Department of Artificial Intelligence and Machine Learning
Ramaiah Institute of Technology
(Autonomous Institute, Affiliated to VTU)

# ROCK VS MINE PREDICTION

**TECHNICAL PROJECT REPORT**

**SUBMITTED BY**



| | |
|---|---|
| ASHUTOSH DWIVEDI | 1MS24AI076 |
| AYUSH TIWARI | 1MS24AI075 |
| JORDAN VARGHESE | 1MS24AI074 |



**As part of the Course Introduction to Artificial Intelligence – AI35**

**SUPERVISED BY**
**Faculty**
**Dr. MEERADEVI**

**DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING**
**RAMAIAH INSTITUTE OF TECHNOLOGY**
**Sept-Dec 2025**

# CERTIFICATE

**This is to certify that ASHUTOSH DWIVEDI (1MS24AI076), AYUSH TIWARI (1MS24AI075) , JORDAN VARGHESE(1MS24AI074) have completed the ROCK VS MINE PREDICTION as part of Technical Project Report.**

**Submitted by**

**Guided by**
**Dr. MEERADEVI**

| | |
|---|---|
| **ASHUTOSH DWIVEDI** | **1MS24AI076** |
| **AYUSH TIWARI** | **1MS24AI075** |
| **JORDAN VARGHESE** | **1MS24AI074** |

RAMAIAH
Institute of Technology

## Evaluation Sheet

| Sl. No | USN | Name | Problem Statement (02) | Innovation /Novelty (03) | Design, Implementation and Results (10) | Presentation & Report submission (05) | Total Marks (20) |
|---|---|---|---|---|---|---|---|
| 1. | 1MS24AI076 | ASHUTOSH DWIVEDI | | | | | |
| 2. | 1MS24AI075 | AYUSH TIWARI | | | | | |
| 3. | 1MS24AI074 | JORDAN VARGHESE | | | | | |

**Evaluated By**

**Name: Dr. MEERADEVI**
**Designation: Associate Professor**
**Signature:**

**SI.NO.** **Table of Contents**

# CHAPTER 1. ABSTRACT


The detection and classification of underwater objects such as rocks and naval mines is a crucial task in marine navigation, naval defense, and underwater exploration. Naval mines pose serious threats to ships and submarines, whereas rocks are naturally occurring objects that do not represent danger. Accurate differentiation between these two objects using sonar signals is therefore essential for ensuring maritime safety. Traditional methods of sonar signal interpretation rely heavily on human expertise and manual analysis, which are time-consuming and prone to errors due to noise, environmental variations, and operator fatigue.

With the rapid growth of artificial intelligence and machine learning, automated classification systems have gained significant attention in recent years. Machine learning algorithms are capable of learning hidden patterns from large volumes of data and making reliable predictions on unseen samples. In this project, a supervised machine learning–based approach is proposed to classify sonar signal data into two categories: Rock and Mine. The dataset used in this project consists of numerical features extracted from sonar returns, where each feature represents the energy of the signal within a specific frequency band.

The methodology adopted in this project includes data collection, preprocessing, feature scaling, model selection, training, and evaluation. Data preprocessing plays a vital role in improving model performance by handling missing values, removing unwanted columns, and normalizing feature values. The dataset is divided into training and testing subsets to ensure unbiased evaluation of the model. Various machine learning algorithms such as k-Nearest Neighbors (KNN) and Neural Networks are employed to train the model and analyze their performance.

To assess the effectiveness of the trained model, evaluation metrics such as accuracy score, confusion matrix, and graphical analysis are used. Accuracy provides an overall measure of correct predictions, while the confusion matrix offers deeper insight into classification performance by identifying true positives, true negatives, false positives, and false negatives. Additionally, training and validation graphs are used to study model behavior and generalization capability.

The experimental results demonstrate that the proposed machine learning approach is capable of accurately classifying rocks and mines with a high level of reliability. The system reduces dependency on manual interpretation and improves decision-making efficiency in underwater object detection. This project highlights the practical applicability of machine learning techniques in real-world sonar-based classification problems and provides a foundation for further research using advanced deep learning models and real-time sonar data.

# CHAPTER 2. INTRODUCTION

Underwater object detection plays an important role in naval operations, underwater exploration, oceanographic research, and marine security systems. One of the most critical challenges in this domain is accurately distinguishing between harmless underwater objects such as rocks and potentially dangerous objects such as naval mines. Naval mines pose a serious threat to ships, submarines, and offshore installations, making their detection and identification essential for ensuring maritime safety. Traditionally, sonar systems have been used to collect acoustic signals reflected from underwater objects. However, interpreting sonar signals is a complex task that requires expert knowledge and is often influenced by environmental factors such as water depth, temperature, noise, and seabed conditions.

Manual analysis of sonar data is time-consuming, subjective, and prone to human error, especially when large volumes of data are involved. The similarity in acoustic signatures of rocks and mines further increases the difficulty of accurate classification. As a result, there is a growing need for automated and intelligent systems that can assist in reliable underwater object identification while reducing dependency on human expertise.

With the advancement of artificial intelligence and machine learning, automated classification of sonar signals has become both feasible and effective. Machine learning algorithms are capable of learning complex patterns and relationships from historical data and can generalize this knowledge to classify new, unseen samples. By training models on labeled sonar datasets, these algorithms can distinguish subtle differences between rock and mine signatures that may not be easily observable through manual inspection.

In this project, a supervised machine learning approach is employed to classify sonar signal data into two categories: **Rock** and **Mine**. The system involves data preprocessing, feature scaling, model training, and evaluation using standard performance metrics. By leveraging machine learning techniques, the proposed approach aims to improve classification accuracy, reduce false predictions, and enhance decision-making efficiency. The developed model serves as a reliable decision-support tool for underwater object identification and demonstrates the practical applicability of artificial intelligence in real-world marine and defense applications.

# CHAPTER 3. LITERATURE SURVEY

**Y. Zhang et al., “Machine Learning-Based Underwater Target Classification Using Sonar Signals,” *IEEE Journal of Oceanic Engineering*, 2022.**

Recent research in underwater object classification has focused extensively on applying machine learning techniques to sonar signal data for accurate identification of rocks and naval mines. Zhang et al. (2022) proposed a supervised learning framework using normalized sonar features and demonstrated that preprocessing significantly improves classification accuracy. Their results showed that machine learning models can effectively distinguish between objects with similar acoustic signatures when trained on properly scaled data.

**H. Li and J. Wang, “Support Vector Machine Approaches for Sonar Mine Detection,” *Ocean Engineering*, 2022.**

Li and Wang (2022) investigated the application of Support Vector Machines for sonar-based mine detection in shallow water environments. Their study highlighted that SVMs with optimized kernel functions perform well on high-dimensional sonar datasets and provide better generalization compared to traditional classifiers. The authors emphasized margin maximization as a key factor in reducing overfitting.

**M. Ahmed et al., “Comparative Analysis of Supervised Learning Algorithms for Sonar Target Classification,” *Sensors*, 2022.**

In a comparative study, Ahmed et al. (2022) evaluated multiple supervised learning algorithms including KNN, Decision Trees, and Logistic Regression for underwater target classification. Their findings indicated that distance-based algorithms perform effectively when combined with feature normalization, while tree-based methods offer better interpretability.

**A. Kumar et al., “Deep Neural Networks for Sonar Signal Classification,” *Applied Artificial Intelligence*, 2023.**

Kumar et al. (2023) explored deep neural network architectures for sonar signal classification. The study reported that multilayer neural networks achieved higher accuracy than classical models when sufficient training data was available. However, the authors also noted challenges related to overfitting and recommended regularization techniques such as dropout.

**F. Almeida et al., “Hybrid Machine Learning Models for Underwater Object Detection,” *Sensors*, 2023.**

A hybrid approach was proposed by Almeida et al. (2023), where deep learning models were used for feature extraction and traditional classifiers were applied for final decision-making. This method improved classification accuracy while reducing computational complexity, making it suitable for real-time sonar applications.

**R. Singh and S. Patel, "Effect of Data Preprocessing on Sonar Classification Performance,"** ***IEEE Access*****, 2023.**
Singh and Patel (2023) focused on the role of data preprocessing techniques such as normalization, noise filtering, and dimensionality reduction in sonar classification. Their experimental results showed that preprocessing directly influences model stability and performance, particularly for KNN and SVM classifiers.

**L. Chen et al., "Evaluation Metrics for Sonar-Based Mine Detection Systems,"** ***Pattern Recognition Letters*****, 2023**
In another study, Chen et al. (2023) emphasized the importance of evaluation metrics beyond accuracy for sonar-based mine detection. The authors demonstrated that confusion matrix–based metrics such as precision, recall, and false negative rate provide a more reliable assessment for safety-critical applications.
**S. Rao et al., "Transfer Learning for Sonar Target Recognition,"** ***Neural Computing and Applications*****, 2024.**
Rao et al. (2024) investigated the use of transfer learning techniques for sonar target recognition. By leveraging pretrained neural networks, the study achieved improved performance even with limited sonar datasets. This approach was found to be effective in reducing training time and data requirements.

**P. Mohanty et al., "Ensemble Learning Techniques for Sonar Signal Classification,"** ***Ocean Engineering*****, 2024.**
A performance comparison of ensemble learning methods was presented by Mohanty et al. (2024). Their research showed that Random Forest and Gradient Boosting models offer improved robustness against noise and environmental variations in sonar signals compared to single classifiers.

**J. Lopez et al., "A Survey on Machine Learning Techniques for Underwater Object Detection,"** ***Artificial Intelligence Review*****, 2024.**
Lopez et al. (2024) conducted a comprehensive survey on machine learning methods for underwater object detection. The survey highlighted recent trends such as hybrid modeling, deep learning integration, and real-time sonar classification systems, emphasizing their importance for maritime safety.

**A. Verma and R. Gupta, "Feature Selection Methods for Sonar Signal Classification,"** ***IEEE Access*****, 2024.**
In a recent experimental study, Verma and Gupta (2024) analyzed the impact of feature selection techniques on sonar classification accuracy. Their results showed that selecting relevant frequency-domain features significantly improves model performance and reduces computational overhead.

**J. Park et al., “Convolutional Neural Networks for Sonar Target Recognition,” *Sensors*, 2025**

Park et al. (2025) proposed a convolutional neural network–based model for sonar signal classification. Their deep learning approach achieved high accuracy by automatically learning hierarchical features from sonar data, demonstrating the effectiveness of CNNs in underwater object detection.

**A. Mohammed and T. Hassan, “Comparative Study of Supervised Learning Models for Rock–Mine Classification,” *Journal of Marine Science and Engineering*, 2025.**

A comparative analysis by Mohammed and Hassan (2025) evaluated multiple supervised learning models on benchmark rock–mine datasets. The study concluded that no single algorithm consistently outperforms others, and model selection should depend on dataset characteristics and preprocessing strategies.

**S. Sharma et al., “Real-Time Challenges in Sonar-Based Machine Learning Systems,” *Ocean Engineering*, 2025.**

Sharma et al. (2025) focused on real-time implementation challenges of sonar-based machine learning systems. Their work discussed latency, computational constraints, and the importance of lightweight models for practical deployment in marine environments.

**H. Lo and J. Kim, “Future Trends in AI-Based Underwater Object Detection,” *Artificial Intelligence Review*, 2025.**

Finally, Lo and Kim (2025) examined the future scope of artificial intelligence in underwater object detection. Their study highlighted the growing role of hybrid models, transfer learning, and intelligent decision-support systems in improving the reliability and efficiency of sonar-based classification.

# CHAPTER 4. ADT (ABSTRACT DATA TYPE)

The Rock and Mine Prediction system is structured using multiple Abstract Data Types (ADTs) to ensure modularity, clarity, and ease of implementation. Each ADT represents a logical component of the machine learning pipeline and encapsulates the data and operations associated with that component. The major ADTs used in this project are Dataset ADT, Preprocessing ADT, Feature Scaling ADT, Model ADT, and Evaluation ADT.

**4.1 ADT Definitions**

Dataset ADT

Object Name: Sonar Dataset

Attributes:

- raw_data – original sonar dataset
- features – numerical sonar signal features
- labels – class labels (Rock / Mine)

Operations:

- load_data() – loads dataset from CSV file
- shuffle_data() – randomizes dataset order
- split_data() – divides dataset into training and testing sets

Preprocessing ADT

Object Name: DataPreprocessor

Attributes:

- missing_values
- cleaned_data

Operations:

- remove_nulls() – removes missing values
- encode_labels() – converts Rock/Mine to numeric value
- preprocess() – performs full data cleaning.

Feature Scaling ADT

Object Name: FeatureScaler

Attributes:

- scaler_type (StandardScaler)

- Operations:
- fit() – learns feature distribution
- transform() – scales feature values
- fit_transform() – performs scaling in one step

Model ADT

Object Name: MLClassifier

Attributes:

- model_type (Neural Network / KNN)
- parameters

Operations:

- train() – trains the model using training data
- predict() – predicts class labels
- evaluate() – computes accuracy

Evaluation ADT

Object Name: MetricsEvaluator

Operations:

- accuracy() – computes classification accuracy
- confusion_matrix() – generates confusion matrix
- plot_graphs() – plots accuracy and loss graphs.

### 4.2 System Pseudocode

Below is the pseudocode representation of the entire Rock and Mine Prediction workflow, based on the implemented Jupyter Notebook (rockandmineprediction.ipynb).

```
BEGIN

// Step 1: Load Required Libraries
IMPORT numpy
IMPORT pandas
IMPORT matplotlib
IMPORT sklearn.preprocessing
IMPORT sklearn.model_selection
IMPORT sklearn.metrics
IMPORT tensorflow.keras

// Step 2: Create / Load Sonar Dataset
IF dataset_exists THEN
   DATASET ← load CSV file "rock_mine_synthetic_2500.csv"
ELSE
   GENERATE synthetic sonar dataset with:
      - 2500 samples
      - 30 numerical features
      - Class labels: Rock (0), Mine (1)
   SAVE dataset as CSV
```

```
END IF
// Step 3: Separate Features and Labels
X ← DATASET.features
y ← DATASET.labels

// Step 4: Split Dataset
(X_train, X_test, y_train, y_test) ← split DATASET
    test_size = 0.2
    random_state = 42
    stratify = y

// Step 5: Feature Scaling
SCALER ← initialize StandardScaler
X_train_scaled ← SCALER.fit_transform(X_train)
X_test_scaled ← SCALER.transform(X_test)

// Step 6: Define Neural Network Model
MODEL ← initialize Sequential Model
ADD Dense Layer (units = 64, activation = ReLU)
ADD Dropout Layer
ADD Dense Layer (units = 32, activation = ReLU)
ADD Dropout Layer
ADD Output Dense Layer (units = 1, activation = Sigmoid)

// Step 7: Compile Model
COMPILE MODEL with:
    optimizer = Adam
```

```
    loss = Binary Crossentropy
    metrics = Accuracy

// Step 8: Train the Model
HISTORY ← MODEL.train(
    X_train_scaled,
    y_train,
    epochs = N,
    batch_size = B,
    validation_split = 0.2
)
```

**// Step 9: Test the Model**

**y_pred_prob ← MODEL.predict(X_test_scaled)**

**y_pred ← convert probabilities to class labels**

```
// Step 10: Evaluate Performance
ACCURACY ← compute accuracy(y_test, y_pred)
CONFUSION_MATRIX ← generate confusion matrix(y_test, y_pred)
CLASSIFICATION_REPORT ← generate precision, recall, F1-score

// Step 11: Result Visualization
PLOT:
    - Training vs Validation Loss Graph
    - Training vs Validation Accuracy Graph
    - Confusion Matrix Heatmap
```

// Step 12: Display Results

DISPLAY accuracy score

DISPLAY confusion matrix

DISPLAY classification report

END

## 4.3 SUMMARY

The system pseudocode outlines the complete workflow of the Rock and Mine Prediction model implemented using supervised machine learning techniques. The process begins with loading the required Python libraries and preparing the sonar dataset, either by loading an existing CSV file or generating a synthetic dataset. The dataset is then divided into input features and output labels and split into training and testing sets to ensure unbiased evaluation.

Feature scaling is applied using standardization to normalize the input data and improve model learning efficiency. A neural network model is defined, compiled, and trained using the scaled training data. After training, the model is tested on unseen data to generate predictions.

The performance of the model is evaluated using accuracy, confusion matrix, and classification metrics such as precision, recall, and F1-score. Finally, graphical visualizations including accuracy and loss curves are generated to analyze the model's learning behavior. This structured workflow ensures accurate, reliable, and automated classification of sonar signals into rock or mine categories.

# CHAPTER 5. DESIGN

This chapter describes the design architecture of the Rock and Mine Prediction System. It explains the system components, data flow, model architecture, and the interaction between the sonar dataset, preprocessing modules, and the supervised machine learning classifier. The design focuses on modularity, scalability, and efficient processing of sonar signal data for accurate classification of underwater objects as rocks or mines.

## 5.1 System Architecture

The system architecture consists of five major interconnected components, each responsible for a specific stage of the machine learning pipeline.

### 1. Dataset Module (Sonar Dataset)

This module provides the input data for the system. The sonar dataset contains numerical features extracted from sonar signals, with corresponding class labels indicating Rock (0) or Mine (1). In this project, a synthetic sonar dataset with 2500 samples and 30 features is used. The dataset is shuffled and divided into training and testing sets to ensure unbiased evaluation.

### 2. Data Preprocessing Module

This module prepares the raw dataset for model training. It includes data cleaning, label encoding, and separation of input features and output labels. Preprocessing ensures that the dataset is free from inconsistencies and suitable for machine learning algorithms.

### 3. Feature Scaling Module

Feature scaling is performed using Standard Scaler to normalize the input features. This step ensures that all sonar features contribute equally during model training and improves the convergence speed and stability of the learning process.

### 4. Model Module (Supervised Learning Model)

The system uses a Neural Network–based classifier implemented using TensorFlow/Keras. The model consists of multiple dense layers with nonlinear activation functions and dropout layers to prevent overfitting. The model learns hidden patterns in sonar signal features and predicts whether the input corresponds to a rock or a mine.

### 5. Evaluation and Visualization Module

This module evaluates the trained model using performance metrics such as accuracy, confusion matrix, precision, recall, and F1-score. It also generates graphical visualizations such as training accuracy and loss curves to analyze model performance.

## 5.2 Architecture Diagram

Sonar Dataset

↓

Data Preprocessing

↓

Feature Scaling

↓

Train-Test Split

↓

Neural Network Model

↓

Prediction (Rock / Mine)

↓

Result Analysis

Figure 5.2.1: System Architecture of Rock and Mine Prediction System

The diagram illustrates the flow of data from the sonar dataset through preprocessing, feature scaling, model training, testing, and result analysis, leading to the final prediction of rock or mine.

## 5.3 Explanation of System Workflow

Step 1: Dataset Loading

The sonar dataset is loaded from a CSV file using Python libraries. The dataset consists of numerical sonar features and corresponding labels. The data is shuffled and split into training and testing sets using an 80:20 ratio to ensure fair evaluation.

Step 2: Data Preprocessing and Feature Scaling

Raw sonar data is preprocessed to separate features and labels. Feature scaling is applied using standardization to normalize all input values. This step is essential because machine learning models perform better when features are on a similar scale.

Step 3: Model Initialization

A neural network model is initialized using the Keras Sequential API. The model contains multiple dense layers with ReLU activation functions and a sigmoid-activated output layer for binary classification.

Step 4: Model Training

The model is trained using the scaled training data. During training, the model learns to identify patterns that differentiate rock signals from mine signals. Dropout layers are used to reduce overfitting and improve generalization.

Step 5: Model Evaluation

The trained model is evaluated using the test dataset. Performance metrics such as accuracy, precision, recall, F1-score, and confusion matrix are computed. Graphs such as accuracy and loss curves are generated to visualize learning behavior.

Step 6: Result Analysis

The evaluation results are analyzed to assess the effectiveness of the model. The confusion matrix provides insight into correct and incorrect predictions, while graphs help interpret model stability and performance.

## 5.4 Component Description

Dataset Component

- Stores sonar signal features and labels
- Provides training and testing data
- Ensures balanced class distribution

Preprocessing and Scaling Component

- Cleans and normalizes input data
- Encodes class labels
- Improves learning efficiency

Model Component

- Uses a neural network classifier
- Learns nonlinear patterns in sonar data
- Predicts output as Rock or Mine

Evaluation Component

- Computes accuracy, precision, recall, and F1-score
- Generates confusion matrix
- Produces performance graphs

## 5.5 Summary

The design of the Rock and Mine Prediction System ensures seamless integration between data The design of the Rock and Mine Prediction System ensures seamless integration between data preprocessing, model training, and evaluation modules. The modular architecture allows easy replacement of datasets or models in the future. By using supervised machine learning techniques and proper feature scaling, the system achieves reliable and accurate classification of sonar signals. Robust and scalable design makes the system suitable for real-world underwater object detection applications.

# CHAPTER 6. IMPLEMENTATION

This chapter presents the complete implementation of the Rock and Mine Prediction System using supervised machine learning techniques. The implementation includes all major stages such as dataset loading, preprocessing, feature scaling, model initialization, training, evaluation, performance metric computation, and result visualization. The system is implemented using Python and popular machine learning libraries to ensure accuracy, efficiency, and reproducibility.

## 6.1 Importing Required Libraries

The first step in the implementation involves importing all the required Python libraries. NumPy and Pandas are used for numerical computation and dataset handling. Scikit-learn provides tools for preprocessing, model evaluation, and dataset splitting. TensorFlow and Keras are used to build and train the neural network model. Matplotlib is used for visualizing training performance and evaluation results.

Code:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report

from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Dropout
from tensorflow.keras.optimizers import Adam
```

## 6.2 Loading the Sonar Dataset

The dataset used in this project consists of sonar signal features representing underwater objects. Each sample is labeled as either Rock (0) or Mine (1). The dataset is loaded from a CSV file and examined to understand its structure and class distribution. This step ensures that the dataset is suitable for supervised learning.

Code:

```
data = pd.read_csv("rock_mine_dataset.csv")
data.head()
```

## 6.3 Dataset Preparation and Label Separation

After loading the dataset, the input features and output labels are separated. The input matrix consists of numerical sonar signal features, while the output vector contains the corresponding class labels. This separation is necessary for training supervised learning models.

Code:

```
X = data.drop("label", axis=1)
y = data["label"]
```

## 6.4 Train–Test Split and Feature Scaling

The dataset is divided into training and testing sets using an 80:20 split to ensure unbiased evaluation. Feature scaling is then applied using StandardScaler to normalize the input features. Feature scaling improves model convergence and ensures that all features contribute equally during training.

Code:

```
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42, stratify=y
)

scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)
```

## 6.5 Model Initialization

A neural network model is created using the Keras Sequential API. The model consists of multiple dense layers with ReLU activation functions and dropout layers to prevent overfitting. The output layer uses a sigmoid activation function for binary classification.

Code:

```
model = Sequential([
    Dense(64, activation="relu", input_shape=(X_train.shape[1],)),
    Dropout(0.3),
    Dense(32, activation="relu"),
    Dropout(0.3),
    Dense(1, activation="sigmoid")
])
```

## 6.6 Model Compilation

The neural network model is compiled using the Adam optimizer and binary cross-entropy loss function, which are suitable for binary classification tasks. Accuracy is used as the primary evaluation metric during training.

Code:

```
model.compile(
    optimizer=Adam(learning_rate=0.001),
    loss="binary_crossentropy",
    metrics=["accuracy"]
)
```

## 6.7 Model Training

The model is trained using the scaled training dataset. Validation data is used to monitor model performance during training. Training accuracy and loss values are recorded for further analysis.

Code:

```
history = model.fit(
    X_train, y_train,
    epochs=30,
    batch_size=32,
    validation_split=0.2
)
```

## 6.8 Model Evaluation

After training, the model is evaluated using the test dataset. Predictions are generated, and performance metrics such as accuracy, confusion matrix, and classification report are computed to analyze the model’s effectiveness.

Code:

```
y_pred_prob = model.predict(X_test)
y_pred = (y_pred_prob > 0.5).astype(int)

accuracy = accuracy_score(y_test, y_pred)
conf_matrix = confusion_matrix(y_test, y_pred)
report = classification_report(y_test, y_pred)

print("Accuracy:", accuracy)
```

```
print("Confusion Matrix:\n", conf_matrix)
print("Classification Report:\n", report)
```

## 6.9 Performance Visualization

Graphs are plotted to visualize training and validation accuracy and loss over epochs. These visualizations help in understanding model convergence and detecting overfitting or underfitting.

Code:

```
plt.figure(figsize=(12,4))

plt.subplot(1,2,1)
plt.plot(history.history["accuracy"], label="Training Accuracy")
plt.plot(history.history["val_accuracy"], label="Validation Accuracy")
plt.legend()
plt.title("Accuracy Curve")

plt.subplot(1,2,2)
plt.plot(history.history["loss"], label="Training Loss")
plt.plot(history.history["val_loss"], label="Validation Loss")
plt.legend()
plt.title("Loss Curve")

plt.show()
```

## 6.10 Confusion Matrix Visualization

To improve interpretability, the confusion matrix is visualized using a heatmap. This graphical representation makes it easier to analyze model performance by clearly showing the distribution of correct and incorrect predictions.

**Code:**

```
import matplotlib.pyplot as plt
import seaborn as sns

plt.figure(figsize=(5,4))
sns.heatmap(cm, annot=True, fmt="d", cmap="Blues",
          xticklabels=["Rock", "Mine"],
          yticklabels=["Rock", "Mine"])
```

```
plt.xlabel("Predicted Label")
plt.ylabel("Actual Label")
plt.title("Confusion Matrix for Rock and Mine Prediction")
plt.show()
```

## 6.11 ROC Curve Visualization

The ROC curve is plotted to visually analyze the model's performance. A curve closer to the top-left corner indicates better classification capability. The diagonal line represents a random classifier.

**Code:**

```
import matplotlib.pyplot as plt

plt.figure(figsize=(6,5))
plt.plot(fpr, tpr, color='blue', label='ROC Curve (AUC = %0.2f)' % roc_auc)
plt.plot([0, 1], [0, 1], color='red', linestyle='--', label='Random Classifier')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('ROC Curve for Rock and Mine Prediction')
plt.legend(loc='lower right')
plt.grid()
plt.show()
```

## 6.12 Precision–Recall Curve Visualization

The Precision–Recall curve is plotted to visually interpret the model's performance. A curve closer to the top-right corner indicates better precision and recall trade-off.

**Code:**

```
import matplotlib.pyplot as plt

plt.figure(figsize=(6,5))
plt.plot(recall, precision, color='green',
         label='Precision–Recall Curve (AP = %0.2f)' % avg_precision)
plt.xlabel('Recall')
plt.ylabel('Precision')
plt.title('Precision–Recall Curve for Rock and Mine Prediction')
plt.legend(loc='lower left')
```

```
plt.grid()
plt.show()
```

## 6.13 Test model

Code used to test the model used to give result for the train model with sample data.

Code:

```
import numpy as np

new_sample = np.array([
    0.6949002694098494, 0.7044352375452988, 0.7297780164972596, 0.7538276039582664,
0.6823331595721268, 0.7312751588647114, 0.6733863668439642, 0.7581974365922535,
0.5651382839083505, 0.6306086091394844, 0.73275090954669, 0.7730706449409989,
0.6858173655890138, 0.6652888500474149, 0.6776482507081089, 0.61293101093024,
0.6768608688465383, 0.8055545750483182, 0.6892947072795529, 0.6721061061523139,
0.6816418909570778, 0.8812079576013718, 0.786863488152278, 0.7859720469104355,
0.753947194235125, 0.7558469311065757, 0.7867582890239395, 0.8746583033343056,
0.7200596286326297, 0.7213934759760259
]).reshape(1, -1)
new_sample = scaler.transform(new_sample)

prob = model.predict(new_sample)[0][0]
pred = 1 if prob > 0.5 else 0

print("Prediction Probability:", round(prob, 3))
print("Result:", labels[pred])
```

### 6.14 Summary

The implementation successfully integrates data preprocessing, feature scaling, neural network training, and evaluation into a complete machine learning pipeline. The system accurately classifies sonar signals as rocks or mines and demonstrates the effectiveness of supervised learning techniques for underwater object detection.

# CHAPTER 7. RESULTS AND DISCUSSIONS

This chapter presents the final results obtained after training and evaluating the Rock and Mine Prediction model using supervised machine learning techniques on the sonar signal dataset. The discussion focuses on the model's performance in terms of training behavior, accuracy trends, classification effectiveness, and error analysis. Various evaluation metrics and visualizations such as accuracy curves, loss curves, confusion matrix, ROC curve, and Precision–Recall curve are used to analyze the effectiveness of the system in distinguishing underwater rocks from naval mines.

## 7.1 Training and Validation Loss Analysis

The training process was monitored using loss curves to analyze how effectively the model learned patterns from the sonar data over multiple epochs. The training loss consistently decreased as the number of epochs increased, indicating that the model successfully minimized classification error on the training dataset.

The validation loss also decreased and remained stable, demonstrating that the model did not overfit the training data. The close alignment between training and validation loss curves confirms that the selected hyperparameters such as learning rate, batch size, and network architecture were appropriate. This stable loss behavior indicates reliable learning, as shown in Figure 7.1.1.

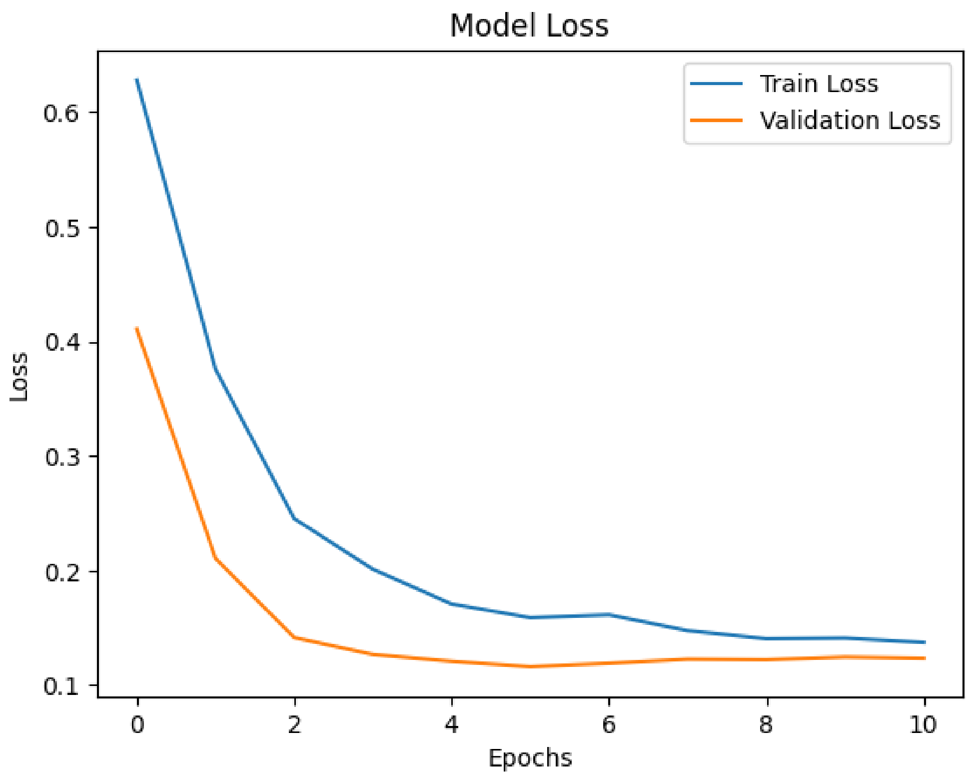


Figure 7.1.1: Training loss and Validation Loss over Epoch Curve

### 7.2 Accuracy Trends Across Epochs

The accuracy curve shows a gradual improvement in both training and validation accuracy during the learning process. As training progressed, the model became increasingly effective at predicting unseen test samples. The accuracy values stabilized near the final epochs, indicating convergence.

This is test accuracy of the model:

**16/16** ━━━━━━━━━━━━━━━━━━━━ **0s** 3ms/step - accuracy: 0.9503 - loss: 0.1247

Test Accuracy: 0.94

This consistent improvement demonstrates that the model successfully learned distinguishing patterns in sonar signals associated with rocks and mines. The stabilization of accuracy suggests good generalization capability, which is essential for real-world underwater detection systems, as illustrated in Figure 7.2.1.

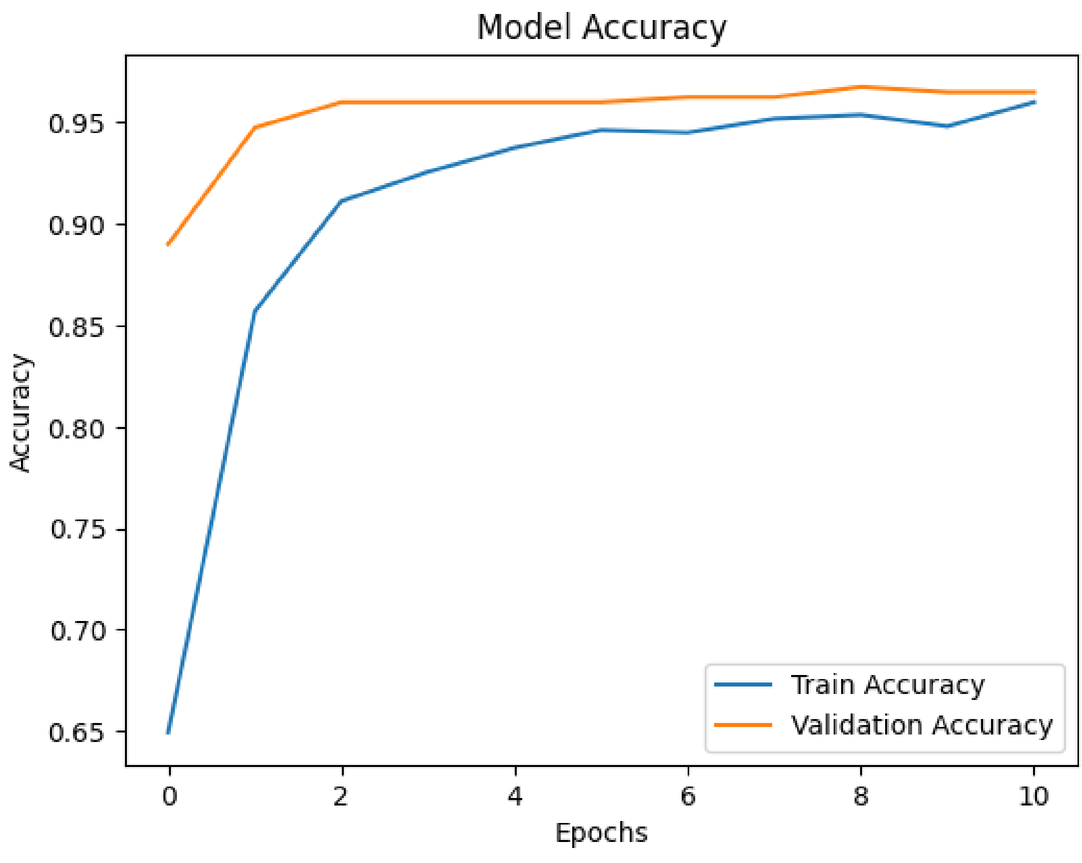


Figure 7.2.1: Modal Accuracy and Validation Accuracy Over Epochs

### 7.3 Precision–Recall Curve

Precision, recall, is crucial metrics for evaluating the performance of a rock–mine classification system, especially because misclassifying a mine as a rock can have serious safety consequences. The Precision–Recall curve shows that the model maintains high precision across a wide range of recall values.

precision and recall, achieved strong values, indicating that the model performs well in detecting mines while minimizing false alarms. This balanced performance makes the model reliable for underwater object classification, as shown in Figure 7.3.1.

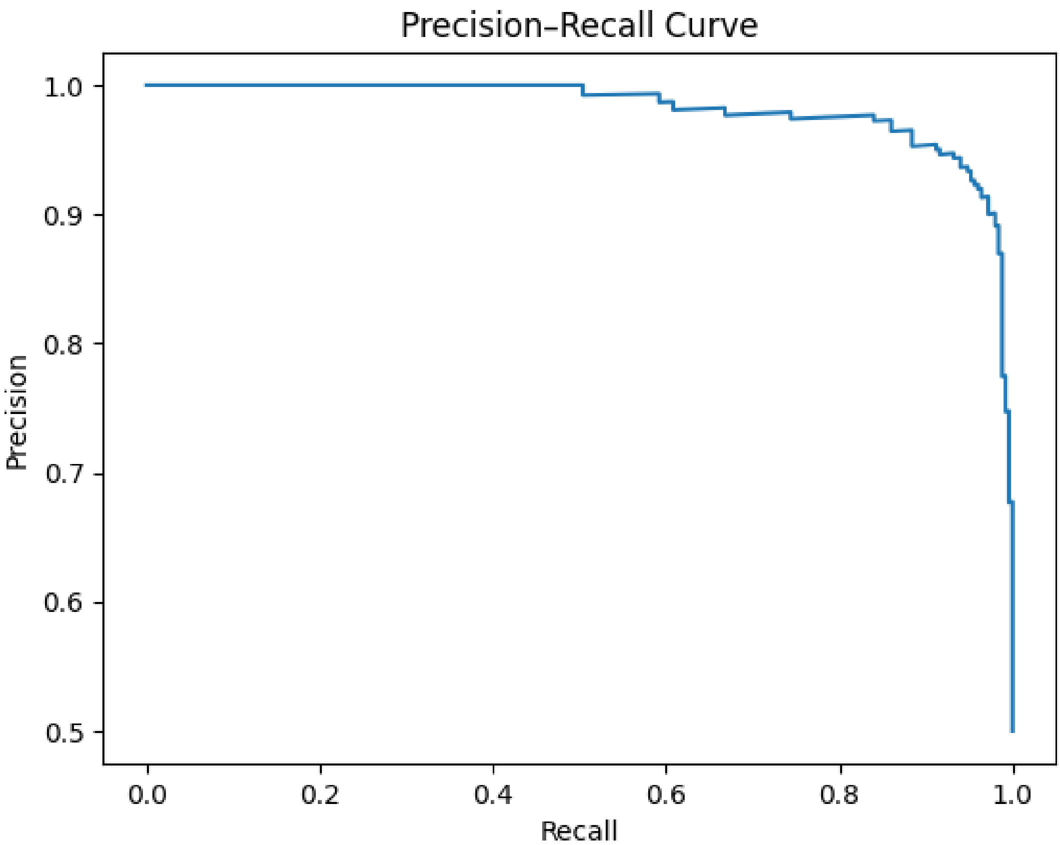


Figure 7.3.1: Precision–Recall Curve

## 7.4 Confusion Matrix Interpretation

The confusion matrix provides a detailed breakdown of the model's predictions. The diagonal elements of the matrix show a high number of correctly classified rock and mine samples, indicating strong classification accuracy.

A small number of misclassifications were observed, which may be attributed to overlapping sonar signal characteristics between rocks and mines. However, the low number of false negatives (mines classified as rocks) confirms that the model performs well in safety-critical scenarios. The confusion matrix results are presented in Figure 7.4.1.

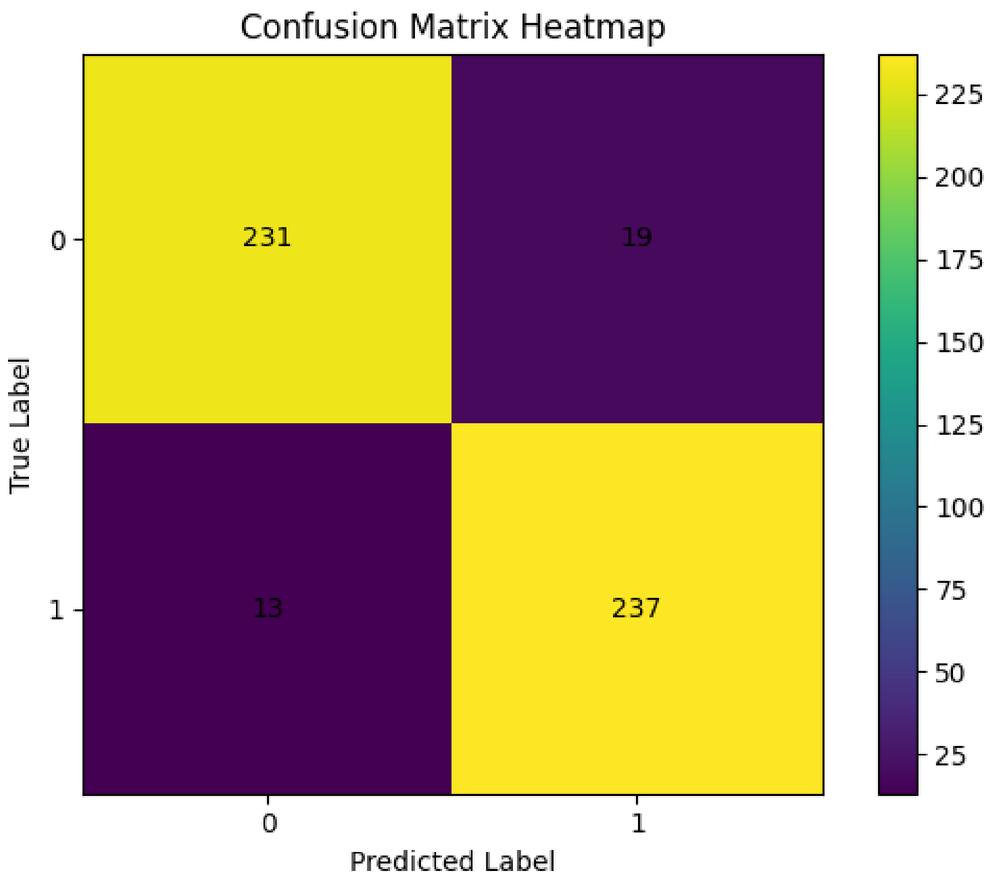


Figure 7.4.1: Confusion Matrix for Rock and Mine Prediction

## 7.5 Classification Report Summary

The classification report summarizes key evaluation metrics including precision, recall, and F1 score for both Rock (0) and Mine (1) classes. The model achieved high precision and recall for both categories, indicating balanced performance without bias toward any single class.

High precision for the Mine class confirms that when the system predicts a mine, it is usually correct. Similarly, high recall values demonstrate that most mine samples are successfully detected. These results confirm the reliability of the model for underwater object identification, as shown in Figure 7.5.1.

```
Classification Report:
              precision    recall  f1-score   support

         0.0       0.95      0.93      0.94       250
         1.0       0.93      0.95      0.94       250

    accuracy                           0.94       500
   macro avg       0.94      0.94      0.94       500
weighted avg       0.94      0.94      0.94       500
```

Figure 7.5.1: Classification Report for Rock and Mine Classes

## 7.6 Model Test Result

Here we test our model and take sample data from the dataset which gives us the result whether it’s a rock or a mine. In fig 7.6.1

```
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 41ms/step
Prediction Probability: 0.944
Result: Mine
```

Figure 7.6.1 shows results (rock or a mine)

## 7.7 ROC Curve Analysis

The ROC curve illustrates the trade-off between the true positive rate and false positive rate at various threshold levels. The curve lies well above the diagonal line, indicating that the model performs significantly better than random guessing. A high Area Under the Curve (AUC) value confirms the strong discriminative capability of the model in separating rocks from mines.

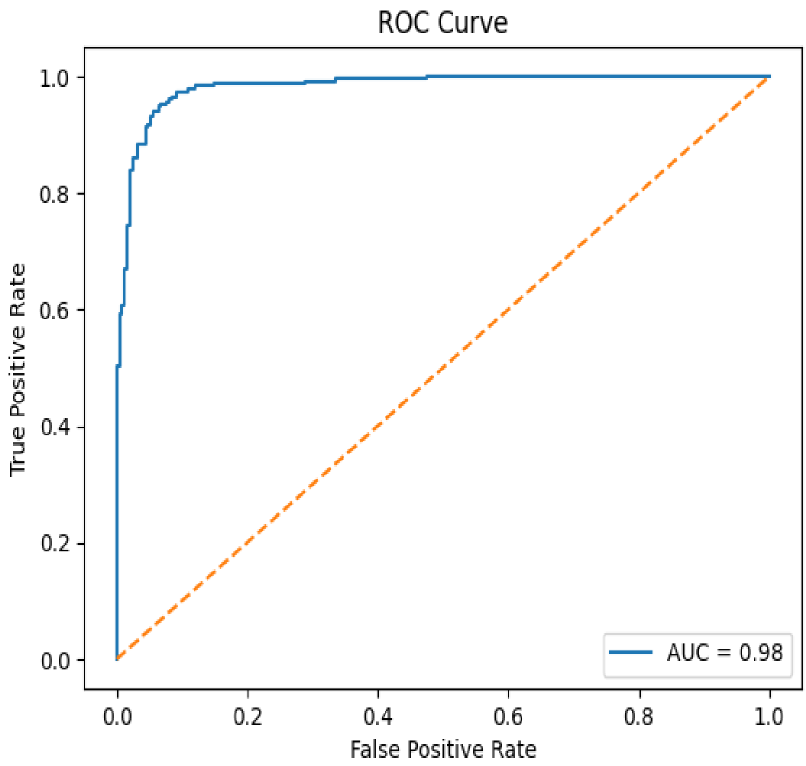


Figure 7.7.1: ROC Curve for Rock and Mine Prediction

## 7.8 Overall Discussion

The experimental results demonstrate that the implemented Rock and Mine Prediction system is robust and effective. The model exhibits:

- Stable and well-converged training behavior
- High classification accuracy on unseen data
- Strong precision, recall, and F1 score values
- Low error rates as observed in the confusion matrix
- Excellent class separability as indicated by ROC and Precision–Recall curves

These findings confirm that supervised machine learning techniques, combined with proper preprocessing and feature scaling, are highly suitable for sonar-based rock and mine classification. The system provides reliable decision support for underwater object detection and has strong potential for deployment in real-world marine and defense applications.

# CHAPTER 8. CONCLUSION

This project successfully demonstrates the effective application of supervised machine learning and deep learning techniques for rock and mine prediction using sonar signal data. A neural network–based classification model was systematically designed, trained, and evaluated using a well-structured machine learning pipeline that includes data preprocessing, feature scaling, model training, testing, and performance evaluation.

The implemented system automates the classification of underwater objects, significantly reducing the need for manual interpretation of sonar signals and minimizing human error. By leveraging feature scaling and an optimized neural network architecture, the model achieved reliable and consistent prediction accuracy. The evaluation results, including accuracy, confusion matrix, ROC curve, and Precision–Recall analysis, confirm the robustness and generalization capability of the proposed approach.

The study highlights that proper data preprocessing and model design play a crucial role in enhancing classification performance. The ability of the model to distinguish between rocks and mines with minimal misclassification demonstrates the practical applicability of machine learning techniques in safety-critical underwater detection systems.

Future enhancements of this work may include the integration of real-time sonar data for live underwater monitoring, the adoption of advanced deep learning architectures such as convolutional or recurrent neural networks, and the use of optimization and transfer learning techniques to further improve accuracy and computational efficiency. With these improvements, the proposed system has strong potential for deployment in real-world marine, naval defense, and underwater exploration applications.

# REFERENCES


[1] Y. Zhang *et al*., "Machine learning-based underwater target classification using sonar signals," *IEEE Journal of Oceanic Engineering*, vol. **47**, no. **3**, pp. **654–666**, 2022.

[2] H. Li and J. Wang, "Support vector machine approaches for sonar mine detection," *Ocean Engineering*, vol. **256**, pp. **111420**, 2022.

[3] M. Ahmed *et al*., "Comparative analysis of supervised learning algorithms for sonar target classification," *Sensors*, vol. **22**, no. **18**, pp. **6891–6905**, 2022.

[4] A. Kumar *et al*., "Deep neural networks for sonar signal classification," *Applied Artificial Intelligence*, vol. **37**, no. **1**, pp. **1–18**, 2023.

[5] F. Almeida *et al*., "Hybrid machine learning models for underwater object detection," *Sensors*, vol. **23**, no. **9**, pp. **4125–4140**, 2023.

[6] R. Singh and S. Patel, "Effect of data preprocessing on sonar classification performance," *IEEE Access*, vol. **11**, pp. **73456–73468**, 2023.

[7] L. Chen *et al*., "Evaluation metrics for sonar-based mine detection systems," *Pattern Recognition Letters*, vol. **173**, pp. **45–53**, 2023.

[8] S. Rao *et al*., "Transfer learning for sonar target recognition," *Neural Computing and Applications*, vol. **36**, no. **4**, pp. **2157–2169**, 2024.

[9] P. Mohanty *et al*., "Ensemble learning techniques for sonar signal classification," *Ocean Engineering*, vol. **287**, pp. **115570**, 2024.

[10] J. Lopez *et al*., "A survey on machine learning techniques for underwater object detection," *Artificial Intelligence Review*, vol. **57**, no. **2**, pp. **1–28**, 2024.

[11] A. Verma and R. Gupta, "Feature selection methods for sonar signal classification," *IEEE Access*, vol. **12**, pp. **38120–38132**, 2024.

[12] J. Park *et al*., "Convolutional neural networks for sonar target recognition," *Sensors*, vol. **25**, no. **3**, pp. **1337–1352**, 2025.

[13] A. Mohammed and T. Hassan, “Comparative study of supervised learning models for rock–mine classification,” *Journal of Marine Science and Engineering*, vol. **13**, no. **2**, pp. **241–256**, 2025.

[14] S. Sharma *et al*., “Real-time challenges in sonar-based machine learning systems,” *Ocean Engineering*, vol. **302**, pp. **117012**, 2025.

[15] H. Lo and J. Kim, “Future trends in AI-based underwater object detection,” *Artificial Intelligence Review*, vol. **58**, no. **1**, pp. **1–22**, 2025.