PVT-EfficientNet Dual Encoder with Mamba Head for Efficient Underwater Image Classification

Aufaclav Zatu Kusuma Frisky*[,1], Ari Dwi Hartanto[2], Hanum Khairana Fatmah[1], Waffiq Maaroja[1], Fadillah Siva[1], Novelio Putra Indarto[1], Mohammad Akbar Ghifari Tuasikal[1] and Jia-Ching Wang[3]
[1]Department of Computer Science and Electronic, Universitas Gadjah Mada, Yogyakarta, Indonesia
[2]Department of Mathematics, Universitas Gadjah Mada, Yogyakarta, Indonesia
[3]Department of Computer Science and Information Engineering, National Central University, Taiwan
E-mail: aufaclav@ugm.ac.id, ari@ugm.ac.id
*Corresponding author

Keywords: Underwater image classification, PVT-v2, EfficientNet, Mamba block, multi-scale feature fusion

Received: September 22, 2025

Identification and classification of marine organisms remain challenging due to pose variation, partial occlusions, and underwater imaging conditions. Existing Convolutional Neural Network (CNN) and Transformer models often struggle to obtain long-range contextual understanding while maintaining computational efficiency. This research proposes a new method named PVT Fusion Mamba architecture, which integrates PVT-v2-B2 and EfficientNet-B0 in a dual-encoder backbone, followed by a hierarchical fusion neck and a Mamba-based classification head. This architecture enables effective multi-scale feature integration and efficient long-range dependency modeling with linear complexity, while dynamically emphasizing organism-relevant features and suppressing background noise. We conducted experiments using the ROUD Dataset across ten marine organism classes. An extensive ablation study confirmed the synergistic effect of the dual-encoder fusion, demonstrating that the combined PVT Fusion Mamba architecture significantly outperforms its single-encoder counterparts (EfficientNet-B0 and PVT-v2-B2) in terms of convergence speed and final accuracy. Furthermore, in comparative studies against models like ResNet50 and YOLOv8, our proposed architecture achieved superior performance. The PVT Fusion Mamba architecture attained state-of-the-art accuracy of 98.6% with an optimized validation loss of 0.062 (at configuration $C_1 = 96, C_2 = 288, C_3 = 192$). Analysis of the confusion matrix reveals excellent classification performance, with most errors occurring only between morphologically similar species. The results demonstrate that PVT Fusion Mamba successfully overcomes the limitations of previous methods, achieving superior accuracy and robustness with reduced computational cost compared to established deep learning models.

Povzetek: Predlagana arhitektura PVT Fusion Mamba združuje dvojni kodirnik PVT-$v2$-$B2$ in $EfficientNet$-$B0$ s klasifikacijsko glavo na osnovi modela Mamba, s čimer dosega vrhunsko natančnost 98.6% na naboru podatkov ROUD ter učinkovito modelira dolgosežne odvisnosti z linearno kompleksnostjo za robustno identifikacijo morskih organizmov.

1 Introduction

The classification task for underwater objects is an important task for exploring underwater conditions and monitoring the environment and empowering underwater objects [1]. This is still challenging because the underwater environment often displays images due to unclear images, distracting backgrounds, and low contrast [2][3][4]. This makes the classification process using conventional methods of manual pattern recognition often inefficient and also prone to errors. Various studies from the use of classic CNNs show that the models still have limitations and errors in capturing complex features. VGGNet, ResNet and MobileNet have been modified to improve accuracy in this task [5][6]. However, research shows that the CNN model still has limitations, especially in capturing complex features and crucial long-range dependencies. This encouraged researchers to explore more sophisticated architectures so that the Transformer emerged with its self- attention mechanism which can capture global relationships throughout the image [7]. This approach overcomes object degradation and uses a hybrid model to combine the power of local feature extraction. Research by [8] proposed TC_YOLO, a model that combines features from transformers to improve underwater object detection performance, while [9] came up with a hybrid CNN-Transformer architecture that has been explored for underwater object segmentation.

In this paper, we propose the PVT Fusion Mamba architecture which integrates EfficientNet-B0 and PVT-v2-B2 within a dual-encoder backbone. Our contribution is threefold:

– Implement a dual-encoder strategy to simultaneously

leverage the local efficiency of depthwise convolutions and the global reach of pyramid attention.

- Introduce a hierarchical fusion neck that serves as an adaptive feature balancer, essentially acting as an adaptive filter to reconcile conflicting visual data from the two encoders.
- Employ a Mamba-based classification head that functions as an adaptive control mechanism. By managing state transitions over time, the Mamba block selectively emphasizes organism-relevant features while suppressing the noisy backgrounds characteristic of complex reef environments.

Experimental results on the ROUD dataset demonstrate that our proposed architecture achieves a state-of-the-art accuracy of 98.6% with significantly reduced computational cost compared to established deep learning models. This performance suggests that the integration of adaptive fusion and selective state-space modeling is highly effective for robust marine organism classification in real-world underwater exploration systems.

2 Related works

Over the last decade, Convolutional Neural Networks (CNNs) have served as the foundational paradigm for image classification tasks. Classical architectures such as AlexNet, VGGNet, and ResNet have demonstrated significant efficacy in increasing classification accuracy, particularly when applied to specialized datasets [10][11][12][13][14][15]. In the context of marine biology, residual convolutional networks have been specifically modified to address the inherent challenges of blurriness and low contrast when identifying fish and invertebrates [16][17]. Recent studies have also combined ResNet50 with preprocessing methods like max-RGB and shades of gray to enhance underwater vision before classification [18]. Furthermore, CNNs have been integrated with second-order pooling to capture temporal correlations in radiated acoustic signals for sonar-based object classification [19]. Despite these advancements, CNNs remain limited by their inductive bias, which favors local features over long-range contextual dependencies [20].

To address the receptive field limitations of CNNs, the Vision Transformer (ViT) and its variants emerged as a solution for modeling global relationships between image patches [21][22]. Specialized attention modules, such as Inception Attention (IA), have been shown to significantly outperform classic networks like AlexNet and InceptionV3 in underwater classification tasks [23]. Hybrid approaches have also explored integrating Swin Transformer attention into frameworks like YOLOv8 to improve feature extraction from complex imagery and mitigate background noise [24]. Other innovations include the use of ensemble deep learning with YOLOV9 and domain attention mechanisms that weight input from feature maps to enhance processed outputs [25]. Recently, hybrid classical-quantum CNN methods have even been proposed for on-board underwater image classification [26].

A key strategy for improving robust classification involves the simultaneous leverage of detailed spatial information and high-level semantic data. This is commonly achieved through Feature Pyramid Networks (FPN) and multi-pathway fusion methods [27][28][29]. In underwater research, hybrid models like TC-YOLO have combined transformer features to improve detection performance [8], while the FLSSNet architecture explored CNN-Transformer hybrids for sonar image segmentation [9]. Our proposed method builds upon these concepts by integrating EfficientNet-B0 and PVT-v2 as a dual-encoder backbone.

The emergence of State-Space Models (SSMs), specifically the Mamba architecture, provides a path toward modeling long-range dependencies with linear complexity relative to spatial resolution [30][31]. Unlike standard self-attention, the Mamba block utilizes a selective memory mechanism to dynamically emphasize organism-relevant regions while suppressing the high-entropy background noise characteristic of reef environments [32]. This research integrates a Mamba-based classification head as a robust state-space controller to maintain stable feature extraction even when the input signal is degraded by underwater conditions.

3 Methodology

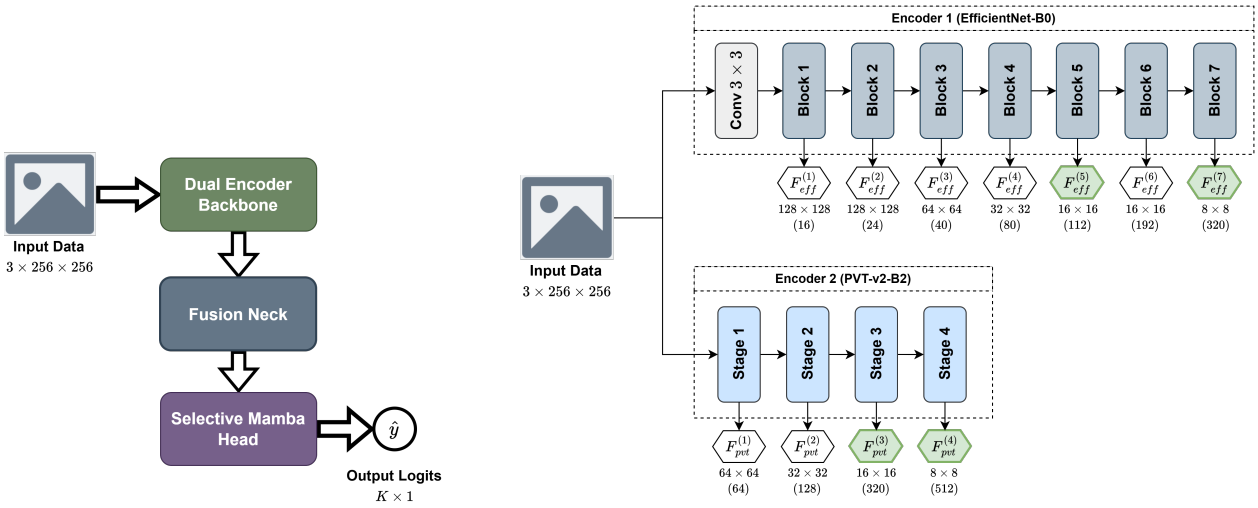
This section details the development of the PVT Fusion Mamba architecture, a hybrid framework engineered for the robust classification of marine organisms in challenging underwater environments, as well as the dataset preparation and training steps used for our experiments.

3.1 Architectural overview

The PVT Fusion Mamba is designed as an adaptive control system to solve the problem of image ”noise” in underwater environments. As shown in Figure 1a, the architecture consists of three main parts that work together to maintain stable classification performance: a dual-encoder backbone, a hierarchical fusion neck, and a selective state-space classification head.

3.2 Multi-scale feature extraction backbone

The backbone serves as the primary sensing unit of the control system, responsible for converting raw pixel data into high-dimensional feature representations. As illustrated in Figure 1b, the architecture utilizes two parallel pathways via Encoder 1 and Encoder 2 to ensure that both local textures and global context are captured simultaneously.



(a) Proposed overall architecture.

(b) Detailed architecture of the dual-encoder backbone.

Figure 1: Overview of the model components: (a) shows the high-level PVT Fusion Mamba architecture, and (b) details the dual-encoder backbone flow.

3.2.1 Encoder 1

Encoder 1 utilizes EfficientNet-B0 to capture fine-grained spatial details [33]. This encoder is structured through a series of Mobile Inverted Bottleneck Convolutions (MBConv), which utilize depthwise separable convolutions to maintain efficiency. Given an input image $I \in \mathbb{R}^{3\times256\times256}$, the process can be defined as:

$$\Phi_{eff}(I) = \left\{F_{eff}^{(1)}, ..., F_{eff}^{(7)}\right\} \tag{1}$$

where $F_{eff}^{(i)}$ is feature map from i-th block output. In this model, we used $F_{eff}^{(5)} \in \mathbb{R}^{112\times16\times16}$ which represents intermediate local features and $F_{eff}^{(7)} \in \mathbb{R}^{320\times8\times8}$ which captures more complex, high-level spatial patterns.

3.2.2 Encoder 2

Encoder 2 employs PVT-v2-B2 to provide long-range contextual understanding [34]. Unlike standard CNNs, the Transformer uses a pyramid attention mechanism to process the image at different resolutions. The core operation involves a multi-head self-attention (MHSA) function and the process can be defined as:

$$\Phi_{pvt}(I) = \left\{F_{pvt}^{(1)}, ..., F_{pvt}^{(4)}\right\} \tag{2}$$

where $F_{pvt}^{(i)}$ is output feature extracted from i-th stage. In this model, we used the output from stage 3 and 4 feature map, $F_{pvt}^{(3)} \in \mathbb{R}^{320\times16\times16}$ and $F_{pvt}^{(4)} \in \mathbb{R}^{512\times8\times8}$. By combining these Transformer-based features with the CNN-based features, the model can effectively identify organisms even when they are partially occluded or blend into the background.

3.3 Adaptive fusion neck and feature conversion

The fusion neck acts as an adaptive integration mechanism that reconciles the distinct visual signals from the dual-encoder backbone. To ensure the Transformer-based features are compatible with the Convolutional features, the process begins with signal alignment followed by hierarchical multi-scale fusion as shown in Figure 2.

3.3.1 Transformer feature converter (TFC)

Before integration, the features from Encoder 2 undergo standardization to align their dimensionality with the CNN feature space from Encoder 1. The conversion is defined by the function Φ_{tfc}:

$$\Phi_{tfc}(X, c) = \sigma_{ReLU}(\text{BN}(\text{Conv}_{1\times1}(X, c))) \tag{3}$$

where X is input data, $\text{Conv}_{1\times1}$ is a point-wise convolution with target channel size c, BN denotes Batch Normalization, and σ_{ReLU} is the ReLU activation function. This function convert the channel size of input data, while maintaining the resolution (width and height).

We want to fuse the feature from Encoder 1 and 2 with same resolution, so block 5 feature map (from Encoder 1) will be paired with stage 3 feature map (from Encoder 2), and block 7 feature map (from Encoder 1) will be paired with stage 4 feature map (from Encoder 2). Thus we transform the Encoder 2 feature map to match the channel size from the respective channel size in Encoder 1, and we get transformed features as:

$$F_{tfc}^{(3)} = \Phi_{tfc}\left(F_{pvt}^{(3)}, 112\right) \in \mathbb{R}^{112\times16\times16} \tag{4}$$

$$F_{tfc}^{(4)} = \Phi_{tfc}\left(F_{pvt}^{(4)}, 320\right) \in \mathbb{R}^{320\times8\times8} \tag{5}$$

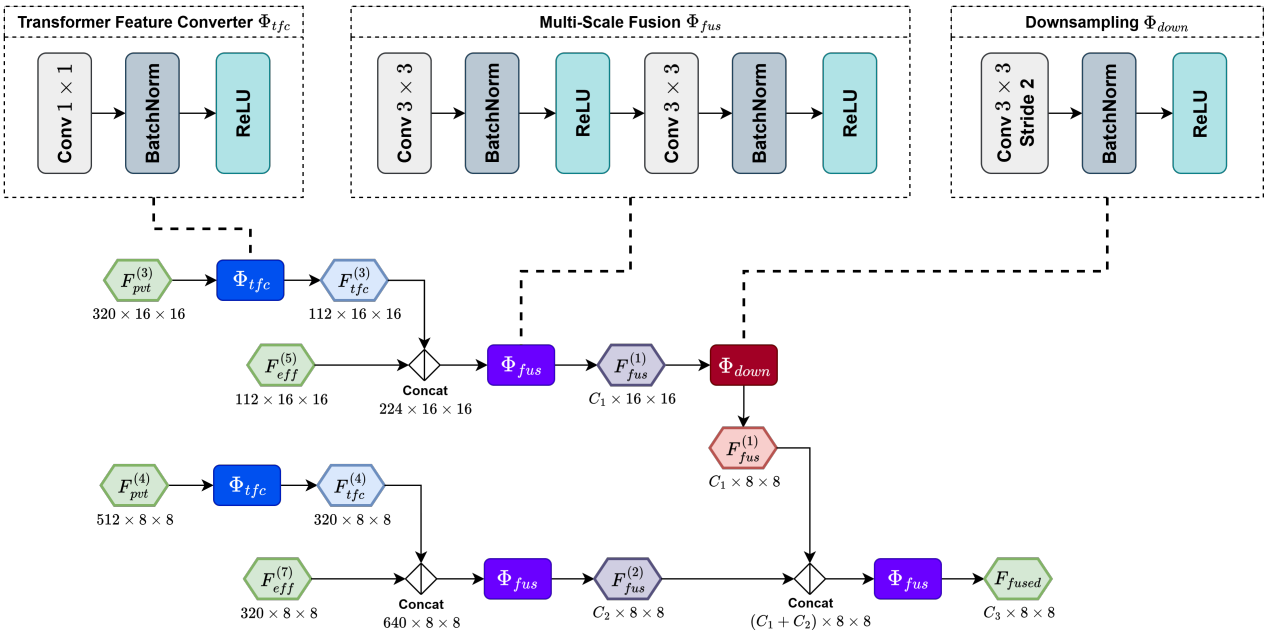


Figure 2: Detailed architecture of the hierarchical fusion neck, illustrating the adaptive integration of multi-scale features from CNN and Transformer encoders.

3.3.2 Hierarchical multi-scale fusion

The fusion neck employs a cascaded strategy to integrate fine-grained spatial details with high-level contextual patterns while controlling parameter growth. The main fusion process is defined by the function Φ_{fus}:

$$\Phi_{fub}(X, c) = \sigma_{ReLU}(\text{BN}(\text{Conv}_{3\times3}(X, c))) \quad (6)$$

$$\Phi_{fus}(X_1, X_2, c) = \Phi_{fub}\left(\Phi_{fub}(X_1 \| X_2, c), c\right) \quad (7)$$

where X_1 and X_2 are input tensor data with same resolution, c is the target channel size, and $\|$ is channel-wise concatenation operator. Then the fusion neck process is divided into two adaptive stages:

Same Resolution Fusion. The converted transformer features from Encoder 2 are concatenated with the respective CNN features from Encoder 1 with same resolution to form an integrated tensor by the fusion function:

$$F_{fus}^{(1)} = \Phi_{fus}\left(F_{eff}^{(5)}, F_{tfc}^{(3)}, C_1\right) \quad (8)$$

$$F_{fus}^{(2)} = \Phi_{fus}\left(F_{eff}^{(7)}, F_{tfc}^{(4)}, C_2\right) \quad (9)$$

The result is a refined representation $F_{fus}^{(1)} \in \mathbb{R}^{C_1\times16\times16}$ and $F_{fus}^{(2)} \in \mathbb{R}^{C_2\times8\times8}$, where C_1 and C_2 are target channel for 16×16 and 8×8 fused feature, respectively.

Cross Resolution Fusion. To incorporate deeper semantic information, 16×16 fused feature is downsampled via a stride-2 convolution block to halved the resolution, and fused again with 8×8 fused feature:

$$\Phi_{down}(X) = \sigma_{ReLU}(\text{BN}(\text{Conv}_{3\times3,\ \text{stride}\ 2}(X))) \quad (10)$$

$$F_{fused} = \Phi_{fus}\left(\Phi_{down}\left(F_{fus}^{(1)}\right), F_{fus}^{(2)}, C_3\right) \quad (11)$$

The final output is the representation $F_{fused} \in \mathbb{R}^{C_3\times8\times8}$, where C_3 is target channel for fused feature, which provides a balanced and robust feature set for classification. This hierarchical approach allows the model to adaptively emphasize relevant organism features even when the input signal is degraded by underwater turbidity.

3.4 Selective mamba head

The classification head serves as the final robust filter, responsible for mapping the high-dimensional fused representation $F_{fused} \in \mathbb{R}^{C_3\times8\times8}$ into a stable multi class probability distribution. To maintain structural stability against underwater noise and pose variations, the head utilizes a Mamba-based state-space block configured with a 4-directional spatial scanning strategy as shown in Figure 3.

3.4.1 Selective state-space modeling

The core of the Mamba block is a discretized State-Space Model (SSM) that functions as a robust observer. The transition of the hidden state h_t and the filtered output y_t are governed by:

$$h_t = \bar{\mathbf{A}}h_{t-1} + \bar{\mathbf{B}}x_t, \quad y_t = \mathbf{C}h_t \quad (12)$$

where $\bar{\mathbf{A}}$ and $\bar{\mathbf{B}}$ are discretized transition matrices. To handle underwater uncertainty, the system employs a selection mechanism where $\bar{\mathbf{B}}$, $\mathbf{C}$, and the step size Δ are dynamic functions of the input x_t:

$$\Delta = \text{Softplus}(\text{Parameter} + \text{Linear}(x_t)) \quad (13)$$

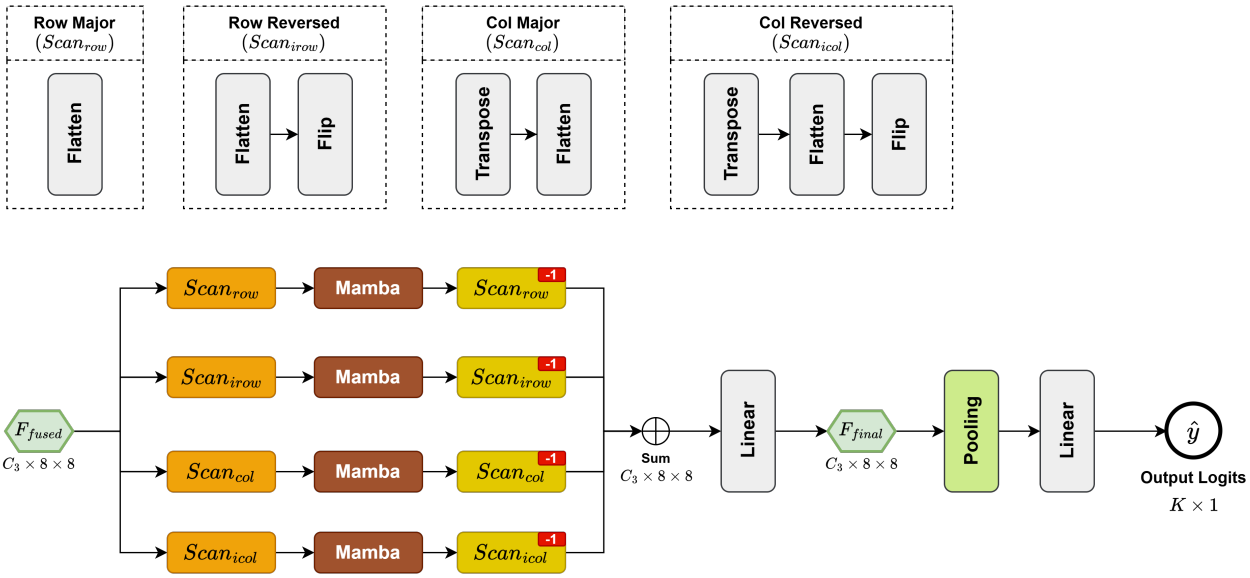


Figure 3: Detailed architecture of the Mamba-based classification head, illustrating the 4-directional scanning logic (row-wise, column-wise, and their reversals) to ensure spatial robustness.

This selectivity allows the head to function as an adaptive controller to remembers organism-relevant signals in the latent state while forgetting stochastic disturbances caused by background noise or turbidity.

3.4.2 4-directional spatial scanning

Standard Mamba blocks are designed for 1D sequences. However, underwater organisms appear in unpredictable orientations. To ensure spatial invariance, the head flattens the input tensor $X \in \mathbb{R}^{W \times H}$ into four distinct scan paths using this functions:

$$\text{Scan}_{row}(X) = \text{Flatten}(X) \tag{14}$$

$$\text{Scan}_{irow}(X) = \text{Flip}(\text{Scan}_{row}(X)) \tag{15}$$

$$\text{Scan}_{col}(X) = \text{Flatten}(X^T) \tag{16}$$

$$\text{Scan}_{icol}(X) = \text{Flip}(\text{Scan}_{col}(X)) \tag{17}$$

where Flatten is the standard operation that serializes the 2D spatial grid into a 1D sequence of length $L = H \times W$, Flip represents the operator to reverse the ordering of sequence. The output of each function $\text{Scan}_{row}, \text{Scan}_{irow}, \text{Scan}_{col}, \text{Scan}_{icol}$ representing row-major, row-reversed, column-major, and column-reversed trajectories, respectively.

Given fused feature, we have $F^{(i)}_{fused} \in \mathbb{R}^{8\times8}$ as i-th feature map from F_{fused}. Each of feature map will be processed separately using this function:

$$\Phi_{scan}(X, dir) = \text{Scan}^{-1}_{dir}(\text{Mamba}(\text{Scan}_{dir}(X))) \tag{18}$$

$$\Phi_{mamba}(X) = \text{Linear}\left(\sum_{dir} \Phi_{scan}(X, dir)\right) \tag{19}$$

where the summation acts as an ensemble of directional filters $dir \in \{row, irow, col, icol\}$, ensuring that the model captures relevant features regardless of the subject's pose or movement direction, and Mamba is Mamba block function. Thus we get the processed feature $F^{(i)}_{final} = \Phi_{mamba}\left(F^{(i)}_{fused}\right)$ for each i, and we combine into single tensor $F_{final} \in \mathbb{R}^{C_3\times8\times8}$.

3.4.3 Global pooling and classification

The filtered tensor is processed through a global average pooling (GAP) layer to reduce the spatial dimensions to a C_3 dimension vector. This vector is fed into a fully connected layer:

$$y = \text{Linear}(\text{Pooling}(F_{final})) \tag{20}$$

where Pooling is global average pooling layer and Linear yielding the final classification across the K marine organism classes with linear complexity relative to the input resolution. Thus we get the output logits $y \in \mathbb{R}^K$.

3.5 Dataset adaptation and preprocessing

The study utilizes the ROUD dataset, which was originally developed for underwater object detection [35]. The original dataset comprises approximately 14,000 images containing ten distinct marine organism categories labeled with bounding boxes.

To adapt this for a classification task, each bounding box was cropped from the original high-resolution frames. This process extracted all individual organism instances, resulting in the following raw distribution across the ten categories as shown in Table 1.

Table 1: Distribution of extracted organism crops from the original ROUD dataset

Category	Count	Category	Count
Fish	9,090	Diver	4,393
Echinus	7,867	Cuttlefish	3,721
Corals	6,132	Turtle	2,824
Starfish	5,745	Jellyfish	1,881
Holothurian	5,244	Scallop	5,038

To ensure the model learns from the most distinctive morphological features, we selected the top 1,000 largest cropped images from each category. This largest-area selection strategy serves as a quality filter, prioritizing samples where the organism is most visible relative to background noise and turbidity.By choosing an equal number of samples per class, we created a balanced dataset of 10,000 images. This prevents the model from developing a classification bias toward more frequent species, such as fish. Finally, all selected images were resized to a uniform 256×256 resolution to match the input requirements of the dual-encoder backbone.

3.6 Augmentation and training strategy

To improve generalization and ensure the robustness of the PVT Fusion Mamba, we implement a data augmentation pipeline designed to simulate the stochastic nature of underwater environments. The training dataset undergoes a series of transformations to account for the unpredictable conditions of the marine medium. These include:

- **Spatial Augmentation**: To address varied subject orientations, we apply Random Horizontal Flipping and Random Rotation of up to $10°$. All images are resized to a uniform 256×256 resolution to match the input requirements of the dual-encoder backbone.
- **Photometric Augmentation**: To simulate variable lighting and water clarity, we utilize Color Jittering with adjustments to brightness (0.2), contrast (0.2), saturation (0.2), and hue (0.1).
- **Normalization**: Pixel values are normalized using standard ImageNet statistics ($mean = [0.485, 0.456, 0.406]$, $std = [0.229, 0.224, 0.225]$) to ensure stable gradient flow during optimization.

The model is trained for 20 epochs using the Adam optimizer with a weight decay of 10^{-2} to prevent overfitting. The training parameters are defined as follows:

- **Learning Rate Control**: We initiate training with a learning rate of 10^{-4}.
- **Adaptive Scheduling**: A learning rate scheduler is implemented to reduce the rate by a factor of 0.2 if the validation loss does not show improvement for two consecutive epochs.
- **Gradient Management**: Gradient clipping is employed to maintain numerical stability throughout the training process.

This combination of robustness-driven augmentation and precise optimization allows the architecture to achieve high classification accuracy while maintaining computational efficiency.

4 Results and discussions

4.1 Training dynamics and convergence analysis

The learning performance of the proposed architecture was evaluated over 20 training epochs using cross-entropy loss and the Adam optimizer. As illustrated in Figure 4, the model trained with sample hyperparameter setting for channel size $C_1 = 128, C_2 = 320$, and $C_3 = 256$ demonstrated a rapid and stable convergence profile, which is critical for maintaining robust performance under uncertain underwater conditions.

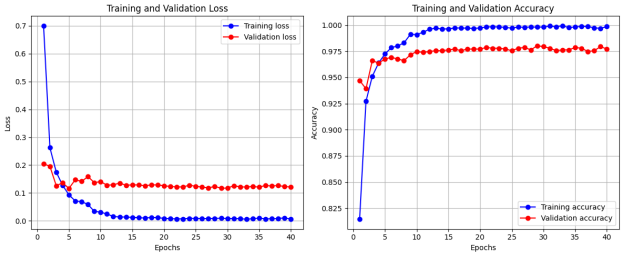


Figure 4: Training and validation comparison of loss and accuracy for the PVT Fusion Mamba architecture

The training process exhibited the following characteristics:

- **Training Stability**: Training accuracy consistently reached high levels 99.9% using sample hyperparameter configurations. The training loss effectively converged to minimal values 0.022, indicating that the dual-encoder backbone successfully captured the complex features required for marine organism classification.
- **Validation Performance**: This model achieved a peak validation accuracy of 97.7% with a corresponding validation loss of 0.12. This high performance on unseen data demonstrates strong generalization capabilities, even when faced with the high-entropy noise typical of underwater reef environments. The narrow gap between training and validation accuracy (approximately 2.2%) suggests that the architecture does not merely memorize the training set but has learned robust, invariant features.
- **Convergence Speed**: Models demonstrated high efficiency, reaching convergence within starts from 12

Table 2: Hyperparameter sensitivity analysis results across 27 configurations

C_1	C_2	C_3	**Epoch**	**Val Loss**	**Val Acc**
96	256	192	14	0.075	0.981
96	256	224	15	0.069	0.981
96	256	256	13	0.070	0.983
112	256	192	15	0.062	0.983
112	256	224	18	0.068	0.982
112	256	256	13	0.065	0.982
128	256	192	18	0.069	0.982
128	256	224	11	0.068	0.980
128	256	256	15	0.075	0.981
96	**288**	**192**	**14**	**0.062**	**0.986**
96	288	224	16	0.066	0.985
96	288	256	13	0.059	0.985
112	288	192	12	0.070	0.977
112	288	224	10	0.072	0.979
112	288	256	9	0.070	0.983
128	288	192	18	0.069	0.982
128	288	224	19	0.072	0.982
128	288	256	11	0.071	0.981
96	320	192	19	0.068	0.983
96	320	224	18	0.076	0.980
96	320	256	12	0.068	0.981
112	320	192	19	0.074	0.982
112	320	224	18	0.072	0.978
112	320	256	18	0.071	0.982
128	320	192	15	0.059	0.982
128	320	224	19	0.075	0.982
128	320	256	12	0.079	0.981

epochs. This rapid learning is aided by the adaptive learning rate scheduler, which reduced the learning rate by a factor of 0.2 if the loss failed to improve for two consecutive epochs.

From a control systems perspective, the stable gap between training and validation metrics confirms that the adaptive fusion neck and selective Mamba head function effectively as robust filters. These components managed to suppress stochastic background noise and environmental disturbances, allowing the model to focus on organism-relevant signals without succumbing to overfitting.

4.2 Hyperparameter sensitivity analysis

The performance of the PVT Fusion Mamba was evaluated across 27 distinct configurations, systematically varying the 16×16 fusion channel size C_1 $(96, 112, 128)$, 16×16 fusion channel size C_2 $(256, 288, 320)$, and cross resolution fusion channel size C_3 $(192, 224, 256)$, to determine the optimal balance between computational efficiency and classification accuracy. This process is analogous to tuning a control system to achieve maximum stability despite environmental noise. Training was performed for 9-19 epochs until convergence, with the results as shown in Table 2.

Based on that results, all configurations achieving remarkably high training accuracy ranging from 99.3% to 99.9%. The training loss consistently converged to very low values between 0.004 and 0.022, indicating effective learning across all parameter settings. Validation performance showed more variation, with validation accuracy ranging from 97.7% to 98.6% and validation loss between 0.059 and 0.079.

4.2.1 Impact of 16×16 fusion channel size (C_1)

The 16×16 fusion channel size parameter control the depth of the adaptive fusion. It shows a pronounced sweet spot at $C_1 = 96$, where all top-performing configurations cluster. Increasing C_1 to 112 or 128 consistently resulted in degraded validation performance, suggesting that higher-dimensional vision features introduce unnecessary complexity that hampers generalization. This counterintuitive finding challenges the common assumption that increased feature dimensionality always improves performance, instead highlighting the importance of feature dimension optimization for domain-specific applications. The optimal $C_1 = 96$ appears to capture sufficient visual information while avoiding the curse of dimensionality in the marine organism classification context.

4.2.2 Impact of 8×8 fusion channel size (C_2)

The 8×8 fusion channel size parameter determines the granularity at which the model observes the underwater signal. It demonstrates a clear optimal point at 288, where configurations consistently achieve superior validation performance compared to both smaller $(C_2 = 256)$ and larger $(C_2 = 320)$ values. This finding suggests that channel size 288 provides the optimal granularity for capturing marine organism features while maintaining computational efficiency. Configurations with $C_2 = 320$, despite achieving the highest training accuracy (99.9%), exhibited signs of overfitting with relatively lower validation accuracy, indicating that excessively large patch sizes may lead to over-parameterization for this specific domain.

4.2.3 Impact of cross resolution fusion channel size (C_3)

The cross resolution fusion channel size parameter control the depth of residual fusion between all encoders and Mamba-based filtering. It exhibits more nuanced behavior, with optimal performance achieved across all range of values when combined with optimal C_1 and $C - 2$ settings. However, $C_3 = 192$ consistently produces the most stable results with minimal overfitting, suggesting it provides adequate temporal modeling capacity without excessive complexity. The relatively stable performance across different C_3 values indicates that temporal features, while beneficial, are less critical than spatial feature extraction parameters for static marine organism classification tasks.

4.2.4 Optimal system configuration

The optimal configuration was identified as $C_1 = 96$, $C_2 = 288$, $C_3 = 192$, achieving the highest validation accuracy of 98.6% with a validation loss of 0.062 after converging in 14 epochs. This configuration demonstrated superior generalization capability compared to more complex parameter settings. The second and third best performing configurations were $C_1 = 96$, $C_2 = 288$, $C_3 = 224$ (98.5% validation accuracy) and $C_1 = 96$, $C_2 = 288$, $C_3 = 256$ (98.5% validation accuracy), respectively. Notably, all top-performing configurations shared the same value of $C_1 = 96$ and $C_2 = 288$, suggesting these parameters provide an optimal balance for the marine organism classification task.

The training dynamics analysis reveals interesting convergence patterns across configurations. Models with optimal hyperparameter settings ($C_1 = 96, C_2 = 288$) demonstrate rapid convergence within 13-16 epochs, while suboptimal configurations require longer training periods or fail to achieve comparable validation performance. The consistent achievement of high training accuracy ($> 99\%$) across all configurations, combined with varying validation performance, underscores the importance of regularization and hyperparameter optimization in preventing overfitting. Early stopping mechanisms prove essential, as several configurations achieve optimal validation performance well before 20 epochs.

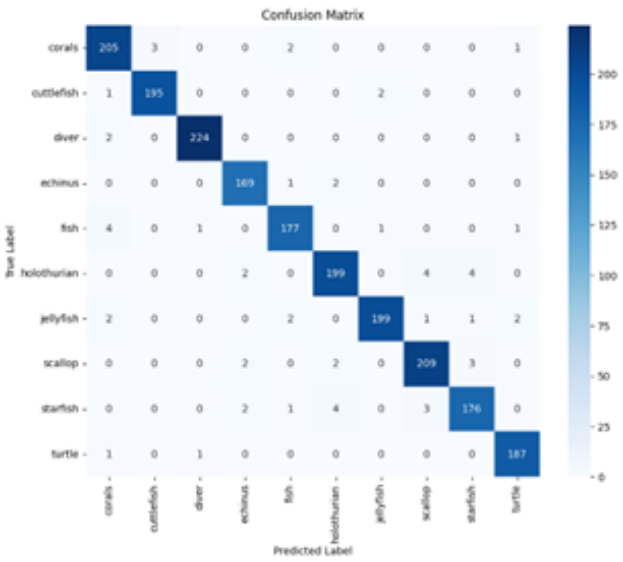


Figure 5: Confusion matrix for the best hyperparameters scenario

The confusion matrix analysis of the best-performing model reveals excellent classification performance across all ten marine organism classes, as detailed in Figure 5. The model achieved particularly strong performance on visually distinct classes such as diver (98.68% precision and recall), turtle (99.47% precision, 98.94% recall), and corals (97.16% recall, 92.34% precision). Classes with more subtle morphological differences, such as starfish (94.62% recall) and holothurian (95.22% precision and recall), showed slightly lower but still highly competitive performance metrics. This per-class performance analysis demonstrates that the proposed architecture effectively captures discriminative features for marine organism classification, with classification errors concentrated primarily among morphologically similar species pairs.

4.3 Computational efficiency and hardware scalability

The hardware evaluation, summarized in Table 3, focused on inference latency, throughput (FPS), and computational density (GFLOPs). The model maintains a constant memory footprint of approximately 261.88 MB with 30.08 million parameters, making it viable for edge deployment.

The experimental results reveal several key insights into the architectural efficiency. First, transitioning from a single-core baseline to a dual-core configuration yielded the most substantial performance improvement, reducing inference latency by approximately 24% (from 190.085 ms to 144.652 ms). This suggests that the dual-encoder backbone effectively utilizes parallel processing threads for simultaneous feature extraction. Second, a phenomenon of diminishing returns was observed when moving from two to three cores, with latency decreasing by only 2.8%. This behavior indicates that the computational gains are partially offset by the overhead of thread management and data synchronization required by the hierarchical fusion neck. Third, peak performance was achieved using four cores, reaching a minimum latency of 126.841 ms and a maximum throughput of 7.884 FPS. The consistent increase in GFLOPs (from 60.60 to 90.82) demonstrates the model's ability to leverage additional compute density.

From an architectural standpoint, while the model is highly efficient, the scaling is ultimately limited by hardware memory bandwidth bottlenecks. However, the ability to maintain over 5 FPS on a single core confirms that the PVT Fusion Mamba is well-suited for low-power, real-time underwater monitoring systems.

4.4 Comparative analysis with baseline models

The comparative study evaluated how different architectural designs handle the high-entropy noise of underwater imagery. As shown in Figure 6, the validation loss curves illustrate the stability and convergence speed of the proposed model relative to the baselines.

4.4.1 Learning behavior analysis

The baseline models exhibited varying degrees of sensitivity to underwater image uncertainty. From the convolutional baselines, EfficientNet-B0 demonstrated stable loss convergence (0.089 to 0.103) but was limited by its focus

Table 3: Hardware performance metrics across different CPU core configurations

CPU Cores	Inference Time (ms)	FPS	RAM Usage (MB)	GFLOPs
1	190.085	5.261	966.367	60.603
2	144.652	6.913	974.438	79.638
3	140.592	7.113	966.410	81.937
4	126.841	7.884	973.605	90.820

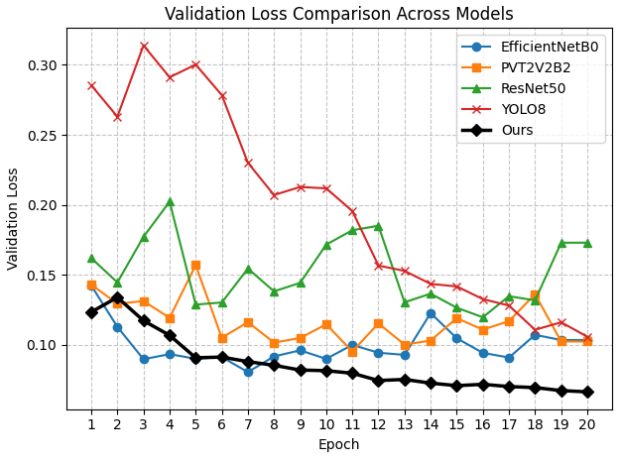


Figure 6: Comparative validation loss curves over 20 epochs for different models

on local features, which lacked the global context needed for complex reef environments. Conversely, ResNet50 showed high and fluctuating loss throughout training, likely due to exploding gradient issues in the deep residual layers. From the transformer baselines, PVT-v2-B2 provided competitive but fluctuating loss (0.095 to 0.115), reflecting the difficulty of maintaining stable global attention in noisy underwater scenes. Meanwhile from the detection-based baselines, YOLOv8 exhibited the highest initial loss (0.313) and required the longest duration to suppress error, only reaching a loss of 0.105 at the end of training.

4.4.2 Performance summary

The PVT Fusion Mamba successfully integrated the local extraction strength of EfficientNet and the long-range dependency modeling of PVT-v2. By utilizing the Mamba block as a robust filter, the proposed model suppressed loss by approximately 30–40% compared to single-encoder baselines and up to 50% compared to ResNet50.

These results confirm that our dual-encoder architecture acts as a more effective adaptive control system than standard single-pathway models, providing higher accuracy and stability for marine organism classification.

4.5 Ablation study

The Mamba block is a critical component of the classification head, designed to act as a robust state-space filter for underwater feature refinement. To justify its inclusion, we conducted an ablation study by replacing the 4-directional Mamba head with simpler alternative structures.

We compared the proposed architecture against two ablated versions:

- **Global Average Pooling (GAP) Only**: The Mamba block was removed, and the final fused features ($F_{final} \in \mathbb{R}^{C_3 \times 8 \times 8}$) were passed directly to a GAP layer followed by the linear classifier.
- **Standard MLP Head**: The Mamba block was replaced with a multi-layer perceptron (MLP) to test the impact of a static, non-sequential classification head.
- **Single-Encoder Baselines**: Performance was also compared against the standalone EfficientNet-B0 and PVT-v2-B2 backbones to verify the necessity of the dual-encoder fusion.

In this context, the fusion channel size configuration was choosed from best performance model. As shown in Table 4, the removal of the Mamba block led to a degradation in classification performance, particularly in distinguishing morphologically similar species.

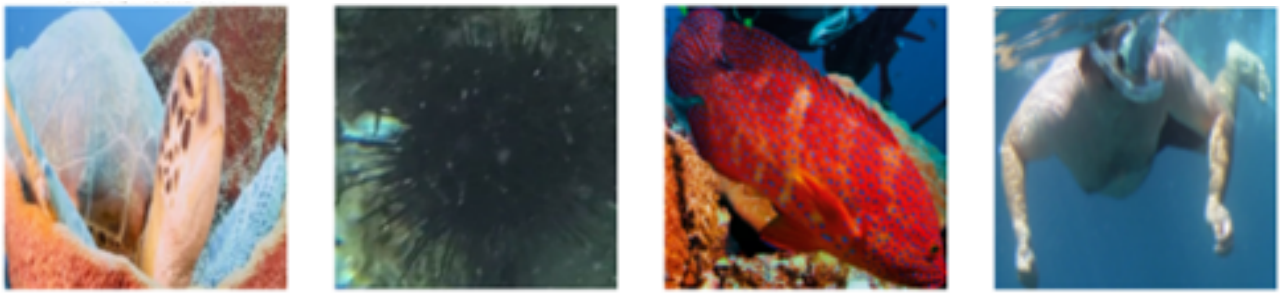
The experimental results confirm that the Mamba block serves as more than a simple feature aggregator. From a control systems perspective, the MLP and GAP-only configurations act as ”static” observers that treat all incoming spatial signals with equal weight. In contrast, the selective state-space mechanism in the Mamba head allows the model to adjusts the filtering parameters based on the input noise levels, it actively forgets background interference, such as distracting reef structures, while remembering organism-relevant patterns in the latent state, also the 4-directional scanning strategy ensures the model remains robust against unpredictable subject orientations, a feature lacking in standard convolutional or MLP heads. The synergy between the dual-encoder fusion and the Mamba head is what enables the model to significantly outperform its single-encoder counterparts in both convergence speed and final accuracy.

4.6 Error analysis and misclassification study

The detailed misclassification analysis based on the confusion matrix (Figure 5) reveals systematic patterns that provide valuable insights into both the model’s learning behavior and the inherent challenges of marine organism classification. The most frequent bidirectional confusion occurs

Table 4: Ablation results isolating the Mamba block performance gain

Architecture Configuration	Val Loss	Val Accuracy (%)
Dual-Encoder + GAP (No Mamba)	0.082	96.1
Dual-Encoder + MLP Head	0.080	96.5
Single Encoder (EffNet-B0 Only)	0.081	96.2
Single Encoder (PVT-v2-B2 Only)	0.095	95.7
PVT Fusion Mamba (Full)	**0.062**	**98.6**



(a) True: Turtle Pred: Corals (b) True: Holothurian Pred: Echinus (c) True: Fish Pred: Corals (d) True: Diver Pred: Turtle

Figure 7: Representative misclassification samples. These errors typically occur in high-entropy scenes where the subject shares color patterns or gross morphology with the background.

Table 5: Comparison of best validation performance across models

Model	Loss	Acc (%)
ResNet50	0.101	94.7
YOLOv8	0.120	93.9
PVT-v2-B2	0.095	95.7
EfficientNet-B0	0.081	96.2
Ours	**0.062**	**98.6**

between morphologically similar organism pairs, indicating that the model has learned biologically meaningful feature representations while struggling with subtle inter-class distinctions. Some selected misclassification samples are shown in Figure 7.

The corals-fish confusion pattern (2 corals misclassified as fish, 4 fish misclassified as corals) represents a particularly interesting case of ecological context interference. This bidirectional misclassification stems from several technical factors, which is the co-occurrence of fish and corals in natural reef environments creates complex visual scenes where spatial attention mechanisms may focus on background coral structures when classifying fish, or conversely, incorporate fish-like color patterns present in certain coral species. The asymmetric nature of this confusion (more fish→corals than corals→fish) suggests that coral features are more visually dominant in mixed scenes, potentially due to their larger spatial footprint and more consistent appearance compared to highly mobile fish subjects.

The echinus-holothurian confusion (2 misclassifications in each direction) highlights challenges in discriminating between marine invertebrates with similar gross morphology. Sea urchins (echinus) and sea cucumbers (holothurian) share several confounding visual characteristics, which is both exhibit roughly spherical to elongated body shapes, similar size ranges, and comparable surface textures at certain image resolutions. The bidirectional nature of this confusion indicates that the model's feature extraction mechanism struggles to consistently identify discriminative morphological features such as spine patterns (echinus) versus smooth surface textures (holothurian). This suggests that the current patch-based attention mechanism may be insufficiently fine-grained to capture critical taxonomic differences at the species level.

The starfish-holothurian confusion pattern (4 starfish misclassified as holothurian, 3 holothurian misclassified as starfish) reveals another systematic challenge in benthic organism classification. This confusion likely arises from pose-dependent feature variation, which is starfish photographed from certain angles or with arms folded may present elongated profiles similar to holothurians, while sea cucumbers in contracted states may appear more compact and starfish-like. The slight asymmetry (more starfish→holothurian errors) suggests that starfish features are more variable across different poses and orientations, making them more susceptible to misclassification when appearing in atypical configurations.

From a technical perspective, these misclassification patterns indicate several areas for architectural improvement. The vision transformer component appears to rely heavily on global shape and color features while potentially underutilizing local texture and fine-grained morphological details that are crucial for marine taxonomy. The temporal

fusion mechanism, while effective for capturing movement patterns in video sequences, may not be optimally configured for distinguishing between organisms with similar locomotion characteristics or static poses. Additionally, the current attention mechanism may be insufficient for handling the extreme pose variations and partial occlusions common in underwater imagery.

The misclassification analysis also reveals the model's relative robustness to challenging imaging conditions. The low confusion rates with visually distinct classes (turtle, diver) demonstrate effective learning of distinctive shape and texture patterns, while the concentrated confusion among morphologically similar classes suggests that errors occur primarily at biologically reasonable boundaries. This pattern indicates that the model has successfully learned hierarchical feature representations that align with marine biological taxonomy, failing primarily at fine-grained distinctions that even human experts might find challenging without additional contextual information or higher resolution imagery.

5 Conclusion

This study introduced the PVT Fusion Mamba architecture, a dual-encoder framework designed to overcome the inherent uncertainties of underwater imaging. By combining EfficientNet-B0 and PVT-v2-B2 with a hierarchical fusion neck and a Mamba-based classification head, the system functions as an adaptive control mechanism. This design effectively integrates multi-scale features while modeling long-range dependencies with linear complexity, enabling stable performance despite challenges like turbidity, blur, and varied organism poses.

Experimental results across 27 hyperparameter configurations confirmed the model's high accuracy and stability. The optimal setting ($C_1 = 96, C_2 = 288, C_3 = 192$) achieved a validation accuracy of 98.6% and a validation loss of 0.062. Comparative studies further demonstrated that this architecture outperforms established models like ResNet50 and YOLOv8, particularly in its ability to converge quickly and maintain a high signal-to-noise ratio in complex reef environments. Analysis of the confusion matrix and hardware performance metrics highlighted the model's practical utility. While classification errors were primarily limited to morphologically similar species, the overall system demonstrated excellent per-class precision. Furthermore, the model's efficient design and scalability across CPU cores make it well-suited for deployment in real-world autonomous underwater vehicles (AUVs) and ecological monitoring systems.

Future research will focus on enhancing fine-grained recognition to better distinguish between closely related species and further optimizing the selective memory mechanism for real-time video stream processing. Ultimately, the PVT Fusion Mamba provides a balanced and robust foundation for advancing marine biological research and underwater environmental monitoring.

Acknowledgement

The authors gratefully acknowledge financial support from the Flagship FMIPA 2025 Grant, Universitas Gadjah Mada (Grant Number: 3841/UN1/FMIPA.1.3/KP/PT.01.00/2025). This work also partially supported by the Department of Computer Science and Electronics, Universitas Gadjah Mada under the publication Funding Year 2026.

References

[1] S. Roy, P. Ghosh, T. Goto, and M. Sen, "OBDTL: Under Water Object Classification Using Transfer Learning," *2024 4th International Conference on Computer, Communication, Control Information Technology (C3IT)*, pp. 1–6, 9 2024. [Online]. Available: https://doi.org/10.1109/c3it60531.2024.10829461

[2] P. Saravanan and K. Vadivazhagan, "RELIABLE UNDERWATER IMAGE CLASSIFICATION WITH ENHANCED CNN MODELS USING QUOKKA OPTIMIZATION," *Journal of Theoretical and Applied Information Technology*, vol. 103, pp. 650–671, 2025.

[3] A. Nunes, A. R. Gaspar, and A. Matos, "Critical object recognition in underwater environment," *OCEANS 2019 - Marseille*, 6 2019. [Online]. Available: https://doi.org/10.1109/oceanse.2019.8867360

[4] R. A. Dakhil and A. R. H. Khayeat, "Review on Deep Learning Techniques for Underwater Object Detection," *3rd International Conference on Data Science and Machine Learning (DSML 2022)*, pp. 49–63, 9 2022. [Online]. Available: https://doi.org/10.5121/csit.2022.121505

[5] M. J. Er, J. Chen, Y. Zhang, and W. Gao, "Research challenges, recent advances, and popular datasets in Deep Learning-Based Underwater Marine Object Detection: a review," *Sensors*, vol. 23, no. 4, p. 1990, 2 2023. [Online]. Available: https://doi.org/10.3390/s23041990

[6] A. Jesus, C. Zito, C. Tortorici, E. Roura, and G. De Masi, "Underwater Object Classification and Detection: first results and open challenges," *OCEANS 2022 - Chennai*, pp. 1–6, 2 2022.

[7] Z. Gao, Y. Shi, and S. Li, "Self-attention and long-range relationship capture network for underwater object detection," *Journal of King Saud University - Computer and Information Sciences*, vol. 36, no. 2, p. 101971, 2 2024. [Online]. Available: https://doi.org/10.1016/j.jksuci.2024.101971

[8] K. Liu, L. Peng, and S. Tang, "Underwater Object Detection Using TC-YOLO with Attention Mechanisms," *Sensors*, vol. 23, no. 5, p. 2567, 2 2023. [Online]. Available: https://doi.org/10.3390/s23052567

[9] J. Lei, H. Wang, Z. Lei, J. Li, and S. Rong, "CNN–Transformer Hybrid Architecture for underwater sonar image segmentation," *Remote Sensing*, vol. 17, no. 4, p. 707, 2 2025. [Online]. Available: https://doi.org/10.3390/rs17040707

[10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 6 2016. [Online]. Available: https://doi.org/10.1109/cvpr.2016.90

[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 5 2017. [Online]. Available: https://doi.org/10.1145/3065386

[12] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 4 2015. [Online]. Available: https://doi.org/10.1007/s11263-015-0816-y

[13] K. Simonyan and A. Zisserman, "Very deep convolutional networks for Large-Scale image recognition," *3rd International Conference on Learning Representations, ICLR 2015*, 9 2014. [Online]. Available: https://arxiv.org/abs/1409.1556v6

[14] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–9, 6 2015. [Online]. Available: https://doi.org/10.1109/cvpr.2015.7298594

[15] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2818–2826, 6 2016. [Online]. Available: https://doi.org/10.1109/cvpr.2016.308

[16] E. Q. Nuranti, N. S. Intizhami, M. I. S. Tassakka, I. S. Areni, O. I. A. Ghozy, and M. R. Jefri, "Multi-Head attention in residual networks to improve coral reef structure classification," *JOIV International Journal on Informatics Visualization*, vol. 8, no. 2, p. 700, 5 2024. [Online]. Available: http://dx.doi.org/10.62527/joiv.8.2.2392

[17] Z. Zhou, X. Yang, H. Ji, and Z. Zhu, "Improving the classification accuracy of fishes and invertebrates using residual convolutional neural networks," *ICES Journal of Marine Science*, vol. 80, no. 5, pp. 1256–1266, 4 2023. [Online]. Available: https://doi.org/10.1093/icesjms/fsad041

[18] A. Li, Y. Song, J. Dao, and C. Yang, "Enhancing underwater images via Deep Learning: A Comparative study of VGG19 and RESNET50-Based Approaches," *arXiv (Cornell University)*, 8 2025. [Online]. Available: https://doi.org/10.48550/arxiv.2508.17397

[19] X. Cao, R. Togneri, X. Zhang, and Y. Yu, "Convolutional neural network with Second-Order pooling for underwater target classification," *IEEE Sensors Journal*, vol. 19, no. 8, pp. 3058–3066, 12 2018. [Online]. Available: https://doi.org/10.1109/jsen.2018.2886368

[20] Z. Wang and L. Wu, "Theoretical analysis of inductive biases in deep convolutional networks," *arXiv (Cornell University)*, 5 2023. [Online]. Available: http://arxiv.org/abs/2305.08404

[21] S. Takahashi, Y. Sakaguchi, N. Kouno, K. Takasawa, K. Ishizu, Y. Akagi, R. Aoyama, N. Teraya, A. Bolatkan, N. Shinkai, H. Machino, K. Kobayashi, K. Asada, M. Komatsu, S. Kaneko, M. Sugiyama, and R. Hamamoto, "Comparison of vision transformers and convolutional neural networks in Medical Image Analysis: a systematic review," *Journal of Medical Systems*, vol. 48, no. 1, p. 84, 9 2024. [Online]. Available: https://doi.org/10.1007/s10916-024-02105-8

[22] M. Naseer, K. Ranasinghe, S. Khan, M. Hayat, F. S. Khan, and M.-H. Yang, "Intriguing properties of vision transformers," in *Proceedings of the 35th International Conference on Neural Information Processing Systems*, ser. NIPS '21. Red Hook, NY, USA: Curran Associates Inc., 2021.

[23] M. Yang, H. Wang, K. Hu, G. Yin, and Z. Wei, "IA-NET: An Inception–Attention-Module-Based network for classifying underwater images from others," *IEEE Journal of Oceanic Engineering*, vol. 47, no. 3, pp. 704–717, 2 2022. [Online]. Available: https://doi.org/10.1109/joe.2021.3126090

[24] S. V. T, "Underwater Image Enhancement and Object Detection with Multi-Color Space Residual Network and Swin-Yolo Fusion," *SSRN Electronic Journal*, 1 2025. [Online]. Available: https://doi.org/10.2139/ssrn.5249972

[25] G. Abirami, S. Nagadevi, J. D. D. Jayaseeli, T. P. Rao, R. S. M. L. Patibandla, R. Aluvalu, and K. Srihari, "An integration of ensemble deep learning with hybrid optimization approaches for effective underwater object detection and classification model," *Scientific Reports*, vol. 15, no. 1, p. 10902, 3 2025.

[Online]. Available: https://doi.org/10.1038/s41598-025-95596-5

[26] S. R. Warrier, D. S. H. Reddy, S. Bada, R. Achampeta, S. Uppapalli, and J. Dontabhaktuni, "On-board classification of underwater images using hybrid classical-quantum CNN-based method," *Quantum Machine Intelligence*, vol. 6, no. 2, 10 2024. [Online]. Available: https://doi.org/10.1007/s42484-024-00206-8

[27] G. Yu, Y. Wu, J. Xiao, and Y. Cao, "A Novel Pyramid Network with Feature Fusion and Disentanglement for Object Detection," *Computational Intelligence and Neuroscience*, vol. 2021, no. 1, p. 6685954, 1 2021. [Online]. Available: https://doi.org/10.1155/2021/6685954

[28] D. Guo, Z. Wu, J. Feng, and T. Zou, "Multi-scale semantic enhancement network for object detection," *Scientific Reports*, vol. 13, no. 1, p. 7178, 5 2023. [Online]. Available: https://doi.org/10.1038/s41598-023-34277-7

[29] T. Van Quyen and M. Y. Kim, "Feature pyramid network with multi-scale prediction fusion for real-time semantic segmentation," *Neurocomputing*, vol. 519, pp. 104–113, 11 2022. [Online]. Available: https://doi.org/10.1016/j.neucom.2022.11.062

[30] H. Chen, Y. Wang, L. Wu, H. Hu, J. Yan, H. Xu, and G. Lei, "Mamba-convolution hybrid network for underwater image enhancement," *Scientific Reports*, vol. 15, no. 1, p. 31975, 8 2025. [Online]. Available: https://doi.org/10.1038/s41598-025-15404-y

[31] X. Luan, J. Wang, S. Rong, H. Yu, and B. He, "Degradation information-guided Mamba for underwater image enhancement," *Optics Laser Technology*, vol. 192, p. 113542, 7 2025. [Online]. Available: https://doi.org/10.1016/j.optlastec.2025.113542

[32] R. Cong, Z. Yu, H. Fang, H. Sun, and S. Kwong, "UIS-Mamba: Exploring Mamba for Underwater Instance Segmentation via Dynamic Tree Scan and Hidden State Weaken," *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 343–352, 10 2025. [Online]. Available: https://doi.org/10.1145/3746027.3755131

[33] M. Tan and Q. V. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, ser. Proceedings of Machine Learning Research, K. Chaudhuri and R. Salakhutdinov, Eds., vol. 97. PMLR, 2019, pp. 6105–6114. [Online]. Available: http://proceedings.mlr.press/v97/tan19a.html

[34] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao, "Pvt v2: Improved baselines with pyramid vision transformer," *Computational Visual Media*, vol. 8, no. 3, pp. 415–424, 2022.

[35] C. Fu, R. Liu, X. Fan, P. Chen, H. Fu, W. Yuan, M. Zhu, and Z. Luo, "Rethinking general underwater object detection: Datasets, challenges, and solutions," *Neurocomputing*, vol. 517, pp. 243–256, 2023.