accuracy*. Untuk mengetahui nilai- nilai tersebut dapat menggunakan Persamaan 8, Persamaan 9 dan Persamaan 10. Contoh hasil perhitungan dari nilai *precision*, *recall* dan *accuracy* dijabarkan sebagai berikut.

## 2. Analisis Hasil Training dan Testing Model Faster R-CNN

Seperti yang telah dilakukan saat menganalisis model YOLO, pada model Faster R-CNN ini peneliti juga berfokus menganalisis tren nilai *loss* yang dihasilkan selama proses *train* berjalan dari awal hingga selesai agar dapat dibandingkan hasilnya dengan model YOLO. Grafik *tren* pada proses *train* dapat dilihat pada Gambar 8.

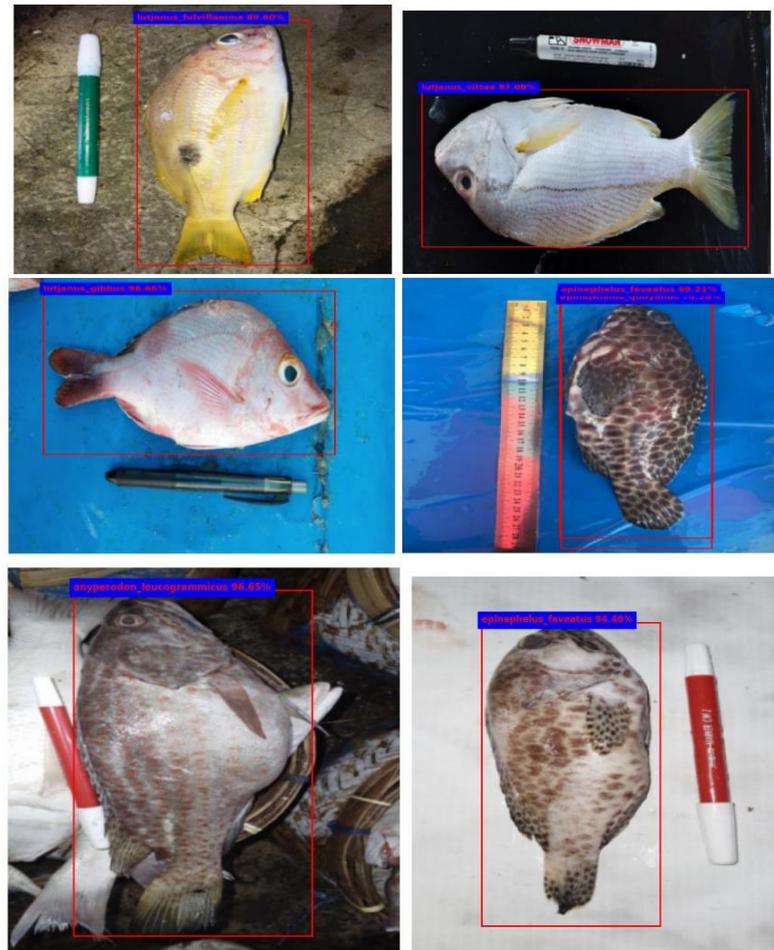


Gambar 8. Grafik loss pada Faster R-CNN



Berdasarkan Gambar 23, tren yang terlihat pada proses *training* model Faster R-CNN adanya penurunan nilai, sama seperti yang terjadi pada model YOLO selama proses *training* dijalankan. *Box loss* pada model ini menurun dari 0,139013 menjadi 0,029612, sementara *classification loss* menurun dari 0,205246 menjadi 0,024550. Penurunan signifikan pada *box loss* terlihat pada *epoch* 5 dan seterusnya, sedangkan penurunan pada *classification loss* terlihat pada *epoch* 6 dan seterusnya.

Setelah proses *training* selesai, dilanjutkan pada pengujian model menggunakan data yang telah dikategorikan menjadi data uji dengan contoh hasil *output* prediksi yang dapat ditunjukkan pada 9.



**Gambar 9 Hasil deteksi Faster R-CNN pada sampel**

Berdasarkan 9 hasil pengujian pada model Faster R-CNN yang telah dilatih sebelumnya, sama seperti model YOLO, model Faster R-CNN juga dapat mengenali dan mendeteksi jenis ikan dengan baik. Hal tersebut dapat dilihat pada prediksi jenis yang diberikan sesuai serta tingkat *confidence* yang cukup tinggi yang berada pada rentang 80% hingga 95%. Hanya saja masih terdapat beberapa *bounding box* prediksi yang terlihat menumpuk satu sama lain serta prediksi yang diberikan berbeda pada setiap kotaknya. Hal tersebut dapat terjadi karena adanya kemiripan warna, corak serta bentuk tubuh pada beberapa jenis ikan serta faktor eksternal lainnya sehingga adanya potensi kemunculan dari penumpukan *bounding box* tersebut.



### 3. Analisis Hasil *Testing* pada Jenis Ikan dengan Kemiripan Visual

Pengujian kali ini menggunakan metode cosine similarity untuk melihat seberapa mirip citra-citra ikan dari berbagai spesies, dimana metode ini mengukur seberapa dekat kesamaan antara dua citra dalam bentuk angka antara 0 hingga 1. Nilai 1 berarti kedua gambar sangat mirip, sedangkan nilai 0 berarti tidak ada kesamaan antara keduanya. Hasil dari pengujian ini dapat dilihat pada Tabel 2.

**Tabel 2. Hasil pengujian *cosine similarity***

| Kelompok | Jenis 1                     | Jenis 2                      | Cosine Similarity |
|----------|-----------------------------|------------------------------|-------------------|
| 1        | <i>Cephalopholis</i>        | <i>Cephalopholis</i>         | 0,7176            |
|          | <i>cyanostigma</i>          | <i>miniata</i>               |                   |
| 2        | <i>Epinephelus faveatus</i> | <i>Epinephelus ongus</i>     | 0,7224            |
|          | <i>Epinephelus faveatus</i> | <i>Epinephelus quoyanus</i>  | 0,5054            |
|          | <i>Epinephelus ongus</i>    | <i>Epinephelus quoyanus</i>  | 0,5254            |
| 3        | <i>Lutjanus bengalensis</i> | <i>Lutjanus carponotatus</i> | 0,6169            |
|          | <i>Lutjanus bengalensis</i> | <i>Lutjanus decussatus</i>   | 0,4881            |
|          | <i>Lutjanus</i>             | <i>Lutjanus decussatus</i>   | 0,6557            |
|          | <i>carponotatus</i>         |                              |                   |
| 4        | <i>Cephalopholis argus</i>  | <i>Cephalopholis boenak</i>  | 0,6454            |

Tabel 2 menunjukkan hasil pengujian *cosine similarity* antara citra ikan dari empat kelompok berbeda. Pada Kelompok 1, *Cephalopholis cyanostigma* dan *Cephalopholis miniata* memiliki kemiripan visual tinggi dengan nilai *cosine similarity* 0,7176. Kelompok 2 menunjukkan bahwa *Epinephelus faveatus* dan *Epinephelus ongus* memiliki kemiripan tertinggi dengan nilai 0,7224, sementara pasangan lainnya dalam kelompok ini memiliki nilai yang lebih rendah, sekitar 0,5 hingga 0,6. Kelompok 3 memperlihatkan bahwa *Lutjanus bengalensis* dan *Lutjanus carponotatus* memiliki nilai *cosine similarity* 0,6169, sedangkan kemiripan *Lutjanus bengalensis* dengan *Lutjanus decussatus* lebih rendah yaitu 0,4881. Pasangan *Lutjanus carponotatus* dan *Lutjanus decussatus* memiliki nilai 0,6557. Kelompok 4, pasangan *Cephalopholis argus* dan *Cephalopholis boenak* menunjukkan kemiripan visual dengan nilai 0,6454. Secara keseluruhan, hasil pengujian menunjukkan bahwa beberapa pasangan data citra tersebut memiliki tingkat kemiripan visual yang tinggi dengan nilai *cosine similarity* di atas 0,7, sementara pasangan lainnya menunjukkan kemiripan yang lebih rendah dengan nilai di kisaran 0,5 hingga 0,6.

Selain pengujian dengan *cosine similarity* tadi, *confusion matrix* juga digunakan untuk menguji model pada data dengan kemiripan visual tersebut sebagai penunjang analisis. Adapun *confusion matrix* pada pengujian model YOLO dan Faster R-CNN dapat dilihat pada 10 dan Gambar 11.



| True Label / Predicted Label | C. argus | C. boenak | C. cyanostigma | C. miniata | E. faveatus | E. ongus | E. quoyanus | L. bengalensis | L. carponotatus | L. decussatus | Unknown |
|------------------------------|----------|-----------|----------------|------------|-------------|----------|-------------|----------------|-----------------|---------------|---------|
| C. argus                     | 16       | 1         | 0              | 0          | 0           | 0        | 0           | 0              | 0               | 0             | 1       |
| C. boenak                    | 0        | 18        | 0              | 0          | 0           | 0        | 0           | 0              | 0               | 0             | 0       |
| C. cyanostigma               | 0        | 0         | 18             | 1          | 0           | 0        | 0           | 0              | 0               | 0             | 0       |
| C. miniata                   | 0        | 0         | 0              | 14         | 0           | 0        | 0           | 0              | 0               | 0             | 3       |
| E. faveatus                  | 0        | 0         | 0              | 0          | 8           | 1        | 0           | 0              | 0               | 0             | 11      |
| E. ongus                     | 0        | 0         | 0              | 0          | 1           | 17       | 0           | 0              | 0               | 0             | 0       |
| E. quoyanus                  | 0        | 0         | 0              | 0          | 0           | 0        | 18          | 0              | 0               | 0             | 2       |
| L. bengalensis               | 0        | 0         | 0              | 0          | 0           | 0        | 0           | 18             | 0               | 0             | 2       |
| L. carponotatus              | 0        | 0         | 0              | 0          | 0           | 0        | 0           | 0              | 19              | 0             | 0       |
| L. decussatus                | 0        | 0         | 0              | 0          | 0           | 0        | 0           | 0              | 0               | 17            | 3       |

**Gambar 10. Confusion matrix untuk jenis ikan yang mirip secara visual pada model YOLO**

| True Label / Predicted Label | C. argus | C. boenak | C. cyanostigma | C. miniata | E. faveatus | E. ongus | E. quoyanus | L. bengalensis | L. carponotatus | L. decussatus | Unknown |
|------------------------------|----------|-----------|----------------|------------|-------------|----------|-------------|----------------|-----------------|---------------|---------|
| C. argus                     | 16       | 1         | 0              | 0          | 0           | 0        | 0           | 0              | 0               | 0             | 0       |
| C. boenak                    | 0        | 18        | 0              | 0          | 0           | 0        | 0           | 0              | 0               | 0             | 3       |
| C. cyanostigma               | 1        | 0         | 18             | 0          | 0           | 0        | 0           | 0              | 0               | 0             | 1       |
| C. miniata                   | 0        | 0         | 0              | 18         | 0           | 0        | 0           | 0              | 0               | 0             | 2       |
| E. faveatus                  | 0        | 0         | 0              | 0          | 14          | 0        | 1           | 0              | 0               | 0             | 5       |
| E. ongus                     | 0        | 0         | 0              | 0          | 0           | 17       | 0           | 0              | 0               | 0             | 3       |
| E. quoyanus                  | 0        | 0         | 0              | 0          | 0           | 0        | 18          | 0              | 0               | 0             | 2       |
| L. bengalensis               | 0        | 0         | 0              | 0          | 0           | 0        | 0           | 18             | 0               | 0             | 2       |
| L. carponotatus              | 0        | 0         | 0              | 0          | 0           | 0        | 0           | 0              | 3               | 0             | 17      |
| L. decussatus                | 0        | 0         | 0              | 0          | 0           | 0        | 0           | 0              | 0               | 4             | 16      |

**Gambar 11 Confusion matrix untuk jenis ikan yang mirip secara visual pada Faster R-CNN**

Berdasarkan pada Gambar 10 dan Gambar 11, hasil pengujian pada confusion matrix untuk model YOLO dan Faster R-CNN menunjukkan bahwa kedua model memiliki tantangan dalam mengklasifikasikan spesies yang sangat mirip satu sama lain. Khusus untuk YOLO, walaupun umumnya model ini berkinerja baik, terdapat kecenderungan yang signifikan untuk mengklasifikasikan spesies *Epinephelus faveatus* sebagai 'Unknown' dengan sebanyak 11 kasus tidak teridentifikasi. Hal ini menandakan bahwa YOLO kurang efektif dalam menangkap fitur-fitur spesifik yang membedakan *Epinephelus faveatus* dari spesies yang serupa, meskipun kondisi pencahayaan dan kualitas gambar dianggap cukup untuk deteksi.

Sementara itu, Faster R-CNN menunjukkan jumlah 'Unknown' yang lebih tinggi pada kelompok *Lutjanus bengalensis*, *Lutjanus carponotatus* dan *Lutjanus decussatus* dengan total kesalahan serupa pada kelompok *Epinephelus faveatus*, *Epinephelus ongus* dan *Epinephelus quoyanus*. Hal ini mengindikasikan bahwa Faster R-CNN juga mengalami kesulitan dalam mengatasi varian visual yang serupa antar spesies tersebut, khususnya dalam situasi di mana ciri-ciri diskriminatif antara spesies-spesies tersebut tidak mencolok atau gambar yang dianalisis tidak memiliki kualitas yang memadai untuk ekstraksi ciri dengan tepat. Dari hasil pengujian yang telah dilakukan, tampak jelas bahwa kedua model menghadapi tantangan yang signifikan dalam membedakan antara spesies yang memiliki kemiripan visual tinggi.

#### 4. Analisis Hasil Estimasi Berat Menggunakan Citra

Untuk melakukan estimasi berat pada data primer, terlebih dahulu model mendeteksi jenis ikan pada data masukan kemudian proses estimasi berat dilakukan menggunakan koefisien yang sesuai. Untuk hasil deteksi pada data primer dapat dilihat pada Tabel 3.



**Tabel 3 Hasil deteksi pada data citra**

| Data   | Hasil Deteksi | Hasil Deteksi | Keterangan   |
|--------|---------------|---------------|--|
|        | YOLO          | Faster R-CNN  |  |
| Data 1 | Benar         | Benar         | Kedua model dapat mendeteksi jenis ikan pada data primer dengan benar.       |
| Data 2 | Benar         | Benar         | Kedua model dapat mendeteksi jenis ikan pada data primer dengan benar.       |
| Data 3 | Benar         | Benar         | Kedua model dapat mendeteksi jenis ikan pada data primer dengan benar.       |
| Data 4 | Benar         | Benar         | Kedua model dapat mendeteksi jenis ikan pada data primer dengan benar.       |
| Data 5 | Benar         | Salah         | YOLO dapat mendeteksi jenis ikan dengan benar, sedangkan Faster R-CNN tidak. |

Berdasarkan pada Tabel 8 model YOLO berhasil mendeteksi jenis ikan dengan benar. Model Faster R-CNN juga menunjukkan kemampuan deteksi yang baik, meskipun terjadi kesalahan deteksi pada Data 5. Selain hasil deteksi, berikut hasil estimasi berat pada citra menggunakan model YOLO yang ditunjukkan pada Tabel 4.

**Tabel 4 Hasil estimasi berat pada citra menggunakan model YOLO**

| No | Data   | Berat Aktual<br>(gram) | Berat Prediksi<br>(gram) |
|----|--------|------------------------|--------------------------|
| 1  | Data 1 | 210                    | 214,14                   |
| 2  | Data 2 | 180                    | 177,83                   |
| 3  | Data 3 | 260                    | 261,99                   |
| 4  | Data 4 | 230                    | 238,06                   |
| 5  | Data 5 | 200                    | 209,31                   |

Pada Tabel 4 telah ditampilkan hasil estimasi berat yang berdasarkan pada jenis yang terdeteksi oleh model YOLO sebagai penentuan konstanta jenis dan pertumbuhan yang diperoleh menggunakan nilai pendekatan, sehingga sistem dapat menentukan estimasi berat dari ikan yang terdeteksi. Berdasarkan nilai aktual dan nilai prediksi yang ada pada Tabel 9 maka dapat dilakukan perhitungan MAPE yang telah dijabarkan pada Persamaan 11. Hasil evaluasi estimasi berat pada citra menggunakan model YOLO ditunjukkan pada Tabel 5.

**Tabel 5 Evaluasi hasil estimasi berat pada citra menggunakan model YOLO**

| No | Data   | Error |
|----|--------|-------|
| 1  | Data 1 | 1,97% |



|             |        |              |
|-------------|--------|--------------|
| 2           | Data 2 | 1,2%         |
| 3           | Data 3 | 0,76%        |
| 4           | Data 4 | 3,5%         |
| 5           | Data 5 | 4,65%        |
| <b>MAPE</b> |        | <b>2,42%</b> |

Pada Tabel 5, terlihat bahwa jenis Data 5 memiliki MAPE sebesar 4,65%, yang mengindikasikan bahwa model YOLO memiliki tingkat kesalahan dalam memperkirakan berat untuk spesies ini. Di sisi lain, Data 3 menunjukkan MAPE terendah yaitu hanya 0,76%, menandakan estimasi model sangat akurat untuk jenis ikan tersebut. Untuk data lainnya memberikan nilai MAPE yang dihasilkan beragam yang mencerminkan tingkat akurasi yang berbeda dalam estimasi berat oleh model. MAPE keseluruhan sebesar 2,42% mengindikasikan bahwa model memiliki tingkat kesalahan rata-rata yang rendah dalam memperkirakan berat ikan. Sama seperti pada model YOLO, berikut hasil estimasi berat pada citra menggunakan model Faster R-CNN yang ditunjukkan pada Tabel 6.

**Tabel 6 Hasil estimasi berat pada citra menggunakan model Faster R-CNN**

| No | Data   | Berat Aktual<br>(gram) | Berat Prediksi<br>(gram) |
|----|--------|------------------------|--------------------------|
| 1  | Data 1 | 210                    | 238,31                   |
| 2  | Data 2 | 180                    | 191,74                   |
| 3  | Data 3 | 260                    | 261,99                   |
| 4  | Data 4 | 230                    | 192,41                   |
| 5  | Data 5 | 200                    | 127,95                   |

Pada Tabel 6 ditampilkan hasil estimasi berat yang berdasarkan pada jenis yang terdeteksi oleh model Faster R-CNN. Pada tahap evaluasi, MAPE juga digunakan sebagai pengukur tingkat simpangan atau kesalahan yang ada dari hasil estimasi. Untuk hasil evaluasi estimasi berat pada citra menggunakan model Faster R-CNN yang ditunjukkan pada Tabel 7.

**Tabel 7 Evaluasi hasil estimasi berat pada citra menggunakan model Faster R-CNN**

| No          | Data   | Error         |
|-------------|--------|---------------|
| 1           | Data 1 | 13,48%        |
| 2           | Data 2 | 6,52%         |
| 3           | Data 3 | 0,76%         |
| 4           | Data 4 | 16,34%        |
| 5           | Data 5 | 36,02%        |
| <b>MAPE</b> |        | <b>14,63%</b> |



Berdasarkan Tabel 7, terdapat kesalahan dalam identifikasi jenis ikan pada Data 5 sehingga memberikan nilai MAPE sebesar 36,02%, yang berarti bahwa estimasi berat untuk jenis ikan ini memiliki tingkat kesalahan yang cukup tinggi. Untuk Data 1 memiliki MAPE sebesar 13,48%, Data 2 dengan MAPE 6,52%, Data 3 dengan MAPE yang sangat rendah yaitu 0,76% dan Data 4 dengan MAPE 16,34%. Nilai-nilai ini menunjukkan bahwa model Faster R-CNN memiliki tingkat akurasi yang lebih rendah dalam memperkirakan berat ikan dibandingkan dengan model YOLO.

Secara keseluruhan, MAPE untuk model Faster R-CNN adalah 14,63%, yang lebih tinggi dibandingkan MAPE keseluruhan dari model YOLO. Ini menunjukkan bahwa pada model YOLO lebih akurat dalam estimasi berat ikan pada data citra digital jika dibandingkan dengan model Faster R-CNN.

### SIMPULAN DAN SARAN

Penelitian ini menunjukkan bahwa YOLO memiliki performa lebih baik dibandingkan Faster R-CNN dalam mendeteksi jenis dan mengestimasi berat ikan kerapu serta kakap. Model YOLO mencapai presisi sebesar 0,98, sedangkan Faster R-CNN memperoleh 0,97. Kedua model menunjukkan hasil yang baik dalam recall dan akurasi selama pengujian. Dalam estimasi berat, YOLO mencatatkan Mean Absolute Percentage Error (MAPE) sebesar 2,42% untuk citra dan 3,66% untuk video, sedangkan Faster R-CNN memiliki MAPE sebesar 14,63% untuk citra dan 13,6% untuk video. Kesalahan dalam identifikasi jenis pada Faster R-CNN berdampak pada tingkat kesalahan yang lebih tinggi dalam estimasi berat.

Untuk pengembangan lebih lanjut, diperlukan optimalisasi skenario pengambilan data dan pengaturan parameter model agar deteksi semakin akurat. Selain itu, sistem yang telah dikembangkan dapat diimplementasikan lebih lanjut sebagai sistem grading ikan otomatis berdasarkan kategori jenis dan berat yang telah ditentukan. Hal ini dapat meningkatkan efisiensi dalam industri perikanan serta membantu dalam penentuan harga dan kualitas ikan secara lebih objektif.

### DAFTAR PUSTAKA

- Anderson, W. D. (1967). *A review of the Lutjanidae, a family of snappers*. Bulletin of Marine Science, **17**(4), 902-985.
- Bashir, R. H., Ali, M. A., Ullah, A., & Ahmed, M. (2019). *Aquaculture development and its role in global food security: A review*. Journal of Fisheries and Aquatic Sciences, **10**(2), 85-97. <https://doi.org/xxxx>
- Chatla, S., Rajan, R. P., & Kizhakudan, J. K. (2019). *A review on groupers (Family: Serranidae) with special reference to their fishery, biology and aquaculture potential in India*. Indian Journal of Fisheries, **66**(3), 5-19. <https://doi.org/xxxx>
- Craig, M. T., & Hastings, P. A. (2007). *A molecular phylogeny of the groupers (Serranidae: Epinephelinae) with a revised classification of the Epinephelini*. Ichthyological Research, **54**(1), 1-17. <https://doi.org/xxxx>
- Darwin, M., Wahyu, I., & Saputra, A. (2020). *Habitat and distribution of groupers in Indonesian waters: A review*. Indonesian Journal of Marine Science, **15**(2), 78-89.
- Ding, S., Liu, M., & Zhang, X. (2006). *Morphological diversity and identification challenges in the genus Epinephelus (Serranidae: Epinephelinae)*. Journal of Fish Biology, **68**(3), 925-937.



- 
- Fernandes, A. F., Soares, F. M., & Rocha, L. M. (2020). *Image processing techniques for fish species identification and weight estimation*. International Journal of Computer Vision and Image Processing, **8**(1), 45-62. <https://doi.org/xxxx>
- Ningsih, T. (2015). *Karakteristik morfologi ikan kerapu Epinephelus spp. di Perairan Indonesia*. Jurnal Perikanan dan Kelautan, **7**(2), 56-63.
- Zainuddin, Zahir. "Detection and Counting of the Number of Cocoa Fruits on Trees Using UAV." *2023 IEEE International Conference on Industry 4.0, Artificial Intelligence, and Communications Technology (IAICT)*. IEEE, 2023.
- Oktaviyani, R. (2018). *Analisis potensi perikanan kakap di Indonesia: Studi kasus perikanan tangkap di perairan tropis*. Jurnal Sumber Daya Perikanan, **5**(1), 45-59.
- Saha, S., Chanda, A., & Basu, S. (2018). *Comparative analysis of Lutjanidae fish species: Distribution, biology, and ecological significance*. Marine Ecology Research, **12**(4), 213-225.