International Journal of Electrical and Computer Engineering (IJECE)
Vol. 16, No. 1, February 2026, pp. 383~394
ISSN: 2088-8708, DOI: 10.11591/ijece.v16i1.pp383-394

Image classification using two neural networks and activation functions: a case study on fish species

Oppir Hutapea[1], Ford Lumban Gaol[2], Tokuro Matsuo[3]
[1]Software Engineering Technology, Faculty of Vocational Studies, Institut Teknologi Del - Sitoluama, Toba Samosir, Indonesia
[2]Computer Science Department, BINUS Graduate Program - Doctor of Computer Science, Bina Nusantara University, Jakarta, Indonesia
[3]Advanced Institute of Industrial Technology, Tokyo, Japan

Article Info

Article history:

Received Jan 23, 2025
Revised Sep 18, 2025
Accepted Nov 23, 2025

Keywords:

Activation function
Data collection
Feature extraction
Image detection
Machine learning
Neural network
Preprocessing

ABSTRACT

Lake Toba is utilized for aquaculture fishing as a clear example of how this technology can be applied. One of the species presents is the red devil fish (*Amphilophus labiatus*), which is known to have started appearing in the last 10 years. This species is known to be very aggressive and damage the ecosystem. When their populations go unchecked, red-devils can cause a decline in local fish populations, potentially destroying the balance of the food chain in those waters. This study used artificial neural network (ANN) and convolutional neural network (CNN) algorithms to successfully create two classification models for fish species from Lake Toba: red devil fish (*Amphilophus labiatus*), mujahir fish (*Oreochromis mossambicus*), sepat fish (*Trichogaster trichopterus*). The purpose of this model is to automatically identify fish species by using image-based classification techniques. According to the study's findings, both models performed exceptionally well and had a high degree of accuracy. This study addresses the lack of effective automated fish classification systems for ecosystems like Lake Toba, Indonesia, which are threatened by invasive species such as the red devil fish. By comparing CNN and ANN models with different activation functions and optimizers, we found that CNN with rectified linear unit (ReLU) activation and Adam optimizer provides the most accurate and stable results. The findings offer practical implications for fisheries management and biodiversity conservation.

Corresponding Author:

Oppir Hutapea
Software Engineer Technology, Faculty of Vocational Studies, Institut Teknologi Del - Sitoluama
Sitoluama, Balige, Toba, North Sumatra 22381, Indonesia
Email: oppir.hutapea@del.ac.id/oppirhutapea20@gmail.com

1. INTRODUCTION

In today's digital era, the application of machine learning (ML) based technology is increasingly widespread in various sectors, ranging from health, transportation, to the environment. One of the sectors that is starting to explore the great potential of this technology is fisheries, specifically in addressing the challenges related to the identification and classification of fish species there are many studies conducted using different methods [1], [2]. In addition, many technologies have been built in this context to develop advanced systems for underwater fish detection and recognition. Fish are an important part of the marine ecosystem as well as human culture and industry [3]. Lake Toba is one of the freshwater waters that has natural resources that are very important for people's lives, especially the fishermen who live around it. Lake Toba as one of the largest and richest aquatic ecosystems in Indonesia, Lake Toba is utilized for aquaculture fishing being a clear example of how this technology can be applied [4]. The lake has very rich biodiversity, including local fish species such as the dominant red devil [5]. It contains a variety of fish species that live and breed. One of the species presents is the red devil fish (*Amphilophus labiatus*), which is known to have started appearing in the last 10 years. The presence of invasive species such as red devil fish poses a serious threat to the balance of the aquatic ecosystem. This species is known to be very aggressive and can damage the ecosystem. Its aggressive characteristics not only disrupt interactions between fish species, but also threaten the existence of local fishes that play an important role in maintaining ecosystem balance. Therefore, proper management is needed to control the red devil fish population.

One of the main reasons for the need to detect red devil fish is its high adaptability. This species can quickly adjust to a wide range of aquatic environmental conditions. In addition, red devils are voracious predators of small fish, which are often endemic or native to a region [6]. When their populations go unchecked, red devils can cause a decline in local fish populations, potentially destroying the balance of the food chain in those waters. In addition to its aggressive nature and rapid adaptation, the red devil also has a very high reproductive rate [7]. This rapid breeding leads to population increases that are difficult to control in a short period of time. Its main breeding areas include around Lake Toba, which is one of the largest and most important freshwater ecosystems in Indonesia. The aquatic ecosystem of Lake Toba itself has been disrupted by the dominance of this species, given that Lake Toba is one of the vital freshwater sources for the surrounding region [8].

Red devil fish (*Amphilophus labiatus*) is characterized by a slender body that is similar to tilapia, but is distinguished by hard, leathery scales and pointed anal and dorsal fins. As an invasive species, it has high adaptability, fast growth, and easy reproduction. Red devil fish tend to be omnivorous with plankton, especially from the Chlorophyceae class, being their main food, and show flexibility in utilizing food resources [9]. The body size of fish found in Jatibarang Reservoir ranged from 8-18.5 cm, with an average weight of 47.8 grams The tilapia (*Oreochromis mossambicus*) is an important freshwater fish species in aquaculture. This species has a high tolerance to various environmental conditions, fast growth, and is easily spawned, making it a major commodity in various regions, including Merauke [10]. Tilapia is known as an invasive species that is able to live together with tilapia (*Oreochromis niloticus*), often causing hybridization that decreases genetic diversity. In molecular studies, tilapia showed low levels of genetic distance from similar species, with only one haplotype identified. This fish has a body size of 20-40 cm and is often found in abundance in nature, making it an affordable main source of protein for the community The sepat fish (*Trichogaster trichopterus*), a species of the family *Anabantidae*, has morphological characteristics that can adapt to its habitat. It originated from Southeast Asian waters and was introduced to Indonesia in 1934. It can live in lentic (such as reservoirs) and lotic (such as rivers) waters [11]. It is body morphology differs according to habitat: fish from rivers have more elongated bodies and straight backs to adapt to fast currents, while fish from reservoirs have curved backs to adapt to still water. In addition, sepat fish have the advantage of modified whip-like pectoral fins for touch, which help to survive in extreme environments.

To address these challenges, the development of a machine learning-based system is a promising solution. Using this technology, the system can be trained to recognize the unique traits of each species, thus being able to accurately distinguish red devil from mujahir and sepat. This system not only helps in detecting the presence of invasive species, but also supports the conservation of local fish that are important to the ecosystem. Through the application of machine learning, fish species identification can be done efficiently, both on a research and water management scale. This technology is expected to be an effective tool in environmental conservation efforts, while supporting the sustainability of freshwater ecosystems in the future. Furthermore, with the ability to quickly and accurately identify red devils, the system can also help cull the invasive fish population in a targeted and systematic manner [12], [13]. By utilizing technology to track and control the spread of red devils, we can reduce their negative impact on local fish. The use of methods such as mass capture or focused culling of red devil populations can be done more efficiently, significantly reducing the number of invasive fish, and providing space for local fish to reproduce. With this approach, it is hoped that Lake Toba can be restored, maintain ecosystem balance, and support the survival of native fish species that are an important resource for the surrounding community.

Application of convolutional neural network (CNN) algorithm for automatic identification of various fish species. CNN is effective in image processing, with a structure consisting of multiple convolution and pooling layers that extract features from fish images. Preprocessing, including image resizing and normalization, is performed prior to model training. The dataset used consisted of live captured fish images [14], [15]. The training results showed an accuracy of 85.18%, indicating that the CNN was able to classify fish well, despite the challenge of distinguishing similar species. This research contributes to the development of an automatic identification system that is beneficial to researchers and observers in the field of aquatic biology[16].

Based on the findings and approaches proposed by two previous researchers, we decided to use convolutional neural network (CNN) and artificial neural network (ANN) algorithms as the main methods in classifying fish species found in Lake Toba [17], [18]. The CNN algorithm was chosen because of its superior ability to recognize visual patterns from images, such as shape, texture, and fish-specific features, while ANN is used to process additional data or other numerical features that support the classification process. The combination of these two algorithms is expected to provide more accurate and reliable results in identifying fish species in the region.

This study presents a novel combination of CNN and ANN for multi-class fish species classification using image data. Unlike most existing approaches that rely on a single architecture, we evaluate the comparative performance of both models using two activation functions (rectified linear unit (ReLU) and Tanh), optimizers (Adam and stochastic gradient descent (SGD)), and multiple learning rates. This integrated approach allows a deeper understanding of how activation and optimization choices affect classification outcomes for real-world aquatic datasets.

2. METHOD

2.1. Research diagram

The data collection process carried out by researchers was carried out manually, by taking image samples directly in the area around Lake Toba. The specific location of data collection was focused on the Balige area, which is one of the strategic areas around Lake Toba. The data collection involved capturing images of fish at various angles and lighting conditions to ensure data diversity that could support accuracy in the process of analyzing and classifying fish species. This manual approach allows researchers to directly ensure the quality of the data and its relevance to the research needs. The complete flow of building the machine learning model has been summarized in a research diagram.

A research diagram is a form of visualization that describes the systematic flow of research carried out with the aim of making it easier to understand the flow of work. The diagram will describe the main steps of the research presented in a structured manner starting from problem identification to conclusion. The use of this diagram will help to ensure the research runs and convey broader information. The research diagram can be seen in Figure 1.

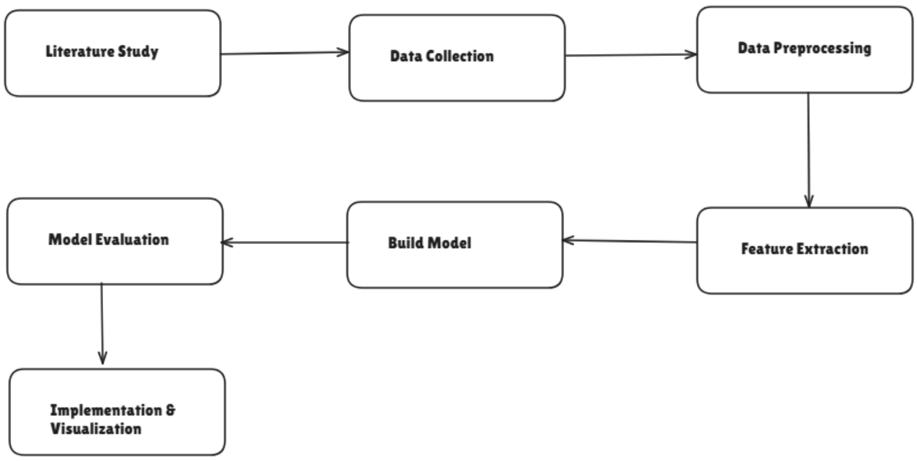

Figure 1. Research flow

2.2. Data collection

After conducting a literature study, the next step is to collect data manually. The data was collected manually by photographing from various angles such as the side, top, front, and also the back from different angles [19]. This is done to further enrich the variety of data. After taking pictures of the fish manually, the original data was collected as many as 651 images. The following is sample data from the data that will be used as a dataset.

2.3. Preprocessing data

After collecting the images manually, the images are entered into the Roboflow for the processing stage. Then the first thing to do is to label each image where there are three labels, namely red devil

(Figure 2(a)), mujahir (Figure 2(b)), and sepat (Figure 2(c)). After all the images have been labeled, to further increase the number of images, the augmentation process is carried out. The augmentation process involves rotation, flipping, brightness, crop, and zooming techniques to create new variations of the original data [20]. The more data used in training the model, the more the model's capabilities will increase as more patterns or images will be learned. So, after the augmentation process, the number of images collected is 3,063 images.

Preprocessing serves to prepare the datasets required in the project in a structured and efficient way. The uses of preprocessing in this context include: Data collection (collecting all image files from the dataset folder and organizing them by category. This makes it easy to identify and access the data required for model training) [21], [22]. Dataset organization (dividing the dataset into three subsets namely train, test, and validation data which will aim to accurately evaluate the performance of the model). This division prevents overfitting and ensures the model can generalize well. Organized folder structure (copy files into separate folders based on predefined sets. This eases the process of data management and access when conducting training and testing).

(a) (b) (c)

Figure 2. Image sampling (a) *Amphilophus labiatus* (red devil), (b) *Oreochromis mossambicus* (mujahir), and (c) *Trichopodus pectoralis* (Flatfish)

2.4. Feature extraction

At this stage, the unique patterns of each fish species are examined. Features include body color, fin shape, texture patterns, and other traits. Using CNN and ANN techniques, all features will be extracted to provide a more accurate representation for use in classification [23], [24]. Feature extraction is a crucial stage in the fish classification process, where the distinctive physical characteristics of each species such as body coloration, fin morphology, texture patterns, and overall body shape are systematically analyzed to capture meaningful visual representations. CNNs play a central role in this phase due to their ability to automatically learn hierarchical spatial features directly from raw image inputs. Through convolutional filters, CNNs can effectively detect edges, color gradients, textures, and complex patterns that differentiate one species from another, even when variations in lighting, orientation, or fish pose occur. This automated feature-learning capability allows CNNs to generate highly discriminative and robust feature maps, providing a more accurate and reliable foundation for the classification stage. By relying on CNN-based feature extraction, the system can achieve consistent performance and improved recognition accuracy across diverse datasets [24].

2.5. Training model

At this stage the machine learning model is trained using training data to learn patterns and relationships between input and output features. Modeling is done iteratively until finding a model that can work well where it can predict the type of fish, namely red devil, mujahir, and also sepat. The result is a model that can predict or classify new data based on learned patterns [24], [25].

2.6. Model evaluation

At this stage, assessing the performance of the trained model using validation or test data. The aim is to measure how well the model works on data that has not been seen before to detect issues such as overfitting or underfitting. This stage helps in selecting the best reliable model for implementation.

3. RESULTS AND DISCUSSION

In research on the classification of fish found in Lake Toba, namely red devil fish (*Amphilophus labiatus*), mujahir fish (*Oreochromis mossambicus*), sepat fish (*Trichogaster trichopterus*) using 2 models, namely convolutional neural network and artificial neural network. Where all data is still taken manually due to limited datasets which are then processed using Roboflow so that the amount of data used is 3,063 images. To implement two hyperparameter activation functions, namely ReLU and Tanh [26], [27]. Several experiments were conducted to find the best accuracy, precision, recall and F1-score values by combining the activation function along with several optimizers and learning rates. The experiments were conducted by using an epoch value of 25. Then by using a learning rate of 0.01 and 0.001 [28]. Also, by using SGD optimizer for Tanh and Adam for ReLU. In the experiment, the epoch value of 25 is more effective, because in the training process the best accuracy value can be generated at epochs 21 to 37 by using the EarlyStop method so that the training process can be stopped at a certain epoch when it reaches the lowest loss value and in the next 5 epochs it does not decrease the loss value. This is done to avoid overfitting the model.

3.1. CNN

3.1.1. Input layer

a. Epoch 15

- Learning rate 0.1 with Tanh and ReLU activation

When the learning rate is set high, like 0.1 as shown in Table 1, the model learns quickly at first, but this can cause problems in deeper layers. The Tanh activation function may struggle with vanishing gradients, making it hard for the model to learn effectively. On the other hand, ReLU helps by keeping the gradient strong, which makes learning more stable. However, a very high learning rate can cause the model to jump over the best solution, leading to unstable loss values. It is important to keep an eye on the validation results to spot any signs that the model's performance.

Table 1. The validation results with 0.1 (learning rate)

Aggregate Function	Optimizer	Learning rate	Class	Val Acc	Precision	Recall	F1-Score
Tanh	SGD	0.1	*Oreochromis mossambicus*	0.97	1.00	1.00	1.00
			Amphilophus labiatus		0.94	1.00	0.97
			Trichopodus pectoralis		1.00	0.94	0.97
ReLU	Adam		*Oreochromis mossambicus*	0.33	0.00	0.00	0.00
			Amphilophus labiatus		0.33	1.00	0.50
			Trichopodus pectoralis		0.00	0.00	0.00

- Learning rate 0.01 with Tanh and ReLU activation

A lower learning rate (0.01) encourages more stable weight updates as shown in Table 2, which is especially useful for Tanh, which suffers from delayed convergence owing to saturating gradients. ReLU adds to this option by retaining gradient propagation, although the model may need longer epochs to reach equal accuracy to the higher learning rate setup. This setup is less susceptible to rapid performance changes, making it ideal for fine-tuning.

Table 2. The validation results with 0.01 (learning rate)

Aggregate Function	Optimizer	Learning rate	Class	Val Acc	Precision	Recall	F1-Score
Tanh	SGD	0.01	*Oreochromis mossambicus*	0.98	1.00	1.00	1.00
			Amphilophus labiatus		1.00	1.00	1.00
			Trichopodus pectoralis		1.00	1.00	1.00
ReLU	Adam		*Oreochromis mossambicus*	0.93	1.00	1.00	1.00
			Amphilophus labiatus		0.94	1.00	0.97
			Trichopodus pectoralis		1.00	0.94	0.97

3.1.2. Hidden layer

a. Epoch 20

- Learning rate 0.1 with Tanh and ReLU activation

The 0.1 learning rate with Tanh activation shows effective feature transformation in the hidden layers at mid-training (epoch 20) as Table 3. Although the rapid learning rate may result in oscillatory behavior in weight updates, the concurrent usage of ReLU aids in maintaining gradient flow across the network's intermediate representations. Batch normalization between layers helps this design, which is especially sensitive to appropriate initialization.

Table 3. The validation results with 0.1 learning rate (Epoch 20)

Aggregate Function	Optimizer	Learning rate	Class	Val Acc	Precision	Recall	F1-Score
Tanh	SGD	0.1	*Oreochromis mossambicus*	0.97	1.00	1.00	1.00
			Amphilophus labiatus		0.94	1.00	0.97
			Trichopodus pectoralis		1.00	0.94	0.97
ReLU	Adam		*Oreochromis mossambicus*	0.50	0.00	0.00	0.00
			Amphilophus labiatus		0.50	1.00	0.50
			Trichopodus pectoralis		0.00	0.00	0.00

- Learning rate 0.01 with Tanh and ReLU activation

More controlled feature refinement is made possible by the hidden layers' lower learning rate of 0.01, Tanh offers smooth nonlinear transformations, while ReLU keeps gradients from completely disappearing, shown in Table 4. Although it could take more epochs to attain performance comparable to higher learning rate configurations, this combination usually exhibits more consistent loss reduction during mid-training phases. For networks with intricate hierarchical feature representations, this configuration is frequently preferred due to its stability.

Table 4. The validation results with 0.01 learning rate (Epoch 20)

Aggregate function	Optimizer	Learning rate	Class	Val Acc	Precision	Recall	F1-Score
Tanh	SGD	0.01	*Oreochromis mossambicus*	1.00	1.00	1.00	1.00
			Amphilophus labiatus		1.00	1.00	1.00
			Trichopodus pectoralis		1.00	1.00	1.00
ReLU	Adam		*Oreochromis mossambicus*	0.94	1.00	0.94	0.97
			Amphilophus labiatus		0.94	1.00	0.97
			Trichopodus pectoralis		1.00	1.00	1.00

3.1.3. Output layer

a. Epoch 25

- Learning rate 0.1 with Tanh and ReLU activation

Table 5 present the 0.1 learning rate with Tanh activation at the output layer during later training stages (epoch 25) might speed up final changes to decision boundaries, which is especially helpful for constrained output spaces. However, when applied to final layers, ReLU's unbounded positive output might need to be carefully scaled, which could cause instability in the network's predictions. To avoid overfitting during these crucial final updates, this arrangement necessitates careful monitoring of validation metrics.

Table 5. The validation results with 0.1 learning rate (Epoch 25)

Aggregate Function	Optimizer	Learning rate	Class	Val Acc	Precision	Recall	F1-Score
Tanh	SGD	0.1	*Oreochromis mossambicus*	0.97	1.00	1.00	1.00
			Amphilophus labiatus		0.96	1.00	0.98
			Trichopodus pectoralis		1.00	0.96	0.98
ReLU	Adam		*Oreochromis mossambicus*	0.50	0.00	0.00	0.00
			Amphilophus labiatus		0.50	1.00	0.50
			Trichopodus pectoralis		0.00	0.00	0.00

- Learning rate 0.01 with Tanh and ReLU activation

Tanh produces outputs that are inherently bounded, which may be useful for some tasks, and the output layer's 0.01 learning rate allows for exact fine-tuning of classification bounds. ReLU's addition to the output layer can work well when combined with this conservative learning rate and appropriate normalization methods. Although it could take more time to train the network to its maximum potential, this configuration produces more stable final models, as illustrated in Table 6. In the graph in the Figure 3, for the model using the optimizer SGD shows that the lowest loss value occurs at the 10^{th} epoch, then the value of the best accuracy occurred at the 10^{th} epoch.

3.1.4. Convolution layer

The input layer of the CNN algorithm shows that the CNN model accepts images as input with a size of 224 pixels high, 224 pixels wide, and 3 color channels (RGB) as shown in Figure 4. These images come from a dataset that is processed using the *load_img* function and resized to fit those dimensions. This size is used as a standard to enable convolution and pooling operations in the network, where each image will go through the process of extracting features such as patterns, textures, and edges. A consistent input size is essential for the model to optimally learn the feature representation at each layer.

Table 6. The validation results with 0.01 learning rate (Epoch 25)

Aggregate Function	Optimizer	Learning rate	Class	Val Acc	Precision	Recall	F1-Score
Tanh	SGD	0.01	*Oreochromis mossambicus*	1.00	1.00	1.00	1.00
			Amphilophus labiatus		1.00	1.00	1.00
			Trichopodus pectoralis		1.00	1.00	1.00
ReLU	Adam		*Oreochromis mossambicus*	0.50	1.00	1.00	1.00
			Amphilophus labiatus		0.96	1.00	0.96
			Trichopodus pectoralis		1.00	0.94	0.98

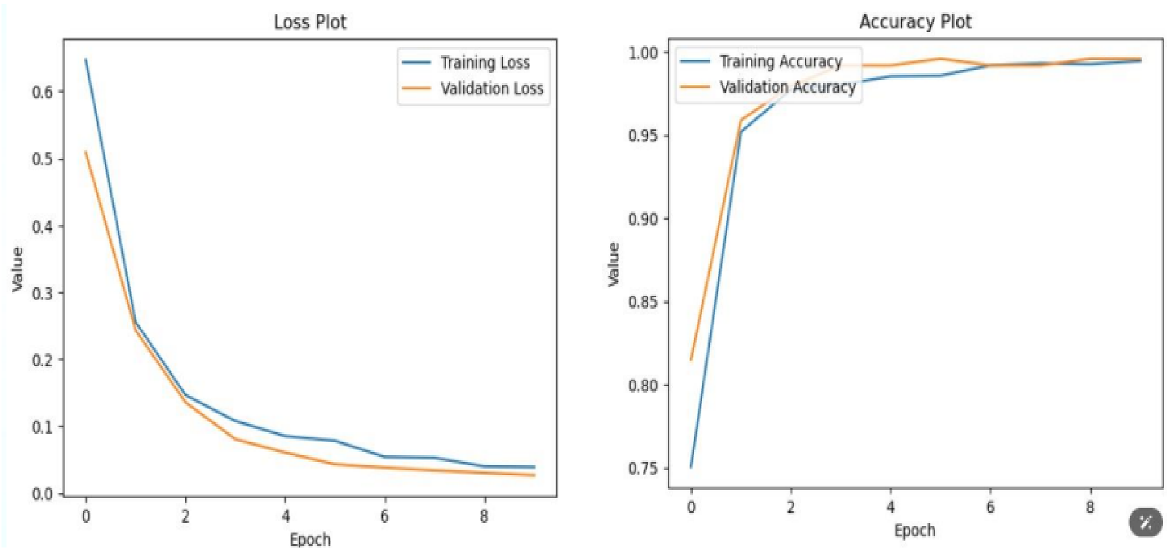

Figure 3. Loss plot vs accuracy plot

Figure 4. Image formatting

3.2. ANN

3.2.1. Epoch 15

- Learning rate 0.1 with Tanh and ReLU activation

At epoch 15(as shown in Table 7), with a high learning rate of 0. 1, models using both Tanh and ReLU start to learn very quickly. This fast learning can help reach the best solution faster, but it also makes it easier to go past the best point, especially with Tanh, which can stop working properly. ReLU handles this better because it does not stop working as easily, but it can still have a problem if the changes become too big.

Table 7. The validation result with 0.1 learning rate

Aggregate Function	Optimizer	Learning rate	Class	Val Acc	Precision	Recall	F1-Score
Tanh	SGD	0.1	*Oreochromis mossambicus*	0.99	1.00	1.00	1.00
			Amphilophus labiatus		1.00	1.00	1.00
			Trichopodus pectoralis		1.00	1.00	1.00
ReLU	Adam		*Oreochromis mossambicus*	0.33	0.00	0.00	0.00
			Amphilophus labiatus		0.00	0.00	0.00
			Trichopodus pectoralis		0.33	1.00	0.50

- Learning rate 0.01 with Tanh and ReLU activation

Using a learning rate of 0. 01 at epoch 15 helps both Tanh and ReLU converge more steadily. Tanh gives smoother gradient changes in the middle range, while ReLU allows quicker training because it is simpler and uses less data. By this point, both activation functions start to improve consistently with less chance of becoming unstable, as shown in Table 8.

Table 8. The validation result with 0.01 learning rate

Aggregate Function	Optimizer	Learning rate	Class	Val Acc	Precision	Recall	F1-Score
Tanh	SGD	0.01	*Oreochromis mossambicus*	0.98	0.97	0.97	0.97
			Amphilophus labiatus		1.00	1.00	1.00
			Trichopodus pectoralis		0.97	0.97	0.97
ReLU	Adam		*Oreochromis mossambicus*	0.33	0.00	0.00	0.00
			Amphilophus labiatus		0.44	1.00	0.61
			Trichopodus pectoralis		1.00	0.73	0.84

3.2.2. Epoch 20

- Learning rate 0.1 with Tanh and ReLU activation

At epoch 20, a learning rate of 0. 1 might still work well and lead to quick improvements, but there could be signs of problems, especially in networks that use Tanh. These networks might struggle with vanishing gradients in deeper layers. ReLU can still work, but if the learning rate is too high, it might cause the loss to fluctuate a lot. This setup might be better for simpler networks or ones that use batch normalization (shown in Table 9).

Table 9. The validation result with 0.1 learning rate (Epoch 20)

Aggregate Function	Optimizer	Learning rate	Class	Val Acc	Precision	Recall	F1-Score
Tanh	SGD	0.1	*Oreochromis mossambicus*	0.99	1.00	1.00	1.00
			Amphilophus labiatus		1.00	1.00	1.00
			Trichopodus pectoralis		0.97	0.97	0.97
ReLU	Adam		*Oreochromis mossambicus*	0.33	0.00	0.00	0.00
			Amphilophus labiatus		0.33	1.00	0.50
			Trichopodus pectoralis		0.70	0.43	0.23

- Learning rate 0.01 with Tanh and ReLU activation

With a learning rate of 0.01, both activation functions show steady progress by the 20th epoch, as shown in Table 10. Tanh keeps performing well in tasks where the output needs to stay within a certain range, while ReLU keeps being effective in situations where only some neurons are active. The slower learning rate helps prevent the model from going off track and improves how well it works with new data.

Table 10. The validation result with 0.01 learning rate (Epoch 20)

Aggregate Function	Optimizer	Learning rate	Class	Val Acc	Precision	Recall	F1-Score
Tanh	SGD	0.01	*Oreochromis mossambicus*	0.98	1.00	1.00	1.00
			Amphilophus labiatus		1.00	1.00	1.00
			Trichopodus pectoralis		0.97	0.97	0.97
ReLU	Adam		*Oreochromis mossambicus*	0.66	0.44	0.77	0.66
			Amphilophus labiatus		0.74	1.00	0.81
			Trichopodus pectoralis		1.00	0.83	0.88

3.2.3. Epoch 25

- Learning rate 0.1 with Tanh and ReLU activation

By the 25th epoch (according to Table 11), using a learning rate of 0.1 can cause training to become unstable or stop improving, especially when using Tanh because it reacts strongly to big changes in gradients. ReLU might still work okay, but there's a chance the model could bounce around the lowest points without settling. This setup is usually not advised unless you also use a learning rate that decreases over time.

- Learning rate 0.01 with Tanh and ReLU activation

At this stage, there is a learning rate of 0. 01 still helps the training go smoothly and reliably for both activation functions. Tanh gives outputs that are more centered, which can be useful in some network designs, while ReLU keeps performing well because it is simple and efficient. This setup works well for training deeper than 25 epochs, as illustrated in Table 12.

Table 11. The validation result with 0.1 learning rate (Epoch 25)

Aggregate Function	Optimizer	Learning rate	Class	Val Acc	Precision	Recall	F1-Score
Tanh	SGD	0.1	*Oreochromis mossambicus*	0.99	1.00	1.00	1.00
			Amphilophus labiatus		1.00	1.00	1.00
			Trichopodus pectoralis		0.97	0.97	0.97
ReLU	Adam		*Oreochromis mossambicus*	0.33	0.00	0.00	0.00
			Amphilophus labiatus		0.33	1.00	0.50
			Trichopodus pectoralis		0.00	0.00	0.00

Table 12. The validation result with 0.01 learning rate (Epoch 25)

Aggregate Function	Optimizer	Learning rate	Class	Val Acc	Precision	Recall	F1-Score
Tanh	SGD	0.01	*Oreochromis mossambicus*	0.99	1.00	1.00	1.00
			Amphilophus labiatus		1.00	1.00	1.00
			Trichopodus pectoralis		1.00	1.00	1.00
ReLU	Adam		*Oreochromis mossambicus*	0.97	1.00	1.00	1.00
			Amphilophus labiatus		1.00	1.00	1.00
			Trichopodus pectoralis		1.00	1.00	1.00

Then Figure 5 shows a graph of accuracy and loss from one of the tests with the highest accuracy value shown above. Both the training and validation losses keep going down as the model goes through more training rounds, which means the model is learning well. The validation loss stays close to the training loss without a big gap, showing that the model is not overfitting, meaning it does not do great on training data but worse on new, unseen data. The final loss values are low, around 0.1, which is a good sign for how well the model works. Both training and validation accuracies go up overtime, reaching very high levels, more than 90%. The validation accuracy follows the training accuracy closely, which again shows the model is not overfitting. The final accuracy values are high, around 95% for training and just a little lower for validation.

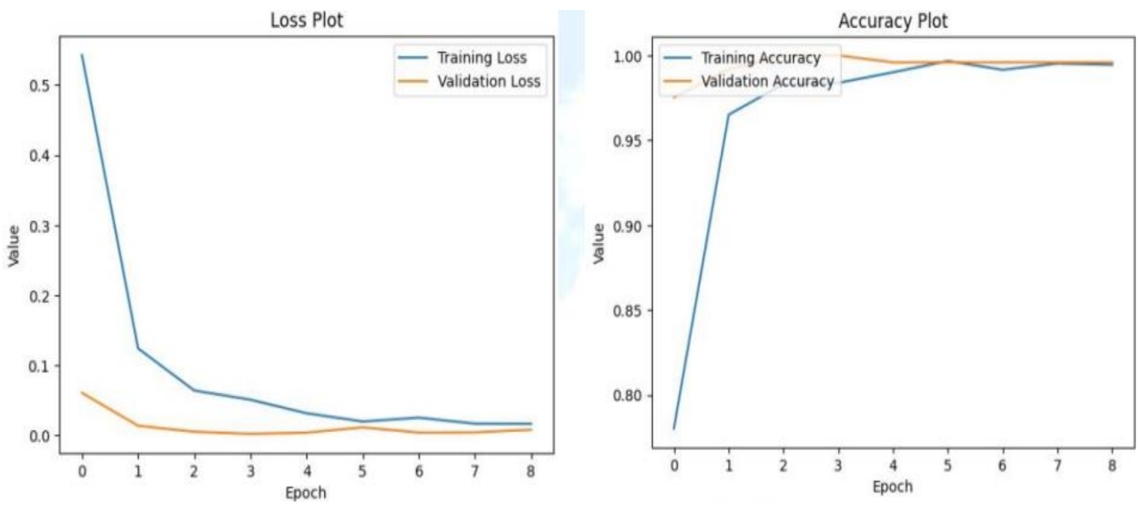

Figure 5. Loss and accuracy plot

4. CONCLUSION

A key contribution of this work lies in the systematic evaluation of activation functions and optimizers in both CNN and ANN models for aquatic image classification. This comparison offers new insights into the architectural performance trade-offs. Future extensions include integrating this system into real-time monitoring platforms for invasive species management and adapting the framework to classify other species in diverse ecosystems. This research successfully developed two classification models for fish species from Lake Toba, namely red devil fish (*Amphilophus labiatus*), mujahir fish (*Oreochromis mossambicus*), sepat fish (*Trichogaster trichopterus*) using ANN and CNN algorithms. This model is designed to utilize image-based classification techniques to automatically detect fish species. From the research results, both models showed excellent performance with a high level of accuracy. The CNN model gave the best results on the test data with a combination of ReLU activation function and Adam optimizer. The ANN model, with a similar configuration, also showed excellent performance, proving that both approaches are equally effective in image classification tasks. Reportedly, the CNN model with ReLU activation function is more stable and fast in the training process than the ANN. Tests were conducted on 10% of the data from the total dataset (3,063 images) that had been separated as test data. To address the challenge of limited training data and enhance model robustness, the implementation of advanced data augmentation techniques is strongly recommended. Below are the details of the test results on the test data for each fish species: red devil: All images were classified correctly without error. Mujahir: The model was able to recognize this species with perfect accuracy. Flathead: No classification errors occurred, indicating the model can distinguish the unique features of this fish. The classification technique used was image-based classification with a deep learning model. CNNs are designed to extract visual features from images using convolution, pooling and dense layers, while ANNs process flattened image data to generate predictions. Both techniques enable the recognition of complex patterns in fish images, with features such as body shape, texture, and color as key indicators. Overall, this research makes a significant contribution in supporting the operational efficiency of the fisheries sector and biodiversity conservation in Lake Toba. The results can serve as a basis for the implementation of similar technologies in other water areas.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Oppir Hutapea	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	
Ford Lumban Gaol		✓		✓		✓			✓	✓	✓	✓		
Tokuro Matsuo	✓			✓		✓				✓				

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

These data may include raw datasets, processed results, or supplementary materials that were used to draw the conclusions presented in this research. Interested researchers may contact the author to request access, provided the request is for legitimate academic or scientific purposes and complies with any applicable ethical or privacy considerations.

REFERENCES

[1] R. F. Rahmat *et al.*, "Identification of the fish survival rate and the fish types to live in Lake Toba using machine learning," *Journal of Theoretical and Applied Information Technology*, vol. 31, no. 10, 2021.

[2] R. Qomaria, "Species diversity of freshwater fish in Lake Toba using shanon wiener," *Formosa Journal of Applied Sciences*, vol. 2, no. 7, pp. 1475–1482, 2023, doi: 10.55927/fjas.v2i7.4936.

[3] F. Yessoufou and J. Zhu, "Classification and regression-based convolutional neural network and long short-term memory configuration for bridge damage identification using long-term monitoring vibration data," *Structural Health Monitoring*, vol. 22, no. 6, pp. 4027–4054, 2023, doi: 10.1177/14759217231161811.

[4] R. E. Lubis, M. H. Irawaty, Ibrohim, and S. E. Indriwati, "Identification and diversity of organisms in the Lake Toba area," *Jurnal Pendidikan IPA Indonesia*, vol. 8, no. 3, pp. 361–370, 2019, doi: 10.15294/jpii.v8i3.19043.

[5] M. Z. Ulhaq, M. Farid, Z. I. Aziza, T. M. F. Nuzullah, F. Syakir, and N. R. Sasmita, "Forecasting upwelling phenomena in Lake Laut Tawar: a semi-supervised learning approach," *Infolitika Journal of Data Science*, vol. 2, no. 2, pp. 53–61, Nov. 2024, doi: 10.60084/ijds.v2i2.211.

[6] M. Sudhakara, M. J. Meena, O. Reddy, V. Mahalakshmi, and A. Balobaid, "A study on fish classification techniques using convolutional neural networks on highly challenged underwater images," *International Journal on Recent and Innovation Trends in Computing and Communication*, vol. 10, no. 4, pp. 1–9, Apr. 2022, doi: 10.17762/ijritcc.v10i4.5524.

[7] A. Mohanty, G. Goldsztein, and R. Pellegrin, "Fish species image classification using convolutional neural networks," *Journal of Student Research*, vol. 11, no. 3, 2022.

[8] H. L. Ohee, N. Mote, M. A. Rice, P. Sujarta, and S. Surbakti, "Length-weight relationship and condition factor of introduced red devil Cichlid Amphilophus labiatus, in Lake Sentani, Indonesia," *JURNAL BIOLOGI PAPUA*, vol. 16, no. 1, pp. 27–33, Apr. 2024, doi: 10.31957/jbp.3321.

[9] S. Boujerfaoui, H. Douzi, R. Harba, and K. Gourrame, "ReRNet: recursive neural network for enhanced image correction in print-cam watermarking," *International Journal of Electrical and Computer Engineering*, vol. 15, no. 1, pp. 356–364, Feb. 2025, doi: 10.11591/ijece.v15i1.pp356-364.

[10] R. Lumbantoruan, N. Rajagukguk, A. U. Lubis, M. Claudia, and H. Simanjuntak, "Two-step convolutional neural network classification of plant disease," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 14, no. 1, pp. 584–591, Feb. 2025, doi: 10.11591/ijai.v14.i1.pp584-591.

[11] T. Noranis, M. Aris, C. Ningning, N. Mustapha, and M. Zolkepli, "Integration of CNN and LSTM networks for behavior feature recognition: an analysis," *International Journal on Advanced Science, Engineering & Information Technology*, vol. 14, no. 5, 2024.

[12] S. I. A. Al-Janabi, S. T. F. Al-Janabi, and B. Al-Khateeb, "Image classification using convolution neural network based hash encoding and particle swarm optimization," in *2020 International Conference on Data Analytics for Business and Industry: Way Towards a Sustainable Economy, ICDABI 2020*, Oct. 2020, pp. 1–5, doi: 10.1109/ICDABI51230.2020.9325655.

[13] N. Nurtiwi, R. Ruliana, and Z. Rais, "Convolutional neural network (CNN) method for classification of images by age," *JINAV: Journal of Information and Visualization*, vol. 3, no. 2, pp. 126–130, Dec. 2022, doi: 10.35877/454ri.jinav1481.

[14] V. Rezende, M. Costa, A. Santos, and R. C. L. De Oliveira, "Image processing with convolutional neural networks for classification of plant diseases," in *Proceedings - 2019 Brazilian Conference on Intelligent Systems, BRACIS 2019*, Oct. 2019, pp. 705–710, doi: 10.1109/BRACIS.2019.00128.

[15] S. Panigrahi, A. Nanda, and T. Swarnkar, "Deep learning approach for image classification," in *Proceedings - 2nd International Conference on Data Science and Business Analytics, ICDSBA 2018*, Dec. 2018, pp. 511–516, doi: 10.1109/ICDSBA.2018.00101.

[16] D. Gangadia, "Activation functions: experimentation and comparison," in *2021 6th International Conference for Convergence in Technology, I2CT 2021*, Apr. 2021, pp. 1–6, doi: 10.1109/I2CT51068.2021.9417890.

[17] N. D. Girsang, "Literature study of convolutional neural network algorithm for batik classification," *Brilliance Research of Artificial Intelligence,* vol. 1, no. 1, pp. 1-7, doi: 10.47709/brilliance.v1i1.1069.

[18] G. P. Zhang, "Neural networks for classification: a survey," *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, vol. 30, no. 4, pp. 451–462, Nov. 2000, doi: 10.1109/5326.897072.

[19] Samuel, M. Dwirastina, and Y. C. Ditya, "Sepat Siam fish community (Trichopodus Pectoralis Regan, 1910) in Semayang Lake, East Kalimantan," in *IOP Conference Series: Earth and Environmental Science*, 2021, p. 12069, doi: 10.1088/1755-1315/890/1/012069.

[20] J. M. J. Mol and S. A. Jose, "Fish species classification using optimized deep learning model," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 9. pp. 825–837, 2022, doi: 10.14569/IJACSA.2022.0130996.

[21] M. Sayouh, M. Ali, Y. Li, Y. F. Tao, S. Q. Lu, and J. Qiang, "Differences in growth performance and meat quality between male and female Juvenile Nile Tilapia (Oreochromis niloticus) during separate rearing," *Animals*, vol. 14, no. 20, Oct. 2024, doi: 10.3390/ani14202954.

[22] W. Sholihah, I. Novianty, A. Hendriana, I. Kusumanti, F. Marcelita, and N. Aziezah, "Weight prediction model of Red Tilapia (Oreochromis Niloticus) using linear regression algorithm," *Universal Journal of Agricultural Research*, vol. 11, no. 6, pp. 1109–1116, 2023, doi: 10.13189/ujar.2023.110618.

[23] M. Jogin, Mohana, M. S. Madhulika, G. D. Divya, R. K. Meghana, and S. Apoorva, "Feature extraction using convolution neural networks (CNN) and deep learning," in *2018 3rd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*, May 2018, pp. 2319–2323, doi: 10.1109/RTEICT42901.2018.9012507.

[24] E. Xi, "Image feature extraction and analysis algorithm based on multi-level neural network," in *Proceedings - 5th International Conference on Computing Methodologies and Communication, ICCMC 2021*, Apr. 2021, pp. 1062–1065, doi: 10.1109/ICCMC51019.2021.9418309.

[25] K. A. Sathi and M. S. Islam, "Hybrid feature extraction based brain tumor classification using an artificial neural network," in *2020 IEEE 5th International Conference on Computing Communication and Automation, ICCCA 2020*, Oct. 2020, pp. 155–160, doi: 10.1109/ICCCA49541.2020.9250760.

[26] A. D. Rasamoelina, F. Adjailia, and P. Sincak, "A review of activation function for artificial neural network," in *2020 IEEE 18th World Symposium on Applied Machine Intelligence and Informatics (SAMI)*, Jan. 2020, pp. 281–286, doi: 10.1109/SAMI48414.2020.9108717.

[27] L. S. Pusztahazi, G. Eigner, and O. Csiszar, "Parametric activation functions for neural networks: a tutorial survey," *IEEE Access*, vol. 12, pp. 168626–168644, 2024, doi: 10.1109/ACCESS.2024.3474574.

[28] Y. Liu *et al.*, "Classification of tank images using convolutional neural network," in *2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, Jun. 2020, pp. 210–214, doi: 10.1109/ITNEC48623.2020.9085151.

BIOGRAPHIES OF AUTHORS

Oppir Hutapea obtained a bachelor's degree in informatics engineering from Institut Teknologi Del, Laguboti, North Sumatra, Indonesia, in 2018. He then earned a master's degree from Bina Nusantara University in 2023. Currently, he serves as a lecturer in the applied bachelor's program in software engineering technology, with research interests in artificial intelligence (AI), machine learning, data science, and big data. Before transitioning to academia, he gained approximately four years of professional experience in the e-commerce industry in Indonesia, working in the same field. This experience provides him with a comprehensive perspective in bridging technical expertise with education in his area of specialization. He can be contacted at email: oppir.hutapea@del.ac.id or oppirhutapea20@gmail.com.

Ford Lumban Gaol is currently an associate professor of informatics engineering and information systems at Bina Nusantara University. He is the vice chair of the Bina Nusantara University doctorate program in computer science and the research interest group leader of "Advanced system in computational intelligence and knowledge engineering." He is also the vice chair of the IEEE Indonesia section for international and professional activities, the past chair of the ACM Indonesia chapter, and the chair of the IIAI Indonesia chapter. He has been involved in several projects related to technology alignment in multinational companies as well as government projects. For international highlights, he was a visiting professor at Kazan Federal University, Russia, in 2014 and 2015, a visiting professor at Vladimir State University, Russia, in 2016, an invited scholar at Aligarh Muslim University, a keynote speaker at ICCNT 2014, and an invited scholar at ICTP, Trieste, Italy. He is a member of the Indonesian Mathematical Society (IndoMS), the association for computing machinery (ACM), the International Association of Engineers (IAENG), and the Indonesia Society for Bioinformatics. He holds a B.Sc. in mathematics, a master's degree in computer science, and a doctoral degree in computer science from the University of Indonesia, obtained in 1997, 2001, and 2009, respectively. He can be contacted at email: fgaol@binus.edu.

Tokuro Matsuo (member, IEEE) received the Ph.D. degree in engineering from the Department of Computer Science, Nagoya Institute of Technology, in 2006. He was a visiting researcher at the University of California at Irvine from 2010 to 2011, a research fellow at Shanghai University from 2010 to 2013, and a research project professor at the Green Computing Research Center, Nagoya Institute of Technology, from 2011 to 2014. He has been a research fellow at SEITI, Central Michigan University, USA, since 2010; the executive director of the International Institute of Applied Informatics since 2011; a guest professor at Bina Nusantara University, Indonesia, since 2015; a research project professor at the Collective Intelligence Research Center, Nagoya Institute of Technology, Japan, since 2015; and an invited professor at the University of Nevada at Las Vegas, USA, since 2016. He has been a full professor at the Advanced Institute of Industrial Technology (AIIT) since 2012. His current research interests include electronic commerce and business, service science and marketing, business management, artificial intelligence, material informatics, tourism informatics, convention administration research, and incentive design on e-services. He can be contacted at email: matsuo@aiit.ac.jp.