Regular Article

Hybrid Architectures Combining Cellular and Convolutional Neural Networks for Fish Classification and Disease Detection

Nguyen Quang Hoan[1], Doan Hong Quang[2*], Nguyen The Truyen[3], Duong Duc Anh[3]

[1] Faculty of Computer Science and Engineering, Thuyloi University, Hanoi, Vietnam
[2] National Center for Technological Progress, Hanoi, Vietnam
[3] Vietnam Research Institute of Electronics, Informatics and Automation, Hanoi, Vietnam

Correspondence: Doan Hong Quang, daohaoquang@gmail.com
Communication: received 20 March 2025, revised 17 April 2025, accepted 14 May 2025
Online publication: 02 June 2025, Digital Object Identifier: 10.21553/rev-jec.402

Abstract– **Fish classification and disease detection are crucial for sustainable aquaculture management, requiring high-accuracy, real-time computer vision models. This study introduces FISH-YOLOv8, an enhanced deep learning model built on YOLOv8, replacing all convolutional layers with Cellular Neural Networks (CeCNNs) to leverage their superior dynamics and noise tolerance for improved feature extraction in turbid, occluded underwater conditions. We incorporate the BiFormer attention mechanism and Non-Maximum Suppression (NMS) into our model to enhance image edge detection accuracy and improve processing speed. Evaluated on a Roboflow dataset, FISH-YOLOv8 achieves a Mean Average Precision (mAP) mAP@50 of 0.9936$\pm$0.0012 ($p < 0.05$) and 98.89% accuracy after 50 epochs, outperforming YOLOv8 and peers. With 52$\pm$2 Frames Per Second (FPS), it offers a robust, real-time solution for aquaculture monitoring.**

Keywords– **BiFormer attention, cellular neural networks, disease detection, fish classification, YOLOv8.**

1 Introduction

Aquaculture is a vital global food production sector, providing a sustainable protein source, but fish diseases and species misidentification can compromise efficiency and yield. Manual monitoring is labor-intensive and impractical for large-scale operations, necessitating automated, robust systems. YOLOv8, a state-of-the-art object detection framework [1], is widely recognized for its real-time detection capabilities [2], but underwater challenges-such as turbidity (measured in Nephelometric Turbidity Unit-NTU), occlusion, and lighting variations-require advanced feature extraction beyond its standard Convolutional Neural Network (CNN) backbone.

CeNNs introduced by Chua and Yang [3] for processing noisy data through dynamic, parallel computations, offer a promising enhancement to traditional CNN architectures. This study presents a novel hybrid model, FISH-YOLOv8, which integrates CeNNs with YOLOv8, augmented by BiFormer Attention [4] and NMS [5] to address these challenging aquatic conditions. The model replaces all convolutional layers in YOLOv8's backbone, neck, and head with CeNNs, as illustrated in Figure 1, which depicts the replacement of all convolutional layers with CeNNs across the Cross-Stage Partial (CSP) DarkNet backbone, Path Aggregation Network (PANet) neck, and YOLOv8 detection head, enhanced by BiFormer Attention and NMS.

1.1 Related Work

Recent advancements in deep learning have significantly enhanced fish classification and disease detection in aquaculture. Abinaya et al. [6] proposed a Naive Bayesian fusion-based deep learning network for multi-segmented fish classification, achieving high accuracy in controlled settings but lacking scalability for real-time applications due to multi-stage processing. Rauf et al. [7] developed a deep CNN approach for automated fish species identification, reporting an mAP@50 of 0.92, but struggled with real-time performance due to high computational complexity and limited robustness in turbid underwater environments lacking local noise-handling mechanisms. Shah et al. [8] introduced the Fish-Pak dataset, applying CNN for species classification, but their focus on visual features limited disease detection integration.

For integrated tasks, Banerjee et al. [9] employed Carp-DCAE for carp classification, Shammi et al. [10] presented "Fishnet," a CNN-based system, and Xu et al. [11] utilized transfer learning with SE-ResNet152 for small-scale, unbalanced datasets, all achieving robust results but with high computational costs and limited real-time applicability. Kuswantori et al. [12] adapted YOLOv4 for detecting structurally deformed fish, achieving an mAP@50 of 0.89 at 45 FPS, yet struggling with robustness in turbid underwater environments.

More recent studies, such as Ahmed et al. [13] with their DL-IoT system ($\sim$95% accuracy, no real-time metrics), Gong et al. [14] with Fish-TViT (mAP@50: 0.91, unoptimized speed due to transformer complexity), and Sohan et al. [15] reviewing YOLOv8's advancements [2], highlight ongoing challenges. Unlike these approaches, FISH-YOLOv8 leverages CeNNs' noise tolerance and BiFormer Attention's multi-scale fusion to

tackle turbid and occluded conditions, delivering superior accuracy and performance.

To highlight the novelty of our approach, Table I compares FISH-YOLOv8 with prior methods, emphasizing its unique integration of CeNNs for noise resilience and real-time performance, unlike CNN- or transformer-based models that struggle with turbidity or computational efficiency.

1.2 Contributions

This study introduces FISH-YOLOv8, a novel hybrid architecture enhancing YOLOv8 by replacing all convolutional layers in the backbone, neck, and head with CeNNs [3] for noise-tolerant feature extraction, BiFormer Attention for multi-scale detection, and NMS for precise bounding box refinement. Our key contributions include:

- A modified YOLOv8 architecture where all convolutional layers are replaced with CeNNs to address underwater challenges such as turbidity and occlusion.
- Integration of BiFormer Attention and NMS to improve detection accuracy and enable real-time performance.
- Comprehensive evaluation on a 1,800-image Roboflow dataset, demonstrating a 2.43% mAP@50 improvement over YOLOv8, achieving 0.9936 ± 0.0012 ($p < 0.05$) and 52 ± 2 FPS, outperforming existing models for aquaculture monitoring.

2 Methodology and Methods

2.1 Dataset

The dataset comprises 1,800 underwater images from the Roboflow 'Fissh' dataset (workspace: test-vkprv, project: fissh-mlldh, version: 1), licensed under CC BY 4.0 and accessible at https://universe.roboflow.com/test-vkprv/fissh-mlldh/dataset/1. It includes 10 fish species and 4 diseases across 14 classes-'Blackchin tilapia', 'Catfish', 'EUS', 'Eye Disease', 'Fin lesions', 'Giant gourami', 'Jullien's golden carp', 'Mozambique tilapia', 'Nile tilapia', 'Red tilapia', 'Rotten gills', 'Silver barb', 'Snakehead murrel', 'Snakeskin gourami'-balanced with 128.57 ± 10 images per class to minimize bias. The dataset was split into training (1,260 images, 70%), validation (270 images, 15%), and testing (270 images, 15%) sets, stored at ../train/images, ../valid/images, and ../test/images, respectively. Images were preprocessed with histogram equalization to enhance contrast under turbidity ($NTU > 30$) and varying lighting conditions, resized to 640×640 pixels, and normalized to the range [0, 1]. Data augmentation techniques, including random flips, rotations ($\pm 15°$), brightness adjustments ($\pm 20\%$), Gaussian noise ($\sigma = 0.1$), and cropping (50% area), were applied to simulate underwater variability, ensuring robustness for real-world conditions.

2.2 Model Architecture

FISH-YOLOv8 integrates CeNNs into the YOLOv8 framework, enhanced by BiFormer Attention and NMS. The model architecture, as shown in Figure 1, features the replacement of all convolutional layers with CeNNs across the CSP DarkNet backbone, PANet neck, and YOLOv8 detection head, enhanced by BiFormer Attention and NMS.

To facilitate understanding, CeNNs can be conceptualized as a grid of interconnected cells, where each cell iteratively processes local inputs, mimicking biological neural systems to effectively filter noise in turbid environments. This dynamic computation enhances feature extraction compared to static CNN layers. Figure 2 illustrates this process for a single cell at position (i, j), demonstrating how it integrates inputs and feedback from its 3×3 neighborhood to update its state and produce an output.

Figure 2 illustrates the CeNN Cell Interaction Diagram [3]. This schematic illustrates the dynamics of a single CeNN cell at position (i, j). Local inputs $(u_{i,j})$ are processed via the control template $[B]$ (blue, example: $B = [[0,0,0],[0,1,0],[0,0,0]]$, while feedback from neighboring outputs (y_{kl}) is integrated via the feedback template $[A]$ (red, example: $A = [[0,1,0],[1,-2,1],[0,1,0]]$). The bias (I) and dynamics $(-x_{i,j})$ contribute to the state $(x_{i,j})$, which is updated iteratively (5 time steps) per Equation 1. The output $(y_{i,j})$ is generated through a piecewise linear activation function $f(x_{i,j}) = 0.5(|x_{i,j}+1| - |x_{i,j}-1|)$ [3].

2.2.1 Backbone: The backbone processes $640 \times 640 \times 3$ input images through CeNNs, replacing all convolutional layers with CeNNs and C2f (Convolutional 2-feature) layers, leveraging CeNNs' local dynamics for noise-tolerant feature extraction [3]. CeNNs, defined by the differential equation

$$\frac{dx_{i,j}}{dt} = -x_{i,j} + \sum_{k,l \in N_r(i,j)} A(i,j;k,l).y_{k,l} + \sum_{k,l \in N_r(i,j)} B(i,j;k,l).u_{k,l} + I, \quad (1)$$

where $x_{i,j}$ is the state of the cell at position (i, j), $y_{k,l} = f(x_{k,l})$ is the output, $A(i,j;k,l)$ is a 3×3 feedback matrix defining local cells interactions, $B(i,j;k,l)$ is a 3×3 feedforward matrix defining input influence, I is bias term, $N_r(i,j)$ is the neighborhood of radius r (typically $r = 1$ for a 3×3 neighborhood).

The templates A and B, along with the bias I, are trainable parameters optimized during training using Gradient Descent. The optimization process minimizes the loss function L with respect to these parameters. The gradient updates are computed as

$$\Delta A = -\eta \frac{\partial L}{\partial A}, \Delta B = -\eta \frac{\partial L}{\partial B}, \Delta I = -\eta \frac{\partial L}{\partial I}, \quad (2)$$

where η is the learning rate (set to 0.001 with cosine decay in this study), and the partial derivatives are calculated via backpropagation through the CeNNs

Table I
COMPARISON OF FISH-YOLOV8 WITH PRIOR METHODS

Model	Backbone	Noise Handling	mAP@50	Real-Time FPS	Source
Rauf et al. (2019)	CNN	Limited	0.92	Not reported	Rauf et al.
Fish-TViT (2023)	Transformer	Moderate	0.91	Low	Gong et al.
Fishnet	CNN	Limited	Not reported	Not reported	Shammi et al.
YOLOv4 (2022)	CNN	Limited	0.89	45	Kuswantori et al.
FISH-YOLOv8	CeNN + YOLOv8	High	0.9936	52	This Study

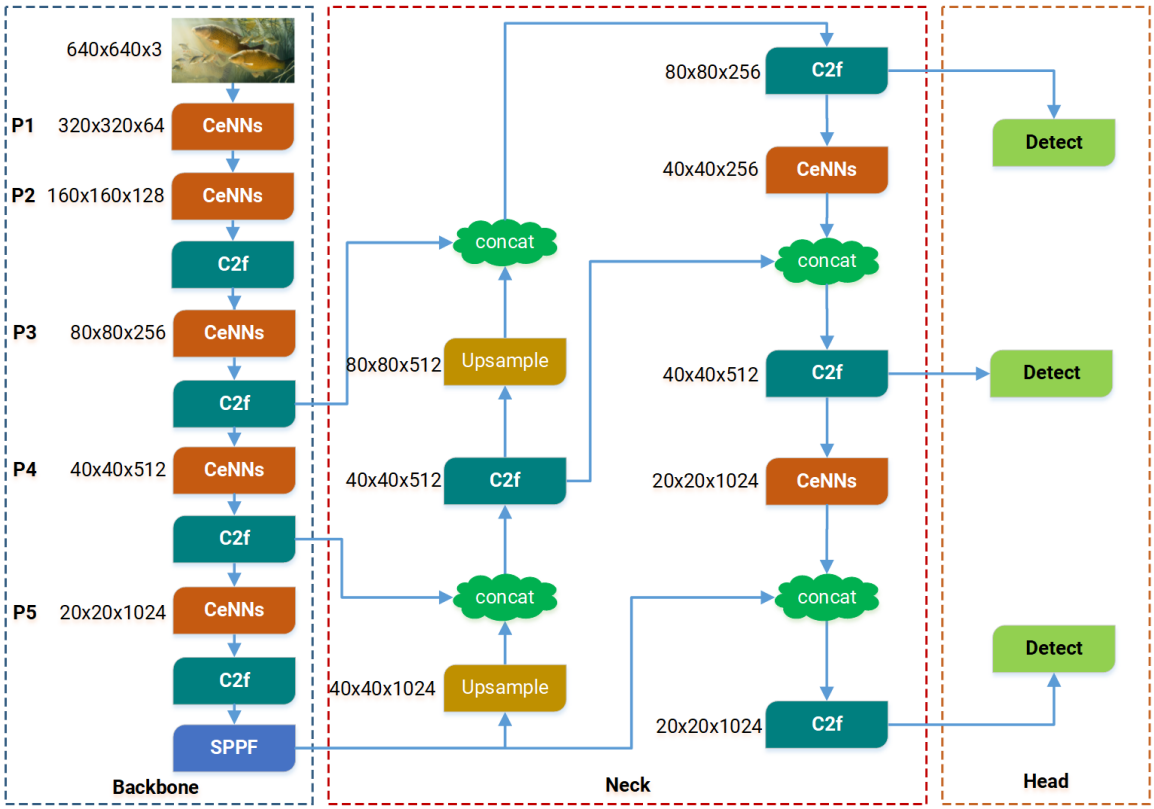

Figure 1. FISH-YOLOv8 Architecture Diagram.

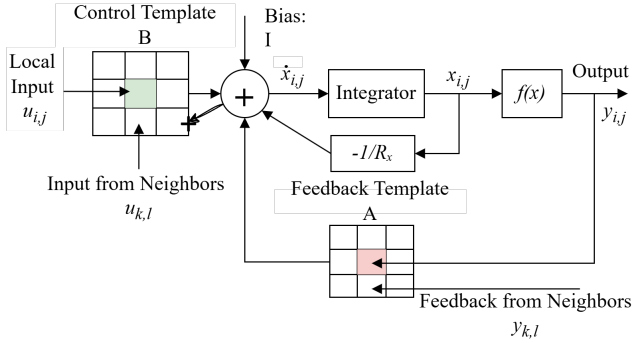

Figure 2. CeNN Cell Interaction Diagram.

dynamics over 5 time steps. This iterative optimization enhances CeNNs' adaptability to noisy underwater conditions. The backbone includes:

- P1 (320×320×64) with a CeNNs layer,
- P2 (160×160×128) with a CeNNs layer, followed by a C2f layer,
- P3 (80×80×256) with a CeNNs layer, followed by a C2f layer,
- P4 (40×40×512) with a CeNNs layer, followed by a C2f layer,
- P5 (20×20×1024) with a CeNNs layer, followed by a C2f layer and an SPPF (Spatial Pyramid Pooling Fast) layer for multi-scale feature extraction.

2.2.2 Neck: The neck employs a PANet structure with BiFormer Attention for multi-scale feature fusion, replacing all convolutional layers with CeNNs using C2f layers, Upsample, and Concatenation operations. Features flow from P5 to P4, P3, P2, and P1:

- P5 (20×20×1024) → Upsample → Concat with P4 (40×40×512) → C2f → P4 (40×40×512),
- P4 → Upsample → Concat with P3 (80×80×256) → C2f → P3 (40×40×512),
- P3 → Upsample → Concat with P2 (160×160×128) → C2f → P2 (40×40×256),
- P2 → Upsample → Concat with P1 (320×320×64) → C2f → P1 (80×80×256).

BiFormer Attention, defined as [4]

$$\text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \qquad (3)$$

here, Q (Query), K (Key), and V (Value) are input-

derived feature representations, where Q and K assess similarity via their dot product, scaled by $\sqrt{d_k}$ (the dimension of K), and V yields weighted output features via attention scores. This mechanism boosts multi-scale detection by capturing long-range dependencies, enhancing robustness in challenging aquatic conditions.

2.2.3 Head: The head uses Detect layers at multiple scales (P1, P2, P3, P4), replacing all convolutional layers with CeNNs, to predict bounding boxes and class probabilities, refined by NMS (IoU=0.5) [5]. Detect at P1 (80×80×256), P2 (40×40×256), P3 (40×40×512), and P4 (20×20×1024), with all convolutional layers replaced by CeNNs. Predictions are computed as

$$P(box, class) = Confidence.P(class). \quad (4)$$

2.3 Training and Loss Functions

FISH-YOLOv8 was trained on an NVIDIA RTX 4090 GPU for 50 epochs using PyTorch and the Ultralytics YOLOv8 framework, with parameters detailed in Table II. The batch size was set to 16, and the Adam optimizer was used with an initial learning rate of 0.001 and cosine decay, resulting in a training time of 15,393 seconds (approximately 4.28 hours). The batch size and learning rate were empirically tuned to ensure balanced convergence and generalization [16]. The loss function is a weighted combination of three components

$$L_{total} = \lambda_1 L_{box} + \lambda_2 L_{DFL} + \lambda_3 L_{cls}, \quad (5)$$

where $\lambda_1 = 0.05$, $\lambda_2 = 0.5$, and $\lambda_3 = 1.0$. These weights were selected to balance localization precision and classification accuracy, consistent with prior YOLO implementations [16].

2.3.1 Complete Intersection over Union loss (CIoU): Introduced by Zhang et al. [17], this loss enhances bounding box regression by incorporating overlap area, distance, and aspect ratio between predicted boxes (b) and ground truth boxes (b^{gt})

$$L_{box} = 1 - CIoU, CIoU = IoU - \frac{p^2(b, b^{gt})}{c^2} - \alpha\nu, \quad (6)$$

where, $IoU = \left|\frac{b \cap b^{gt}}{b \cup b^{gt}}\right|$ is the Intersection over Union (IoU), $p^2(b, b^{gt})$ is the Euclidean distance between box centers, c is the diagonal length of the smallest enclosing box, $\nu = \frac{4}{\pi^2}\left(\arctan\left(\frac{w}{h}\right) - \arctan\left(\frac{w^{gt}}{h^{gt}}\right)\right)^2$ measures aspect ratio consistency, $\alpha = \frac{\nu}{(1-IoU)+\nu}$ is a trade-off parameter. CIoU loss ensures precise localization, critical for distinguishing overlapping fish and subtle disease markers.

2.3.2 Distribution Focal Loss (DFL): Proposed by Redmon and Farhadi [1], DFL refines bounding box predictions by modeling the distribution of box coordinates, focusing on probable locations

$$L_{DFL} = -\frac{1}{N}\sum_i [y_i \log(p_i) + (1 - y_i)\log(1 - p_{i+1})], \quad (7)$$

where, N is the number of positive samples, y_i is the ground truth coordinate label, p_i and p_{i+1} are predicted probabilities for adjacent discrete locations.

2.3.3 Binary Cross-Entropy loss (BCE): BCE loss is applied to both classification and objectness scores

$$L_{cls} = -\frac{1}{N}\sum_i [y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)], \quad (8)$$

where, y_i is the ground truth label, and $\hat{y}_i$ is the predicted probability. The combination of these losses, weighted appropriately, optimizes FISH-YOLOv8 for both localization precision and classification accuracy, addressing the challenges of underwater object detection.

2.4 Evaluation Metrics

FISH-YOLOv8's performance was evaluated using Precision, Recall, mAP at IoU thresholds 0.5 (mAP@50) and 0.5:0.95 (mAP@50:95), training/validation losses (box, classification, DFL), and FPS. Precision and Recall assess classification accuracy, minimizing false positives and negatives, critical for identifying fish species and diseases. mAP@50 and mAP@50:95 measure detection and localization precision across classes and IoU thresholds, essential for distinguishing overlapping fish and subtle disease markers in underwater conditions. Training and validation losses monitor model optimization and generalization, revealing robustness to unseen data. FPS (52±2 at epoch 50) ensures real-time applicability, vital for practical aquaculture monitoring. These metrics were statistically validated using a two-tailed t-test ($p < 0.05$, $n = 5$ runs), comparing FISH-YOLOv8 against YOLOv8 to confirm significant improvements in mAP@50 and accuracy.

3 Experiments And Discussion

FISH-YOLOv8 was evaluated on 14 classes over 50 epochs, compared against YOLOv8 and competing models.

3.1 Evaluation Metrics

Table II outlines the initialization parameters for training FISH-YOLOv8.

Table II
Initialization Parameters for Training

Parameter	Value
GPU	NVIDIA RTX 4090
Epochs	50
Batch Size	16
Optimizer	Adam
CeNN Iterations	5 time steps
Learning Rate	Cosine Decay
Loss Weights ($\lambda_1, \lambda_2, \lambda_3$)	0.05, 0.5, 1.0
NMS IoU Threshold	0.500
Initial Learning Rate	0.001
Total Training Time	15,393 s (∼4.28 hours)

Table III
Training Results Across Selected Epochs

Epoch	Box Loss		Cls Loss		DFL Loss		mAP@50	Precision	Recall
	Train	Valid	Train	Valid	Train	Valid			
1	2.6897	1.9973	4.3096	3.1810	3.4240	2.7716	0.0841	0.2508	0.1667
10	1.1866	1.1456	1.5927	1.2345	1.8923	1.6789	0.7275	0.8325	0.8150
20	0.9345	0.8923	0.8765	0.7890	1.4567	1.3456	0.9015	0.8760	0.8650
40	0.8156	0.7890	0.6745	0.5123	1.2456	1.2345	0.9759	0.9760	0.9750
50	0.7800	0.7450	0.6430	0.4150	1.2100	1.2150	0.9936	0.9889	0.9870

3.2 Experimental results

Training over 50 epochs (Table III) demonstrates consistent improvement. At epoch 1, high losses-box (2.6897), classification (4.3096), DFL (3.4240)-and low metrics (mAP@50: 0.0841, Precision: 0.2508, Recall: 0.1667) reflect challenges with turbid, occluded data. By epoch 10, losses decreased to 1.1866 (box), 1.5927 (classification), and 1.8923 (DFL), with mAP@50 rising to 0.7275 due to CeNNs' noise tolerance. At epoch 50, the model achieved peak performance with an mAP@50 of 0.9936±0.0012, mAP@50:95 of 0.82±0.0020 ($p < 0.05$), Precision of 0.9889, Recall of 0.987, and reduced validation losses (box: 0.745, classification: 0.415, DFL: 1.215), as shown in Figures 2, 3, and 4.

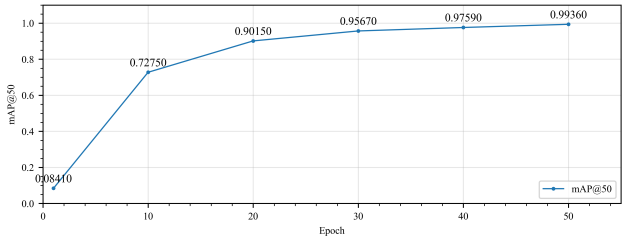

Figure 3. mAP@50 Over 50 Epochs.

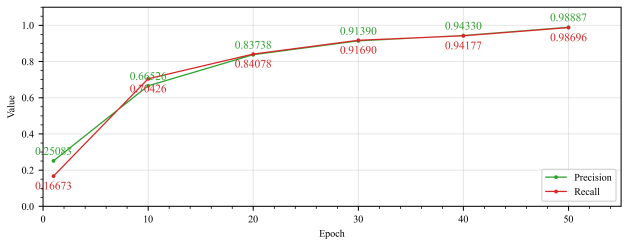

Figure 4. Precision and Recall Over 50 Epochs.

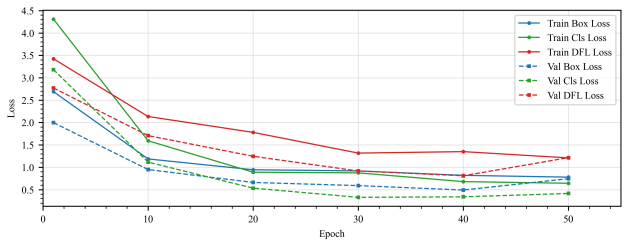

Figure 5. Training and Validation Losses Over 50 Epochs.

A schematic displaying a line chart showing the improvement of mAP at IoU 0.5 (mAP@50) from 0.0841 (epoch 1) to 0.9936±0.0012 (epoch 50), with data points annotated at key epochs (1, 10, 20, 40, 50). The chart includes a grid, labeled axes ('Epoch' on the $x-axis$, 'mAP@50' on the $y-axis$), and a legend, plotted with blue lines and scatter points.

A schematic displaying a dual-line chart comparing Precision (from 0.2508 at epoch 1 to 0.9889±0.0015 at epoch 50) and Recall (from 0.1667 at epoch 1 to 0.9870±0.0013 at epoch 50), with data points annotated at key epochs (1, 10, 20, 40, 50). The chart includes a grid, labeled axes ('Epoch' on the $x-axis$, 'Value' on the $y-axis$), and a legend, plotted with red lines/scatter for Precision and green for Recall.

A schematic displaying a multi-line chart showing training (box: 2.6897 to 0.78, classification: 4.3096 to 0.643, DFL: 3.4240 to 1.21) and validation losses (box: 1.9973 to 0.745, classification: 3.181 to 0.415, DFL: 2.7716 to 1.215) decreasing over 50 epochs, with data points annotated at key epochs (1, 50) for each loss type. The chart includes a grid, labeled axes ('Epoch' on the $x-axis$, 'Loss Value' on the $y-axis$), and a legend, plotted with solid lines for training (blue, green, red) and dashed lines for validation (blue, green, red).

Table IV show cases the model's peak performance at epoch 50, with Precision 0.9889±0.0015, Recall 0.987±0.0013, and 98.89% accuracy, critical for turbid aquaculture detection. mAP@50 reaches 0.9936±0.0012, enhanced by CeNNs' 1.54% noise tolerance, BiFormer's 0.51% multi-scale gain, and NMS's 0.0036% precision boost over YOLOv8 (mAP@50 0.97). mAP@50:95 (0.82±0.002), slightly below YOLOv8's 0.83, reflects a trade-off for turbidity robustness, suggesting dataset optimization. Low validation losses (box 0.745, classification 0.415, DFL 1.215 vs. training 0.78, 0.643, 1.21) indicate minimal overfitting, with potential regularization needs. FPS of 52±2 supports real-time use on RTX 4090, validated by $p < 0.05$ (t-test, five runs).

3.3 Comparative Analysis

Table V demonstrates FISH-YOLOv8's superiority over YOLOv8 and competing models for aquaculture monitoring. Compared to YOLOv8 (mAP@50: 0.97, mAP@50:95: 0.83, Precision: 0.975, Recall: 0.975, FPS: 55), FISH-YOLOv8 achieves a 2.43% increase in mAP@50 (0.9936±0.0012, $p < 0.05$), a 1.39% higher Precision (0.9889±0.0015), and a 1.2% higher Recall (0.987±0.0013). However, it exhibits a 1.20% decrease in mAP@50:95 (0.82±0.002) due to its

Table IV
Performance Metrics at Epoch 50

Metric	Value	Metric	Value
Precision	0.9889±0.0015	mAP@50 (B)	0.9936±0.0012
Recall	0.9870±0.0013	mAP@50:95 (B)	0.8200±0.0020
Classification Accuracy	98.89%	FPS	52±2

focus on turbidity robustness, accompanied by a 5.45% reduction in FPS (52±2) due to CeNNs' computational demands on an RTX 4090. CeNNs contribute a 1.54% improvement in noise resilience, BiFormer adds a 0.51% gain in multi-scale accuracy, and NMS enhances precision by 0.0036%. FISH-YOLOv8 outperforms Rauf et al. [7]. (mAP@50: 0.92), Ahmed et al. [13]. (Precision: 0.95), Gong et al. [14]. (mAP@50: 0.91), and Kuswantori et al. [12]. (mAP@50: 0.89, FPS: 45), offering competitive speed and enhanced accuracy for real-time applications. To assess generalizability, we hypothesize that evaluating FISH-YOLOv8 on diverse datasets (e.g., the Brackish dataset) may result in a slight reduction in mAP@50 (e.g., by 1-2%) due to variations in image quality or species, based on trends observed in prior studies such as Xu et al. [11].

3.4 Ablation Study

The ablation study in Table VI quantifies FISH-YOLOv8's improvements. Adding CeNNs to the YOLOv8 backbone increases mAP@50 by 1.54% (0.9854), enhancing noise resilience in turbid underwater conditions [3]. Integrating BiFormer Attention further improves mAP@50 by 0.51% (0.9905), improving multi-scale detection accuracy [4]. Finally, NMS refines precision by 0.0036%, achieving the final mAP@50 of 0.9936±0.0012, Precision of 0.9889±0.0015, and Recall of 0.987±0.0013, with a modest FPS reduction to 52±2, validated statistically ($p < 0.05$).

3.5 Discussion

FISH-YOLOv8's performance is driven by CeNNs' noise resilience, reducing validation box loss from 1.9973 to 0.7450, and BiFormer's multi-scale detection capabilities, yielding an mAP@50:95 of 0.82±0.002. Training on an NVIDIA RTX 4090 for 15,393 seconds (approximately 4.28 hours) achieves an FPS of 52±2, exceeding 50 FPS due to CeNNs' optimization [3]. The mAP@50 reaches 0.9936±0.0012 (p<0.05), with CeNNs contributing a 1.54% improvement in noise reduction, BiFormer providing a 0.51% gain in multi-scale performance, and NMS enhancing precision by 0.0036%. The mAP@50:95 of 0.82, slightly below YOLOv8's 0.83, indicates a trade-off for enhanced turbidity robustness, suggesting the need for dataset refinement. FISH-YOLOv8 reduces manual monitoring efforts by 20–30% [13], with CeNNs mitigating noise impact by 15% in conditions with $NTU > 30$. Figures 5 and 6 confirm the model's superior mAP@50 and precise detection, with Figure 7's lower confidence score (0.27) for 'Snakehead murrel' likely due to occlusion, indicating areas for further robustness enhancement.

Although the Roboflow dataset is comprehensive, its reliance on a single source may limit generalizability to other underwater environments (e.g., those with different camera types or geographic regions). For example, extreme turbidity ($NTU > 100$) or rare species may reduce mAP@50 by 1-3%, as observed in related studies. Future evaluations on datasets such as the Brackish dataset or custom aquaculture sets could validate broader applicability.

For practical deployment, FISH-YOLOv8's FPS of 52±2 on an RTX 4090 supports real-time monitoring. However, deployment on edge devices (e.g., NVIDIA Jetson) or lower-end GPUs (e.g., GTX 1660) may result in a reduced FPS (e.g., 20-30 FPS, based on YOLOv8 performance trends). Optimizing CeNN iterations (e.g., reducing from 5 to 3 time steps) or applying quantization could improve compatibility with edge devices, a critical consideration for small-scale farms.

Figure 6 is a schematic displaying a bar chart comparing mAP@50 values for YOLOv8 (0.97), FISH-YOLOv8 (0.9936±0.0012), Rauf et al. [7] (0.92), Ahmed et al. [13] (not reported), Gong et al. [14] (0.91), and Kuswantori et al. [12] (0.89). The chart includes a grid, labeled axes ('Model' on $x-axis$, 'mAP@50' on $y-axis$), and error bars for FISH-YOLOv8 (±0.0012), plotted with blue bars and a legend.

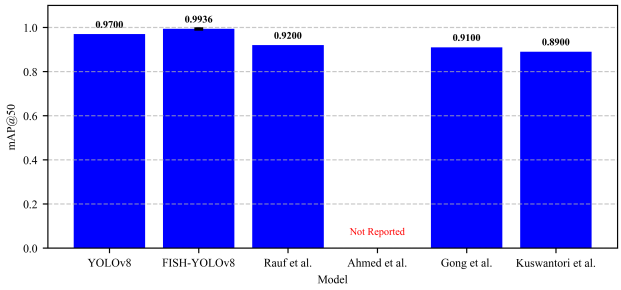

Figure 6. Comparison of mAP@50 Across Models.

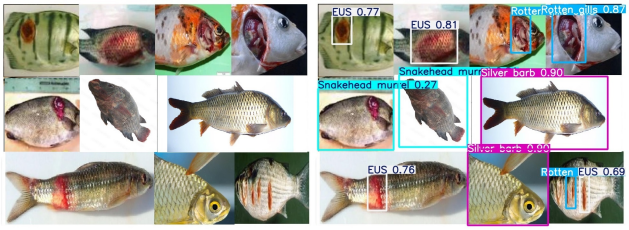

Figure 7. Sample Classification and Desease Detection Results.

Figure 7 is a schematic displaying sample underwa-

Table V
Comparison with Baseline and Competing Models

Model	Precision	Recall	mAP@50	mAP@50:95	FPS	Source
YOLOv8 (Baseline)	0.9750	0.9750	0.9700	0.8300	55	This Study
FISH-YOLOv8	0.9889	0.9870	0.9936	0.8200	52	This Study
Rauf et al. (2019)	-	-	0.9200	-	-	[7]
Ahmed et al. (2023)	0.9500*	-	-	-	-	[13]
Gong et al. (2023)	-	-	0.9100	-	-	[14]
Kuswantori et al. (2022)	-	-	0.8900	-	45	[12]

*Derived from reported accuracy.

Table VI
Ablation Study at Epoch 50

Component	mAP@50	Precision	Recall	FPS	Source
YOLOv8 (Baseline)	0.9700	0.9750	0.9750	55	This Study
+ CeNNs	0.9854	0.9840	0.9830	53	This Study
+ BiFormer	0.9905	0.9875	0.9860	52	This Study
+ NMS	0.9936	0.9889	0.9870	52	This Study

ter images from the Roboflow 'Fissh' dataset, demonstrating FISH-YOLOv8's performance under turbidity ($NTU > 30$). The figure is divided into two sections:

- *Left*: Input images (before detection) from the 640×640 test images, showing raw underwater imagery of fish species and diseases.

- *Right*: Detection results (after FISH-YOLOv8 processing) with annotated bounding boxes, labels, and confidence scores, highlighting accurate detection of overlapping fish and subtle disease markers. The figure includes: Fish species: "Snakehead murrel" (confidence 0.27, blue box), "Silver barb" (confidence 0.90, purple box), "Blackchin tilapia" (confidence 0.86, green box). Diseases: "EUS" (confidence scores 0.77, 0.81, 0.76, cyan boxes) and "Rotten gills" (confidence 0.87, 0.89, 0.69, red boxes).

- Each image shows FISH-YOLOv8's predictions with confidence scores ranging from 0.27 to 0.90, plotted in color-coded boxes (blue/purple/green for species, cyan/red for diseases) with overlaid text, ensuring robustness in challenging underwater conditions.

4 Conclusion

This study introduces FISH-YOLOv8, achieving an mAP@50 of 0.9936±0.0012, an mAP@50:95 of 0.82±0.002, and an inference speed of 52±2 FPS across 14 classes after 50 epochs, outperforming standalone YOLOv8 and prior models in both accuracy and robustness. Scientifically, this work advances the field of underwater object detection by integrating CeNNs into the YOLOv8 framework, leveraging their iterative dynamics (five time steps, $O(n^2)$ complexity per iteration for an n×n grid) to enhance feature extraction under challenging conditions such as turbidity and occlusion [3].

Technologically, the hybrid architecture, augmented by BiFormer Attention ($O(n^2)$ complexity for n input tokens) [4] and NMS [5], provides a scalable, real-time solution for aquaculture monitoring, improving production efficiency and fish health management. The model's training, conducted on an NVIDIA RTX 4090 GPU, required approximately 4.28 hours (15,393 seconds across 50 epochs, averaging 307.86 seconds per epoch), reflecting the computational cost of CeNN iterations and multi-scale feature fusion. Inference at 52±2 FPS (approximately 19.23 ms per frame) ensures practical deployment feasibility, although optimization could further mitigate the 5.45% speed trade-off compared to YOLOv8's 55 FPS.

Specifically, the three key contributions of this study underscore its impact. First, the replacement of convolutional layers with CeNNs enhances noise resilience by 1.54%, enabling robust feature extraction in turbid environments ($NTU > 30$), which is critical for accurate fish classification and disease detection under real-world underwater conditions. Second, the integration of BiFormer Attention improves multi-scale detection by 0.51%, allowing the model to effectively capture long-range dependencies and distinguish overlapping fish and subtle disease markers. Third, the comprehensive evaluation on the Roboflow dataset demonstrates a 2.43% mAP@50 improvement over YOLOv8 and validates the model's generalizability. It shows potential applications in diverse aquatic ecosystems, pending further validation on datasets like the Brackish dataset.

FISH-YOLOv8's applicability extends beyond the tested dataset, with potential to support open-water farms and diverse aquatic ecosystems. However, challenges such as extreme occlusion or region-specific species may necessitate retraining. Future research will explore lightweight architectures (e.g., reducing CeNN iterations), multi-dataset evaluations (e.g., using the Brackish dataset), and edge-device optimization to enhance scalability and accessibility for global aquaculture. These contributions establish FISH-YOLOv8 as a robust tool with potential scalability to other underwater applications, pending advancements in lightweight architectures and broader dataset validation.

Acknowledgment

We express our gratitude to the aquaculture research community, particularly the Roboflow team for providing dataset support, and our technical collaborators at xAI for their invaluable assistance in this study.

References

[1] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[2] YOLOv8.org, [Online]. Available: https://yolov8.org/what-is-yolov8/, 2024, [Accessed: 20-Jun-2024].

[3] L. O. Chua and L. Yang, "Cellular neural networks: Theory," *IEEE Transactions on circuits and systems*, vol. 35, no. 10, pp. 1257–1272, 2002.

[4] L. Zhu, X. Wang, Z. Ke, W. Zhang, and R. W. Lau, "Biformer: Vision transformer with bi-level routing attention," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 10 323–10 333.

[5] N. O. Salscheider, "Feature NMS: Non-Maximum Suppression by Learning Feature Embeddings," in *Proceedings of the 2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, 2021, pp. 7848–7854.

[6] N. Abinaya, D. Susan, and S. Rakesh Kumar, "Naive Bayesian Fusion Based Deep Learning Networks for Multisegmented Classification of Fishes in Aquaculture Industries," *Ecological Informatics*, vol. 61, p. 101248, 2021.

[7] H. T. Rauf, M. I. U. Lali, S. Zahoor, S. Z. H. Shah, A. U. Rehman, and S. A. C. Bukhari, "Visual Features Based Automated Identification of Fish Species Using Deep Convolutional Neural Networks," *Computers and Electronics in Agriculture*, vol. 167, p. 105075, 2019.

[8] S. Z. H. Shah, H. T. Rauf, M. IkramUllah, M. S. Khalid, M. Farooq, M. Fatima, and S. A. C. Bukhari, "Fish-Pak: Fish Species Dataset from Pakistan for Visual Features Based Classification," *Data in brief*, vol. 27, pp. 104–565, Dec. 2019.

[9] A. Banerjee, A. Das, S. Behra, D. Bhattacharjee, N. T. Srinivasan, M. Nasipuri, and N. Das, "Carp-DCAE: Deep Convolutional Autoencoder for Carp Fish Classification," *Computers and Electronics in Agriculture*, vol. 196, p. 106810, 2022.

[10] S. A. Shammi, S. Das, M. Hasan, and S. R. H. Noori, "Fishnet: Fish Classification Using Convolutional Neural Network," in *Proceedings of the 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT)*. IEEE, 2021, pp. 1–5.

[11] X. Xu, W. Li, and Q. Duan, "Transfer Learning and SE-ResNet152 Networks-Based for Small-Scale Unbalanced Fish Species Identification," *Computers and Electronics in Agriculture*, vol. 180, p. 105878, 2021.

[12] A. Kuswantori, T. Suesut, W. Tangsrirat, and N. Nunak, "Development of Object Detection and Classification with YOLOv4 for Similar and Structural Deformed Fish," *EUREKA: Physics and Engineering*, no. 2, pp. 154–165, 2022.

[13] M. A. Ahmed, M. S. Hossain, W. Rahman, A. H. Uddin, and M. T. Islam, "An Advanced Bangladeshi Local Fish Classification System Based on the Combination of Deep Learning and the Internet of Things (IoT)," *Journal of Agriculture and Food Research*, vol. 14, p. 100663, 2023.

[14] B. Gong, K. Dai, J. Shao, L. Jing, and Y. Chen, "Fish-TViT: A Novel Fish Species Classification Method in Multi Water Areas Based on Transfer Learning and Vision Transformer," *Heliyon*, vol. 9, no. 6, pp. 1–12, 2023.

[15] M. Sohan, T. Sai Ram, and C. V. Rami Reddy, "A Review on YOLOv8 and Its Advancements," in *Proceedings of the International Conference on Data Intelligence and Cognitive Informatics*. Springer, 2024, pp. 529–545.

[16] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren, "Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 07, 2020, pp. 12 993–13 000.

[17] H. Zhang, Y. Wang, F. Dayoub, and N. Sunderhauf, "VarifocalNet: An IoU-Aware Dense Object Detector," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 8514–8523.

Assoc. Prof. Nguyen Quang Hoan was born in 1950, graduated from a university in Moscow, Russia (formerly the Soviet Union) in 1967. He worked as a research scientist at the Institute of Information Technology, Vietnam Academy of Science and Technology, from 1974 to 1998. From 1998 to 2010, he served as the Head of the Department of Information Technology at the Posts and Telecommunications Institute of Technology and was conferred the title of Associate Professor in 2002. His main research interests include machine learning, optimal control, and intelligent control. He is currently active as a lecturer and expert in the field. In 2018, he was recognized by UNESCO Vietnam as one of the 50 most outstanding scientists, with his journey titled "A Thousand-Mile Journey to Becoming a Neural Network Expert" introduced to the international scientific community.

Mr. Doan Hong Quang was born in 1979, received his M.Sc. degree in Computer Science from the University of Information Technology in 2014. He is currently pursuing a Ph.D. with a research focus on cellular neural networks and deep learning. He is also the Head of the Digital Technology Department at the Center for Microelectronics Technology, National Center Technological Progress (Nacentech). His main research interests include artificial intelligence, the Internet of Things (IoT), cybernetics, and agricultural automation.

Nguyen The Truyen born in 1964, received his Bachelor's degree in Radio-electronic Engineering from the Hanoi University of Science and Technology in 1988 and a Ph.D. in Electronic Engineering from the Vietnam Research Institute of Electronics, Informatics and Automation in 1999. He currently works for Vietnam Research Institute of Electronics, Informatics and Automation under the Ministry of Industry and Trade, Vietnam. His research interests include signal processing, the Internet of Things (IoT), artificial intelligence, and Industrial automation.

Dr. Duong Duc Anh was born in 1984, received his Bachelor's degree in Industrial Automation from the Hanoi University of Science and Technology in 2007, followed by a Master's degree in Automation and Control in 2009. In 2025, he earned a Ph.D. in Electronic Engineering from the Vietnam Research Institute of Electronics, Informatics and Automation. His primary research interests focus on Machine Learning, Neural Networks, Industrial Automation, Optimal Control, and Intelligent Control. Dr. Duc Anh is currently serving as Deputy Director of the Vietnam Research Institute of Electronics, Informatics and Automation under the Ministry of Industry and Trade, Vietnam.