

IMPLEMENTASI YOLOV8 (YOU ONLY LOOK ONCE) UNTUK DETEKSI SAMPAH DI LINGKUNGAN PERAIRAN INDONESIA

Muhammad Diko Aprilyanto, Reni Rahmadewi

Teknik Elektro, Universitas Singaperbangsa Karawang

Jl. HS.Ronggo Waluyo, Puseurjaya, Telukjambe Timur, Karawang, Jawa Barat 41361, Indonesia

2110631160051@student.unsika.ac.id

ABSTRAK

Peningkatan sampah di lingkungan perairan, khususnya di Indonesia, menimbulkan ancaman serius bagi ekosistem laut dan kesehatan manusia. Permasalahan ini diperburuk oleh kurangnya sistem deteksi yang efisien dan akurat dalam mengidentifikasi sampah di dalam air. Penelitian ini bertujuan mengembangkan sistem deteksi sampah dalam air menggunakan YOLOv8, algoritma deteksi objek yang unggul dalam akurasi dan kecepatan. Dataset citra sampah di dalam air yang terdiri atas 5.130 gambar dianalisis, melibatkan proses anotasi, pelatihan model, dan evaluasi dengan metrik seperti precision, recall, dan mean average precision (mAP). Model dilatih menggunakan 40 epoch dengan ukuran gambar 640x640 piksel, menghasilkan nilai *precision* 0,756 (75%), *recall* 0,651 (65%), dan mAP 0,741 (74%) untuk deteksi keseluruhan. Hasil menunjukkan model efektif mendeteksi sampah seperti diantaranya yaitu plastik, elektronik, dan botol dengan performa yang bervariasi. Kelemahan identifikasi pada deteksi logam yang membutuhkan optimasi lebih lanjut. Sistem ini diharapkan menjadi alat pendukung bagi relawan dan petugas kebersihan untuk mengidentifikasi sampah di perairan secara efisien, meningkatkan upaya pelestarian lingkungan.

Kata kunci : Sampah di Dalam Air, Deteksi Sampah, YOLOv8, Computer Vision, Klasifikasi Objek

1. PENDAHULUAN

Air merupakan sumber daya vital bagi kehidupan semua makhluk hidup. Selain untuk konsumsi, air juga mendukung berbagai aktivitas manusia. Namun, ketersediaan dan kualitas air terus menurun akibat meningkatnya kebutuhan serta pencemaran dari aktivitas industri, domestik, dan pembuangan sampah. Kondisi ini membahayakan makhluk hidup yang bergantung pada air. [1].

Sampah plastik menjadi masalah yang semakin serius dan mengancam ekosistem laut global. Mikroplastik telah ditemukan di laut dan lingkungan lainnya, berisiko dikonsumsi oleh ikan dan akhirnya oleh manusia. Selain itu, limbah plastik mengganggu jalur transportasi laut serta merusak terumbu karang dan padang lamun, yang berdampak pada keseimbangan ekosistem laut. [2].

Sebagai negara maritim, Indonesia menghadapi masalah penumpukan sampah di perairan akibat pengelolaan yang belum optimal. Salah satu solusi adalah penggunaan robot otomatis untuk mengumpulkan sampah, yang memerlukan sistem deteksi objek dalam video. Namun, metode deteksi konvensional masih terbatas, terutama saat kualitas gambar menurun atau objek tertutup [3].

Perkembangan teknologi Computer Vision membuka banyak peluang di berbagai bidang seperti pengolahan citra dan visi mesin, dengan teknik dasar yang serupa. Teknologi ini berkaitan erat dengan disiplin lain seperti kecerdasan buatan, robotika, otomasi industri, pengolahan sinyal, dan neurobiologi. Saat ini, Computer Vision dimanfaatkan untuk berbagai keperluan seperti deteksi wajah, pengenalan ekspresi, dan sering digabungkan dengan jaringan saraf tiruan [4].

Adapun salah satu penerapan penggunaan deteksi objek dapat digunakan sebagai solusi terhadap masalah ini adalah deteksi sampah di dalam air yang berguna untuk mendeteksi beberapa jenis sampah tertentu seperti plastik, jaring ikan, ataupun benda elektronik yang ada di dalam laut secara langsung. Sistem deteksi ini bertujuan untuk memudahkan orang-orang seperti penyelam atau petugas kebersihan apabila ingin melakukan pembersihan sampah di dalam air.

Deteksi objek menghadapi berbagai tantangan, seperti keragaman bentuk, ukuran, orientasi, serta faktor eksternal seperti pencahayaan, posisi, dan kualitas gambar yang dapat memengaruhi akurasi deteksi [5]. Oleh karena itu, diperlukan model yang andal untuk menangani variabilitas tersebut, salah satunya adalah arsitektur YOLO (You Only Look Once), khususnya versi terbarunya, YOLOv8, yang dikenal unggul dalam hal akurasi dan kecepatan. Penggunaan YOLOv8 untuk mendeteksi sampah di dalam air dapat mempercepat proses pencarian di wilayah perairan yang luas, meningkatkan efisiensi pembersihan, serta memungkinkan pencatatan data deteksi secara digital dan terstruktur. Dengan begitu, riwayat deteksi sampah dapat diakses lebih mudah dan membantu petugas kebersihan menjangkau area yang lebih luas secara efektif dalam jangka panjang.

Penelitian ini bertujuan untuk merancang model deteksi sampah di dalam air menggunakan YOLOv8 yang dapat mendeteksi sekaligus mengklasifikasikan beberapa jenis sampah yang dibuang atau dijatuhkan masyarakat dengan efektif dan efisien. Dengan menerapkan arsitektur YOLOv8, diharapkan sistem ini dapat menjadi alat pendukung yang handal bagi petugas kebersihan ataupun relawan dalam proses

identifikasi jenis sampah secara cepat dan akurat agar bisa membersihkan sampah di area perairan yang luas secara efisien.

2. TINJAUAN PUSTAKA

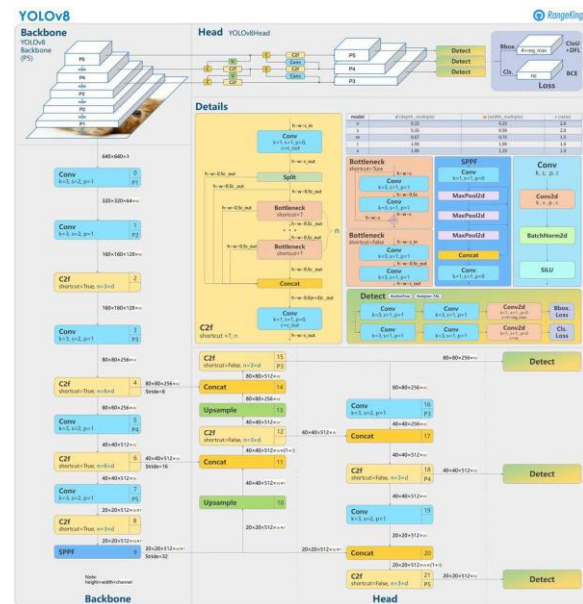
Pencemaran air akibat sampah telah menjadi permasalahan lingkungan yang semakin serius seiring dengan meningkatnya aktivitas industri dan domestik. Sampah di perairan, terutama plastik dan elektronik, berkontribusi terhadap degradasi ekosistem laut serta membahayakan kesehatan manusia. Mikroplastik yang berasal dari limbah plastik dapat dikonsumsi oleh biota laut dan masuk ke dalam rantai makanan manusia, yang pada akhirnya berpotensi menimbulkan gangguan kesehatan [1]. Selain itu, sampah elektronik dan logam mengandung bahan kimia berbahaya seperti merkuri dan timbal yang dapat mencemari air dan tanah, serta menyebabkan keracunan bagi organisme akuatik [2].

Dalam bidang kecerdasan buatan, Computer Vision telah banyak digunakan untuk mendeteksi objek dalam berbagai lingkungan. Computer Vision merupakan teknologi yang memungkinkan komputer untuk memahami dan menafsirkan data visual, dengan salah satu metode utamanya adalah Convolutional Neural Network (CNN). CNN telah terbukti efektif dalam mengklasifikasikan objek dengan tingkat akurasi tinggi dan digunakan dalam berbagai aplikasi, termasuk pengenalan wajah, analisis citra medis, serta deteksi objek pada gambar atau video [4][5].

Salah satu pendekatan yang paling efektif dalam deteksi objek adalah YOLO (You Only Look Once). YOLO merupakan algoritma deep learning yang dikenal dengan kecepatannya dalam melakukan deteksi objek secara real-time. Versi terbaru dari algoritma ini, yaitu YOLOv8, menawarkan peningkatan akurasi dan efisiensi dibandingkan dengan versi sebelumnya. YOLOv8 menggunakan pendekatan anchor-free yang memungkinkan prediksi bounding box secara langsung pada pusat objek, sehingga meningkatkan keakuratan deteksi terhadap objek dengan berbagai ukuran dan orientasi [6]. Beberapa penelitian telah menunjukkan bahwa YOLOv8 memiliki keunggulan dalam mendeteksi berbagai objek dalam kondisi lingkungan yang kompleks, termasuk dalam deteksi sampah di perairan [8].

2.1. Arsitektur YOLOv8

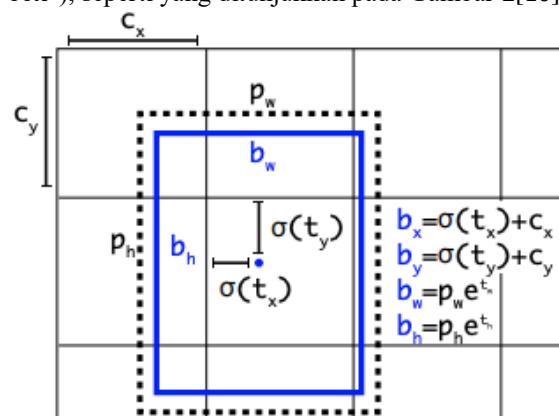
Dalam penelitian ini menggunakan arsitektur YOLOv8 (You Only Look Once) yang dirilis pada tahun 2023[7] sebagai metode penelitiannya. Algoritma YOLOv8 adalah pengembangan terbaru yang berbasis pada Convolutional Neural Network (CNN) yang dibuat oleh ultralytics[8]. YOLO, singkatan dari You Only Look Once, adalah salah satu algoritma deteksi objek yang mampu mengidentifikasi objek dalam gambar atau video secara real-time.



Gambar 1. Bentuk Arsitektur YOLOv8
(Sumber: Y. Yanto, *et al.* 2023)

Arsitektur YOLOv8, seperti yang ditampilkan pada Gambar 2, terdiri atas tiga komponen utama: jaringan backbone, neck, dan head. Jaringan backbone menggunakan Feature Pyramid Network (FPN) untuk mengekstraksi fitur dari gambar masukan. Sementara itu, neck memanfaatkan serangkaian cross-layer connection untuk meningkatkan kualitas fitur yang diekstraksi. Komponen head kemudian menggunakan fitur yang telah ditingkatkan ini untuk memprediksi kotak pembatas, skor kelas objek, serta tingkat akurasi setiap objek dalam gambar[9].

Model ini menggunakan anchor-free detection, yang memungkinkan prediksi langsung pada pusat objek tanpa bergantung pada kotak jangkar (“anchor box”), seperti yang ditunjukkan pada Gambar 2[10].



Gambar 2. Visualisasi anchor box dalam YOLO
(Sumber: Y. Yanto, F. Aziz, dan I. Irmawati. 2023)

Secara umum, YOLO membagi gambar input menjadi beberapa grid yang terdiri dari sel-sel. Setiap sel dalam grid ini bertugas memprediksi beberapa bounding box, beserta kemungkinan kelas objek yang mungkin ada di dalamnya [8].

2.2. Google Colab



Gambar 3. Logo Google Colab
(Sumber: colab.google.com)

Google Colab merupakan *platform cloud computing* dari Google yang memungkinkan pengguna menjalankan kode *Python* langsung di *browser* tanpa perlu instalasi lokal. Platform ini banyak digunakan untuk pemrosesan data, pelatihan model kecerdasan buatan (AI), dan analisis data[11].

2.3. Python



Gambar 4. Logo Python
(Sumber: python.org)

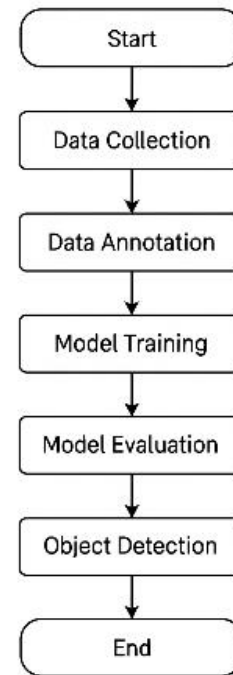
Python adalah bahasa pemrograman tingkat tinggi yang dikembangkan oleh Guido Van Rossum dan pertama kali dirilis pada tahun 1991. Saat ini, Python menjadi salah satu bahasa pemrograman yang sangat populer. Python juga bersifat serbaguna, dengan berbagai aplikasi seperti *machine learning* dan *deep learning*. Ada banyak IDE yang dapat menuliskan Bahasa Python, mulai dari VSCode, Pycharm, Jupyter Notebook, dan Google Colab[12].

Beberapa penelitian sebelumnya menunjukkan bahwa YOLOv8 efektif dalam mendeteksi berbagai jenis sampah di perairan, seperti plastik, botol, jaring ikan, dan elektronik, dengan nilai *precision* dan *recall* yang tinggi [3]. Namun, salah satu tantangan utama dalam deteksi sampah di perairan adalah mengenali objek berbahan logam, karena karakteristik visualnya yang lebih kompleks dan tingkat reflektansi cahaya yang tinggi [7].

Beberapa penelitian terdahulu menunjukkan bahwa metode YOLO memiliki berbagai keunggulan, seperti kecepatan deteksi yang lebih tinggi dibandingkan metode lain seperti Faster R-CNN dan SSD, akurasi yang lebih baik dalam kondisi pencahayaan yang cukup, serta kemampuan untuk mendeteksi berbagai jenis sampah dalam satu model

[10][9][14]. Namun, terdapat beberapa kekurangan yang perlu diatasi, seperti kesulitan dalam mendeteksi objek yang berada di bawah permukaan air yang keruh, akurasi yang lebih rendah dalam mendeteksi sampah berbahan logam, serta kurangnya variasi dalam dataset yang digunakan sehingga mengurangi kemampuan generalisasi model [15][16][12].

Penelitian ini bertujuan untuk mengembangkan sistem deteksi sampah di perairan menggunakan YOLOv8 dengan pendekatan yang lebih optimal. Adapun metode YOLOv8 yang digunakan yaitu pada Gambar 5.



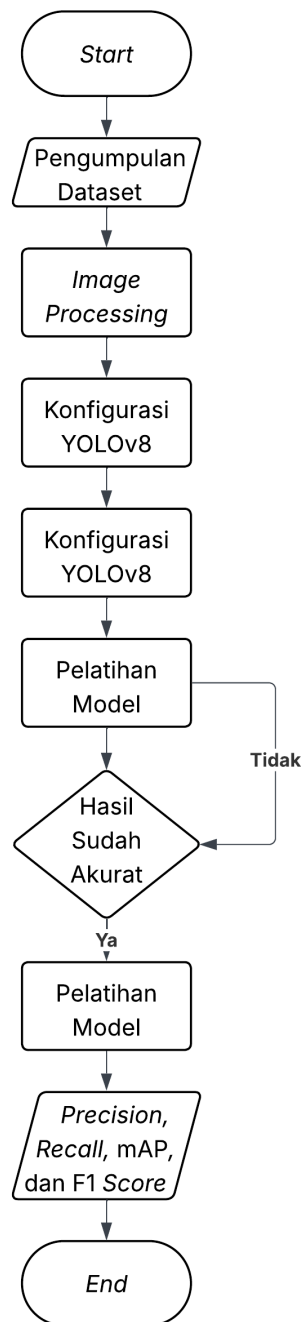
Gambar 5. Flowchart Metode YOLOv8

Berbeda dengan penelitian sebelumnya, model yang dikembangkan dalam penelitian ini menggunakan dataset yang lebih besar dan beragam untuk meningkatkan akurasi deteksi, terutama pada objek logam. Dengan pendekatan ini, diharapkan sistem dapat membantu relawan dan petugas kebersihan dalam mendeteksi dan mengklasifikasikan sampah di perairan secara lebih efisien.

3. METODE PENELITIAN

3.1. Alur Penelitian

Secara umum, alur penelitian yang akan dibuat dijelaskan dengan diagram blok pada Gambar 6 di bawah ini. Dalam metode penelitian ini dijabarkan beberapa tahapan yang akan dilakukan dalam proses deteksi sampah di dalam air.



Gambar 6. Flowchart Alur Penelitian

3.2. Dataset

Data adalah sekumpulan data operasional yang dibutuhkan sebelum dilakukan tahap *Data Mining*[13]. Dataset adalah sekumpulan objek yang dilengkapi dengan atributnya yang mengacu pada karakteristik atau sifat yang dimiliki oleh suatu objek [14]. Dataset yang digunakan berasal dari Kaggle, berisi citra sampah dalam air yang telah dianotasi untuk YOLOv8. Citra mencakup berbagai jenis sampah seperti sarung tangan, botol, dan *handphone* dari beragam sudut pandang.

Gambar 7. Dataset Sampah di Dalam Air
(Sumber: kaggle.com)

Dari dataset tersebut terdapat 5.130 gambar yang telah dilakukan *image pre-processing* dan diberi label dengan label sebelumnya yang terdiri dari data *train*, *test*, dan *validation* yang masing-masing memiliki jumlah gambar sebagai berikut [15].

Tabel 1. Jumlah Gambar Dataset Sampah di Dalam Air

<i>Train</i>	3.628 gambar
<i>Test</i>	501 gambar
<i>Validation</i>	1.001 gambar

Berdasarkan pada Tabel 1, jumlah dari gambar *train* terdiri dari 3.628 gambar atau 70% dari total yang akan dipakai saat *training* model dengan YOLOv8 yang nantinya akan hasil dari model tersebut akan dites dengan 501 gambar atau 15% dari total yang ada di *test*, lalu akan divalidasi hasil test tersebut dengan 15% sisanya atau 1.001 gambar yang ada di *validation*.

3.3. Image Pre-Processing

Proses image pre-processing dilakukan dengan memberi bounding box dan label pada gambar sampah di air menggunakan Roboflow, berdasarkan lokasi dan jenis sampah[16]. Setelah gambar dianotasi, maka disesuaikan ukurannya menjadi 640x640 piksel sesuai dengan standar penggunaan model YOLOv8 demi mencapai performa yang optimal dan akurat selama proses *training*.

3.4. Konfigurasi YOLOv8

Setelah tahap *image pre-processing* selesai, langkah selanjutnya adalah menyiapkan serta mengonfigurasi model YOLOv8. Proses ini dilakukan di Platform Google Colab, meliputi pengecekan perangkat keras akselerator, instalasi pustaka yang diperlukan, impor dataset, pengaturan *hyperparameter*, serta *training model*.

Pengecekan Google Colab dilakukan untuk memastikan ketersediaan GPU, lalu diinstal library

Ultralytics untuk YOLOv8, Matplotlib untuk visualisasi, dan NumPy untuk memproses data citra.

3.5. Pelatihan Model

Model deteksi sampah di dalam air dilatih dengan menggunakan bahasa pemrograman Python pada Google Colab untuk mengeksekusi code dengan nilai *hyperparameter epoch* sebesar 40 kali, *image size* 640x640 untuk evaluasi, dan ambang batas *confidence* 0.25.

Model yang telah dilatih disimpan di Google Drive yang terhubung dengan Google Colab, sehingga dapat diakses dan diuji kembali tanpa perlu pelatihan ulang.

3.6. Evaluasi dan Pengujian Model

Pengujian model dilakukan secara langsung di Google Colab dengan mengakses model yang sudah dilatih yang tersimpan di folder Google Drive. Evaluasi hasil pengujian dilakukan menggunakan berbagai metrik evaluasi, seperti *Precision*, *Recall*, *Mean Average Precision (mAP)*, dan *F1 Score*.

a. Precision

Precision adalah rasio antara *True Positive (TP)* dengan total data yang diprediksi sebagai positif, yaitu gabungan antara *True Positive (TP)* dan *False Positive (FP)*, seperti yang ditunjukkan pada Persamaan 1. *Precision* menggambarkan tingkat keakuratan hasil prediksi model dalam mengidentifikasi data yang sesuai dengan yang diharapkan[17].

$$Precision = \frac{TP}{TP + FP} \quad (1)$$

b. Recall

Recall adalah rasio antara *True Positive (TP)* dan total data positif, yaitu TP ditambah *False Negative (FN)*, seperti pada Persamaan 2. Nilai *recall* mencerminkan sejauh mana model mampu mengidentifikasi kembali informasi yang relevan [17].

$$Recall = \frac{TP}{TP + FN} \quad (2)$$

c. Mean Average Precision (mAP)

Mean Average Precision (mAP) adalah rata-rata dari *Average Precision (AP)* tiap kategori objek, dihitung berdasarkan jumlah kelas (N) dan indeks iterasi (i) seperti pada Persamaan 3.

$$mAP = \frac{1}{N} \sum_{k=1}^{k=N} AP_k \times 100\% \quad (3)$$

d. F1 Score

F1 Score adalah rata-rata harmonis antara *precision* dan *recall*, seperti yang dijelaskan pada Persamaan 4. *F1 Score* mencapai nilai optimal apabila model memiliki *precision* dan *recall* yang tinggi. Nilai terbaik *F1 Score* adalah 1.0, sedangkan nilai terendah adalah 0.0. *F1 Score* sering digunakan pada dataset yang menghadapi masalah ketidakseimbangan, yaitu ketika jumlah

False Negative (FN) dan *False Positive (FP)* tidak sebanding.

$$\frac{1}{F1} = \frac{1}{2} \left(\frac{1}{precision} + \frac{1}{recall} \right) \quad (4)$$

e. Confusion Matrix

Confusion matrix adalah tabel yang membandingkan hasil prediksi model dengan data sebenarnya, dan digunakan untuk mengevaluasi kinerja model seperti pada Gambar 8.

		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Gambar 8. Tabel Confusion Matrix
(Sumber: H. Nurhadiati, et al. 2022)

4. HASIL DAN PEMBAHASAN

Dari hasil pengujian yang telah dilaksanakan menggunakan YOLOV8 sebagai model untuk mendeteksi sampah di dalam air, maka penulis menggunakan 40 *epochs* seperti yang dapat dilihat pada Gambar 11 untuk memperbesar nilai *mAP*, *recall*, dan *precision* dari model ini.

```
!pip install ultralytics
```

Gambar 9. Instalasi Library

```
from ultralytics import YOLO
import numpy as np
from PIL import Image
import requests
from io import BytesIO
import cv2
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns
import os
from google.colab.patches import cv2_imshow
```

Gambar 10. Memanggil Library di Google Colab

```
# Build from YAML and transfer weights
Final_model = YOLO('yolov8n.yaml').load('yolov8n.pt')

# Training The Final Model
Result_Final_model = Final_model.train(
    data="/content/underwater_plastics/data.yaml",
    epochs=40, imgsz = 640, batch = 16 ,lr0=0.01, dropout= 0.15,
    device = 0
)
```

Gambar 11. Konfigurasi Hyperparameter

Seperti terlihat pada Gambar 9 dan 10, peneliti menginstal library ultralytics dan memanggil beberapa library seperti matplotlib.pyplot, YOLO, numpy,

pandas, dan seaborn untuk membangun model, menampilkan hasil training dalam bentuk plot, serta memvisualisasikan metrik pelatihan dan validasi. Parameter pelatihan ditampilkan pada Tabel 2.

Tabel 2. Parameter Pelatihan Model

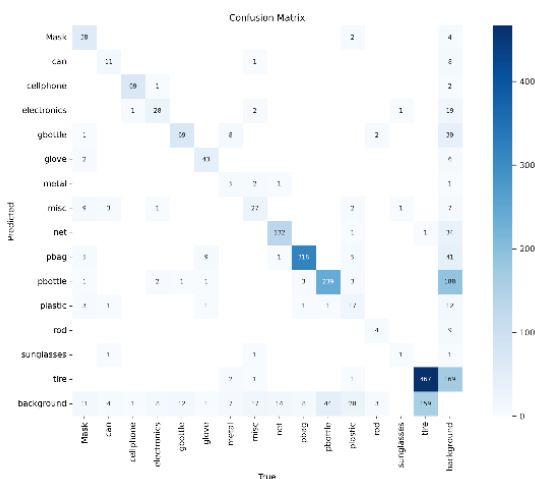
Parameter	Nilai
Epochs	40
Image Size	640
Batch	16

Setelah menentukan parameter untuk melatih model deteksi sampah di dalam air, selanjutnya adalah menjalankan sintaks yang telah disiapkan untuk melakukan pelatihan model seperti pada Gambar 11 yang menunjukkan proses training menggunakan 3.628 gambar *train* yang dilakukan.

Epoch	GPU_mem	box_loss	cls_loss	dfl_loss	Instances	Size	640:100%	227/227	[00:16:00:00, 2.981t/s]
1/40	2.380	1.458	3.379	1.619	44	mAP50	mAP50-95: 1.00%	32/32	[00:12:00:00, 2.531t/s]
	Class	Images	Instances	Box(P)	0	0.16	0.8952		
	all	1801	1891	0.566	0.2				
Epoch	GPU_mem	box_loss	cls_loss	dfl_loss	Instances	Size	640:100%	227/227	[00:12:00:00, 3.131t/s]
2/40	2.410	1.443	2.559	1.584	55	mAP50	mAP50-95: 1.00%	32/32	[00:08:00:00, 3.921t/s]
	Class	Images	Instances	Box(P)	8	0.237	0.143		
	all	1801	1891	0.655	0.26				
Epoch	GPU_mem	box_loss	cls_loss	dfl_loss	Instances	Size	640:100%	227/227	[00:18:00:00, 3.201t/s]
3/40	2.480	1.441	2.336	1.574	54	mAP50	mAP50-95: 1.00%	32/32	[00:10:00:00, 2.861t/s]
	Class	Images	Instances	Box(P)	8	0.269	0.16		
	all	1801	1891	0.67	0.234				
Epoch	GPU_mem	box_loss	cls_loss	dfl_loss	Instances	Size	640:100%	227/227	[00:18:00:00, 3.221t/s]
4/40	2.340	1.438	2.287	1.565	59	mAP50	mAP50-95: 1.00%	32/32	[00:08:00:00, 3.951t/s]
	Class	Images	Instances	Box(P)	8	0.288	0.171		
	all	1801	1891	0.643	0.28				
Epoch	GPU_mem	box_loss	cls_loss	dfl_loss	Instances	Size	640:100%	227/227	[00:11:00:00, 3.171t/s]
5/40	2.340	1.408	2.029	1.537	58	mAP50	mAP50-95: 1.00%	32/32	[00:11:00:00, 2.821t/s]
	Class	Images	Instances	Box(P)	8	0.341	0.207		
	all	1801	1891	0.746	0.316				
Epoch	GPU_mem	box_loss	cls_loss	dfl_loss	Instances	Size	640:100%	227/227	[00:11:00:00, 3.161t/s]
6/40	2.420	1.395	1.925	1.53	48	mAP50	mAP50-95: 1.00%	32/32	[00:07:00:00, 4.041t/s]
	Class	Images	Instances	Box(P)	8	0.307	0.176		
	all	1801	1891	0.59	0.317				

Gambar 11. Proses Pelatihan Model

Dari hasil training yang dilakukan, didapatkan juga hasil tampilan grafik *confusion matrix* seperti yang dapat dilihat pada Gambar X berikut ini.



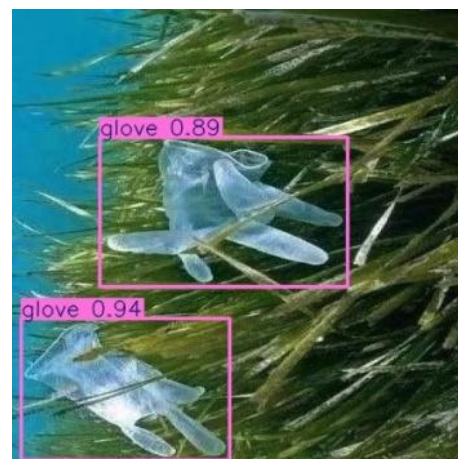
Gambar 12. Confusion Matrix

Setelah dilatih dengan model YOLOV8, maka akan dites oleh gambar sampah di dalam air yang ada pada folder *test* sebanyak 3.628 gambar dan didapatkan hasil *precision*, mAP, dan *recall* untuk masing-masing *class* pada model deteksi sampah di dalam air.

Tabel 3. Evaluasi Model Deteksi Sampah di Dalam Air

No	Class	Precision	mAP	Recall
1	All	0.756	0.741	0.651
2	Mask	0.964	0.852	0.722
3	can	0.804	0.52	0.368
4	cellphone	0.971	0.993	0.978
5	electronics	0.838	0.818	0.737
6	gbottle	0.881	0.67	0.524
7	glove	0.961	0.802	0.722
8	metal	0.168	0.217	0.168
9	misc	0.812	0.581	0.394
10	net	0.881	0.851	0.754
11	pbag	0.804	0.952	0.958
12	pbottle	0.595	0.641	0.642
13	plastic	0.688	0.475	0.3
14	rod	0.335	0.995	1
15	sunglasses	0.737	0.995	1
16	tire	0.896	0.756	0.5

Tabel 3 menunjukkan model telah divalidasi dengan baik, dengan nilai precision 0.756, recall 0.651, mAP 0.741, dan F1 score 0.7. Data tes digunakan untuk memprediksi sampah di air sebagai evaluasi hasil training dan validasi, dengan contoh hasil ditampilkan berikutnya.



Gambar 13. Hasil Prediksi Data Test 1



Gambar 14. Hasil Prediksi Data Test 2



Gambar 15. Hasil Prediksi Data Test 3

Model YOLOv8 berfungsi dengan cara membagi gambar menjadi beberapa kotak pembatas atau yang disebut sebagai *bounding box* untuk mengidentifikasi objek, dalam model ini berbagai jenis sampah tertentu yang ada di dalam air. Gambar 13, 14, dan 15 menunjukkan contoh hasil prediksi pada data *test*, yang didapatkan setelah proses pelatihan dan validasi model ini. Dari hasil klasifikasi yang didapatkan dengan nilai *confidence* yang tinggi seperti pada Gambar 13 yaitu sebesar 89% dan 94%, lalu pada Gambar 14 yaitu sebesar 74%, serta Gambar 15 sebesar 97%. Hal ini menunjukkan bahwa model ini memiliki tingkat kepastian yang cukup baik dalam mendeteksi sampah di dalam air.

5. KESIMPULAN DAN SARAN

Penelitian ini berhasil mengimplementasikan model YOLOv8 untuk mendeteksi berbagai jenis sampah di perairan, seperti plastik, elektronik, dan logam. Model menunjukkan performa baik dengan precision 0.756, mAP 0.741, dan recall 0.651, terutama akurat untuk objek seperti botol, jaring, dan perangkat elektronik, meski masih lemah dalam mendeteksi logam. YOLOv8 terbukti dapat meningkatkan efisiensi identifikasi dan klasifikasi sampah di air. Ke depan, disarankan penggunaan model yang lebih optimal seperti YOLOv11, perluasan variasi dataset, pengoptimalan hyperparameter, dan eksplorasi teknik ensemble. Kolaborasi dengan pihak terkait juga penting untuk penerapan teknologi ini dalam pelestarian lingkungan perairan.

DAFTAR PUSTAKA

- [1] A. Arni and Susilawati, "Pencemaran air sungai akibat pembuangan sampah di desa bagan kuala tanjung beringin Kabupaten Serdang Bedagai," *Nautical : Jurnal Ilmiah Multidisiplin Indonesia*, vol. 1, no. 4, pp. 241–245, 2022, doi: <https://doi.org/10.55904/nautical.v1i4.292>.
- [2] Aulia Ryza Aqilla, A. Razak, Eri Barlian, Nurhasan Syah, and Skunda Diliarosta, "Pengaruh Sampah Plastik Dalam Pencemaran Air," *Gudang Jurnal Multidisiplin Ilmu*, vol. 1, no. 6, pp. 275–280, 2023, doi: <https://doi.org/10.59435/gimi.v1i6.203>.
- [3] B. Tjandra, N. Made, and Handoko, Nyoo S. C., "(PDF) Deteksi Sampah di Permukaan dan Dalam Perairan pada Objek Video dengan Metode Robust and Efficient Post-Processing dan Tubelet-Level Bounding Box Linking," *ResearchGate*, 2023, doi: <https://doi.org/10.48550/arXiv.2307.10039>.
- [4] Tresya Anjali Dompeipen, Sherwin R.U.A. Sompie, and Meicsy E.I. Najoran, "Computer Vision Implementation for Detection and Counting the Number of Humans," *Jurnal Teknik Informatika*, vol. 16, no. 1, pp. 65–76, 2021, doi: <https://doi.org/10.35793/jti.16.1.2021.31471>.
- [5] Dzaky Nafis Alfarizi, Rio Agung Pangestu, Dimas Aditya, M. A. Setiawan, and Perani Rosyani, "Penggunaan Metode YOLO Pada Deteksi Objek: Sebuah Tinjauan Literatur Sistematis," *AI dan SPK : Jurnal Artificial Intelligent dan Sistem Penunjang Keputusan*, vol. 1, no. 1, pp. 54–63, 2023, Available: <https://jurnalmahasiswa.com/index.php/aidansp/article/view/144>.
- [6] M. Sohan, Thotakura Sai Ram, and V. Rami, "A Review on YOLOv8 and Its Advancements," *ResearchGate*, Jan. 07, 2024, https://www.researchgate.net/publication/377216968_A_Review_on_YOLOv8_and_Its_Advan_cements.
- [7] Imam Maulana, Nining Rahaningsih, and Tati Suprpti, "Analisis Penggunaan Model YOLOv8 (You Only Look Once) Terhadap Deteksi Citra Senjata Berbahaya," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 7, no. 6, pp. 3621–3627, Feb. 2024, doi: <https://doi.org/10.36040/jati.v7i6.8271>.
- [8] Y. Yanto, F. Aziz, and I. Irmawati, "YOLO-V8 PENINGKATAN ALGORITMA UNTUK DETEKSI PEMAKAIAN MASKER WAJAH," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 7, no. 3, pp. 1437–1444, Oct. 2023, doi: <https://doi.org/10.36040/jati.v7i3.7047>.
- [9] Y. T. Wulandari, T. Rosandy, "Implementasi Computer Vision Dalam Sistem Deteksi Gerakan Disiplin Kampus," *Jurnal TEKNIKA*, vol. 18(2), pp. 343–354, 2024.
- [10] R. Andarsyah and Amri Yanuar, "Sentimen Analisis Aplikasi Posaja Pada Google Playstore Untuk Peningkatan Pospay Superapp Menggunakan Support Vector Meachine," *Jurnal Teknik Informatika*, vol. 16, no. 2, pp. 1–7, 2024, Accessed: Nov. 20, 2024. [Online]. Available: <https://ejurnal.ulbi.ac.id/index.php/informatika/article/view/3533#>.
- [11] M. R. S. Alfarizi, M. Z. . Al-farish, M. Taufiqurrahman, G. Ardiansah, and M. . Elgar, "Penggunaan Python Sebagai Bahasa Pemrograman untuk Machine Learning dan Deep

- Learning”, *karimahtauhid*, vol. 2, no. 1, pp. 1–6, Jan. 2023.
- [12] H. Rohayani, S. Alam Nur, M. Fauzi, and Rico, “Prediksi Penyebaran Virus COVID-19 Dari Hasil PCR Menggunakan Metode Naïve Bayes,” *J. Comput. Syst. Informatics*, vol. 4, no. 1, pp. 109–115, 2022.
- [13] Moh. Baha’Uddin and Zaehol Fatah, “Penerapan Data Mining Clustering K-Means Dalam Mengelompokkan Data Penduduk Penyandang Disabilitas,” *Gudang Jurnal Multidisiplin Ilmu*, vol. 2, no. 11, pp. 46–94, 2024, doi: <https://doi.org/10.59435/gjmi.v2i11.1040>.
- [14] Arnav Samal, “Underwater Plastic Pollution Detection,” Kaggle.com, 2024. <https://www.kaggle.com/datasets/arnavs19/underwater-plastic-pollution-detection>
- [15] Roboflow, “Preprocess Images | Roboflow Docs,” Roboflow.com, May 23, 2024. <https://docs.roboflow.com/datasets/image-preprocessing> (accessed Nov. 23, 2024).
- [16] H. Nurhadiati, S. A. Wibowo, A. Pratondo, “Analisis Performansi Deteksi Objek Pada Metode *Complex YOLOv4* Untuk *Autonomous Driving*,” *e-Prossiding of Engineering*, pp. 2753–2758, 2022.