

Received 12 April 2023, accepted 16 May 2023, date of publication 29 May 2023, date of current version 16 June 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3280540

RESEARCH ARTICLE

A CNN-OSELM Multi-Layer Fusion Network With Attention Mechanism for Fish Disease Recognition in Aquaculture

YO-PING HUANG^{1,2,3,4}, (Fellow, IEEE),
AND SIMON PETER KHABUSI¹, (Graduate Student Member, IEEE)

¹Department of Electrical Engineering, National Taipei University of Technology, Taipei 10608, Taiwan

²Department of Electrical Engineering, National Penghu University of Science and Technology, Penghu 88046, Taiwan

³Department of Computer Science and Information Engineering, National Taipei University, Taipei 23741, Taiwan

⁴Department of Information and Communication Engineering, Chaoyang University of Technology, Taichung 41349, Taiwan

Corresponding author: Yo-Ping Huang (yphuang@gms.npu.edu.tw)

This work was supported in part by the National Science and Technology Council, Taiwan, under Grant MOST108-2221-E-346-006-MY3 and Grant MOST111-2221-E-346-002-MY3.

ABSTRACT The increasing global population has escalated the demand for fish products. This calls for a stable supply which can only be met through improved aquaculture practices. The automatic recognition of fish diseases from diseased underwater images is one of such practices that aims to control disease, improve fish production and optimize profits. However, due to lack of public fish disease dataset, there has been limited research towards fish disease recognition. Moreover, due to low quality of underwater images and complex underwater environments, traditional hand-designed feature extraction methods or convolutional neural networks (CNNs)-based classifiers cannot adequately recognize fish diseases in real underwater scenes. Therefore, this paper proposes a novel hybrid approach based on multilayer fusion, attention mechanism and online sequential extreme learning machine (OSELM) to recognize fish diseases in aquaculture. We further compare the classification performance of the proposed model with baseline, attention-based, ConvNeXt and Swin transformer models. Feature extraction is enhanced by integrating same level features and focusing on salient features for fish disease recognition. The characteristic information of fish disease is refined by using strongly discriminative features of the infected fish regions and weakening regions of low interest using convolutional block attention module (CBAM). The module is added to the multilayer fusion network to sequentially infer attention maps along the channel and spatial dimensions for every intermediate feature map. Fish disease recognition is then done using OSELM for faster learning and improved classification performance. The models are trained, validated and tested on a custom dataset with image samples collected from various internet sources. The proposed method achieves 94.28% of accuracy, precision of 92.67%, recall of 92.17% and 92.42% of F1- score on dataset with background elimination. The proposed method can be used for fish disease identification in complex underwater environments in aquaculture.

INDEX TERMS Fish disease, aquaculture, multilayer fusion, extreme learning machines, attention mechanism.

I. INTRODUCTION

The global aquaculture production has been growing steadily over the last six decades, increasing the total aquaculture

The associate editor coordinating the review of this manuscript and approving it for publication was Laura Celentano¹.

production share of the total fisheries from 4% in the 1950s, 5% in 1970s, 20% in the 1990s, and 44% in the 2010s. The total fish capture in 2020 was 179 million tonnes, of which 88 million tonnes were from aquaculture, accounting for 47% and US\$265 billion of total revenue. This expansion has greatly improved the overall production in inland waters.

The percentage contribution of aquaculture to global fish production and the contribution of aquaculture to the global food fish consumption are expected to reach 53% and 59% by 2030, respectively [1].

In 2019, the global production of Salmon fish from aquaculture had reached 2.6 million after a 7% increase [2]. The annual growth rate is therefore estimated at 8.8% globally, making it the most dynamic food sector. However, the current annual growth can even be doubled with appropriate aquaculture practices including timely detection of diseases and control [3] since they are one of the major threats to fish production. Fish diseases affect both yield and quality of fish. Disease infestation on an aquarium results into increased fish mortality, low breeding rates, reduced quality of fish, and economic losses. The natural environment is no exception.

Statistics show that disease outbreaks cause an annual financial loss of US\$6 billion to global aquaculture industry, which depicts them as a major risk at farm-level [2]. The shrimp industry has suffered losses of over US\$10 billion since 1990 and new diseases appear every year. For example, Vietnam alone has reported losing an average of US\$1 billion per year to disease. Mortality due to disease accounted for up to 70% of the total losses on a marine salmon farm in Norway [2]. The food and agricultural organization of the United Nations report of 2022 lists disease control and biosecurity as one of the important areas for global transformation of aquaculture.

Fish diseases are caused by a variety of contagious parasitic organisms such as virus, bacteria and protozoa which are associated with at least 18 diseases that affect most of the fish species in fish farms [3]. The affected fishes develop physical and/or behavioral signs and symptoms that vary depending on the nature and stage of the disease. The fish recognition task presented in this paper focuses on physical characteristics of the fish to perform disease recognition using a custom fish disease dataset. In this paper, we focus on five fish diseases whose details are presented in Table 1.

It is worth noting that the disease situation in aquaculture is changing rapidly and becoming very difficult to predict due to the prevailing accelerated evolution in international trading environment affected by globalization, increase in aquaculture production, microbial resistance, water pollution and climate change. Other risk factors leading to disease outbreaks on aquaculture farms include high stocking densities, poor water quality, delayed removal of dead fish from the aquarium, and presence of predator birds around the aquarium. Therefore, research initiatives towards fish disease recognition are novel and very critical in the development of smart aquaculture systems.

The recent trends in artificial intelligence and computer vision provide strong possibilities for the design of smart aquaculture systems for early recognition of fish diseases. Fish disease recognition methods can be classified into two major categories, direct and indirect methods. The direct methods are those that utilize the existence of disease causing






bacteria, protozoa or viruses in the aquarium, and fish appearance to detect disease invasion. These include biochemical, histological, spatio-temporal and computer vision approaches. The biological studies involve the use of tests of antibiotic sensitivity for the detection of parasitic viruses and bacteria that cause diseases [6], [7]. However, histological studies entail the microscopic analysis of fish cells and tissue structures [8], [9]. Whereas the spatio-temporal studies use fish activity or movement to predict the fish's wellness and water environment conditions for early prediction of possible disease invasion. The indirect methods of fish disease detection utilize the prevailing conditions of the aquarium and fish behavior to detect possible future disease invasion. Such approaches majorly deal with the quality of water in the aquarium which is influenced by dissolved oxygen levels, pH, temperature, nitrates, vegetation, feeding, and stocking densities, among others [10].

Existing studies on indirect approaches to fish disease identification include detection of anomalous behavior [11], prediction of water quality [12] and dissolved oxygen [13], [14], [15], [16], [17]. The computer vision-based direct approaches to fish disease detection utilize the physical parameters or physical appearance of fish such as body texture, eye color, appearance of fish head, fish fins, scales, gills and tail to recognize disease. However, such studies are scanty. Moreover, their scope is limited to epizootic ulcerative syndrome (EUS) or redspot disease. This is attributed to various challenges such as unavailability of public dataset of fish disease, limited feature extraction and classification ability of baseline deep learning and machine learning models, complex underwater environments and minute nature of visual presence of fish diseases limited by the quality of underwater images. The complex underwater environment is a source of various interferences caused by brightness imbalances, abrupt fish positional changes, movement of aquatic plants, fish texture and shape, and structure of the seabed [18] which are a major challenge to fish disease recognition.

This study therefore proposes a CNN-OSELM multilayer fusion network with attention mechanism for fish disease recognition in aquaculture. We develop a multilayer fusion network using same level composition (SLC) [19], [20], [21] and integrate attention mechanism to improve feature extraction and focus on strongly discriminative features of the infected fish regions while weakening regions of low interest. Classification is performed by the OSELM, instead of the dense layer, to improve the classification performance. OSELM algorithm utilizes the concepts of extreme learning machine (ELM) developed by [24]. The ELM algorithm was developed for single-hidden layer feedforward neural networks (SLFNs) and has shown to perform extremely fast and with good generalization performance. The main contributions of this study are four-fold.

- 1) Compared with the conventional backbone networks used for feature extraction in image recognition, this paper incorporates SLC structure to build CNN

TABLE 1. Overview of fish diseases considered in the study.

Disease	Preview	Description
EUS		<ul style="list-style-type: none"> - EUS is caused by an oomycete scientifically referred to as <i>Aphanomyces invadans</i> [4]. - The disease manifests as ulcer or red spots. - Affects various fish species. - Conventional control is by adding lime to the aquarium and sun-drying.
Whitespot		<ul style="list-style-type: none"> - Contagious viral disease caused by <i>Ichthyophthirius multifiliis</i>. It is also referred to as the white spot syndrome. - Parasite feeds on fish's body fluids and cells resulting into tissue damage and eventual death [6]. - Fish develop white spots on its skin and is prevalent among crustaceans of order decapoda such as shrimps, lobsters, prawns, and crabs though other fish species are also affected.
Argulus		<ul style="list-style-type: none"> - Belongs to Argulidae family which are parasitic and disease-causing form of fish lice [5]. - The lice attach on the fish skin, suck blood and body nutrients leading to fish malnutrition and death. - Lice is 7mm long by 5mm wide but can be visually seen by a high-resolution underwater camera. - Affects both freshwater and marine fish.
Diplostomiasis		<ul style="list-style-type: none"> - Also known as black spot disease is a parasitic flatworm that attaches on fish fins and skin. - Manifests as tiny black spots. - The disease compromises the fish meat quality.
Hexamitiasis		<ul style="list-style-type: none"> - Parasitic disease which affects fish in both salty/marine such as surgeonfish and fresh water such as goldfish. - It is also called "hole in the head" disease because the head and flanks develop lesions. - Might also cause erosion of the fish's head and lateral line (HLLE). - Fish's food consumption level drops significantly and the fish gets emaciated gradually and eventually dies.

architecture which is less deep and with improved feature extraction.

- 2) To improve feature extraction and ultimately classification accuracy, this study includes the existing CBAM in the composite network. We further compare the performance of the proposed method with baseline, attention-based, ConvNeXt and Swin transformer models.
- 3) We incorporate OSELM for classification instead of the dense layer to reduce the training time and improve the classification performance. The OSELM network randomly sets the input weights and biases of the hidden layer and only updates the output weights.
- 4) This study is the first to apply the concept of SLC with background elimination to overcome the underwater interference for improving recognition of fish diseases.

The rest of the paper is organized as follows: The existing literature is presented in Section II followed by the proposed approach in Section III. Section IV gives data preprocessing methods and experimentation details. Section V presents and describes the study results. The paper is concluded in Section VI and the future direction of this work is suggested.

II. LITERATURE REVIEW

This section discusses the existing work on fish disease classification and detection, extreme learning machines, attention mechanism and mathematical basis of OSELM.

A. FISH DISEASE RECOGNITION

Using deep learning models for fish disease recognition is a novel research area with very limited existing literature. Among these is a binary classification method to classify

salmon fish infected by EUS in aquaculture [27]. The proposed model was trained on a custom dataset and achieved an average accuracy of 92.8% and 94.3% on augmented data. Similar work has been done in [27] and [28] to classify healthy fish that infected by EUS. The fish details were obtained by segregating fish images, morphological and edge detection operations and important features are determined by principal component analysis (PCA). The whitespot disease has also been studied and identified from a custom dataset through segmentation and achieved a test accuracy of 86% [29]. Moreover, fish disease recognition method using fish body surface features was proposed in [30]. In the study, structural segmentation and contour extraction were done using dual-threshold difference method. A precision value of 92.0% and recall value of 74.2% were achieved.

B. HYBRID CNN-ELM APPROACHES TO IMAGE CLASSIFICATION

The ELM algorithm developed by [24] has been reported to perform extremely fast and with better generalization performance compared to other methods that used batch training. Semi-supervised and active learning have been combined to discover hidden useful information from unlabeled samples in order to enhance classification performance in multiple class dataset [31]. Moreover, traffic sign recognition was proposed in [32] using CNN-ELM. CNN was used as a feature extractor and ELM as a classifier with 4,000 hidden nodes achieving a recognition rate of 99.23%. An extreme learning machine based on L1-norm minimization was proposed in [33] to detect disease on peach and strawberry leaves. The proposed algorithm, L1-ELM, was a feature-based detection method with higher generalization capability and lower

learning compared to the conventional methods. The proposed approach achieved an overall accuracy of 98.5% on peach leaves and 97.7% on strawberry leaves compared to SVM which achieved accuracy of 93.0% on peach leaves and 92.3%, the highest amongst the conventional classifiers. Similar studies have been done for pneumonia classification in [34]. An ELM that adjusted output weights by increasing the output weight's corresponding column was proposed in [35]. Images of hyperspectral remote sensing were classified using ELM and based on hierarchical local-receptive-field [36]. Other related studies using composite kernels and ensemble learning have been implemented for both image classification and regression [37], [38], [39], [40], [41]. Moreover, a neural response approach on the basis of extreme learning machine has been proposed for image classification, tested on MNIST dataset, Caltech face dataset and CIFAR-10 dataset and achieved the highest accuracies of 99.18%, 93.93% and 42.43% (for 100 training images in every class), respectively, [42].

C. ATTENTION MECHANISM

There have been enormous efforts to improve the performance of CNNs in image recognition by integrating attention mechanisms into the backbone networks [25], [26]. For example, a residual encoder/decoder attention network was proposed and used for feature map refinement [25]. In this regard, attention was a network engineering concept that aimed to utilize more important features and ignored the less important ones for improved classification, recognition or object detection performance.

To improve the recognition performance of fish disease, our work focuses on emphasizing important regions and weakening those that are less important by integrating attention mechanism in the proposed multilayer fusion network. CBAM has been integrated in the proposed network [44]. The 3-dimensional attention maps of CBAM are decomposed to learn separately via two modules: channel attention and spatial attention. The spatial and channel attention modules are placed sequentially between convolutional blocks starting with the channel attention. The feature map produced by the intermediate CNN blocks is refined adaptively at every block of the deep networks. This arrangement is adopted because it is reported to achieve better results [44].

For input $F \in \mathbb{R}^{C \times H \times W}$ (CNN intermediary feature map) to the block, the module generates an attention map $M_c \in \mathbb{R}^{C \times 1 \times 1}$ (1-dimensional) and spatial attention feature map $M_s \in \mathbb{R}^{1 \times H \times W}$ (2-dimensional). The process of attention is summarized as follows:

$$F' = M_c(F) \otimes F \quad (1)$$

$$F'' = M_s(F') \otimes F' \quad (2)$$

The element-wise multiplication is denoted as \otimes . The multiplication process involves broadcasting (copying) the values of the attention maps accordingly. The values of the channel attention are copied along the spatial dimension and vice

versa, generating F'' as the refined final output value as shown in Fig. 1.

In the channel attention module, the inter-channel relationships between the features are exploited to produce the channel attention maps. For a given input image, the channel attention focuses on the meaningful aspects of the image; hence, every channel of intermediate feature map is taken to be a detector of features of interest. The input feature map's spatial dimension is squeezed to achieve efficient computation of the channel attention. It is noted in the existing works that utilizing both max and average pooling considerably improves the representation ability of networks instead of using one separately. Therefore, CBAM uses both max and average pooling to aggregate the spatial information for a feature map. This generates two different descriptors of spatial context, F_{max}^c and F_{avg}^c , the spatial features resulting from max and average pooling operations, respectively. The two spatial context features are then passed on to a fully connected shared network producing $M_c \in \mathbb{R}^{C \times 1 \times 1}$, which is channel attention map. The shared network is the multilayer perceptron (MLP) with a single hidden layer. The parameter overhead is reduced by setting activation size of the hidden layer to $\mathbb{R}^{C/r \times 1 \times 1}$, where r is a reduction ratio. The MLP is then applied to each of the descriptors and feature vectors of the output are then merged by element-wise addition. In summary, computation of channel attention feature map is as follows:

$$\begin{aligned} M_c(F) &= \sigma(MLP(AvgPool(F)) + MLP(MaxPool(F))) \\ &= \sigma(W_1(W_0(F_{avg}^c)) + W_1(W_0(F_{max}^c))) \end{aligned} \quad (3)$$

The sigmoid function is denoted by σ , $W_0 \in \mathbb{R}^{C/r \times C}$ and $W_1 \in \mathbb{R}^{C \times C/r}$ are MLP weights and are shared for both inputs. The activation function (ReLU) is followed by W_0 .

On the other hand, the spatial attention module produces the spatial attention map using inter-spatial relationships within features. The spatial attention concentrates on finding the informative part that is consistent with the channel attention. Computation of spatial attention involves the application of max and average pooling operations along channel axis and then the results are concatenated in order to obtain a systematic feature descriptor. A convolution layer is then applied on the feature descriptor to produce the spatial attention map $M_s(F) \in \mathbb{R}^{H \times W}$. The attention map encodes to suppress or emphasize the regions as shown in Fig. 1.

The channel information of the feature map is combined by max and average pooling operations. This generates two 2-dimensional feature maps, $F_{avg}^s \in \mathbb{R}^{1 \times H \times W}$ and $F_{max}^s \in \mathbb{R}^{1 \times H \times W}$. The two generated features are then concatenated and followed by convolutional operation using the standard convolutional layer. This produces the attention map of two dimensions. The computation of the spatial attention is given below.

$$\begin{aligned} M_s(F) &= \sigma(f^{7 \times 7}([AvgPool(F); MaxPool(F)])) \\ &= \sigma(f^{7 \times 7}([F_{avg}^s; F_{max}^s])) \end{aligned} \quad (4)$$

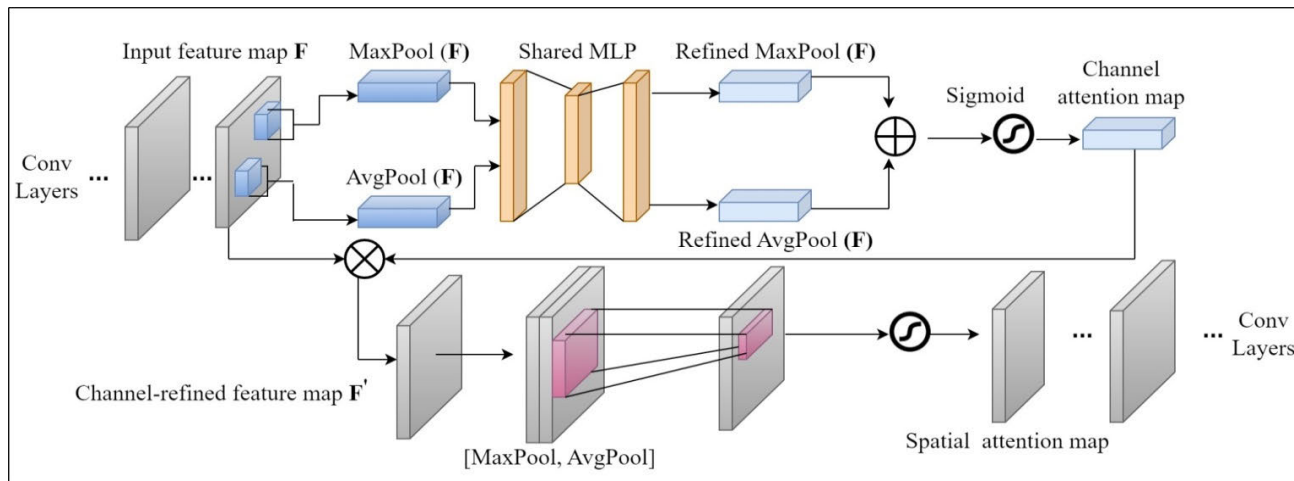


FIGURE 1. Architectural structure of CBAM.

where σ is a sigmoid function and $f^{7 \times 7}$ refers to convolution operation using 7×7 filter size. The overall attention process is shown in Fig. 1.

D. ONLINE SEQUENTIAL EXTREME LEARNING MACHINE (OSELM)

This section presents the mathematical basis for single hidden layer feedforward neural network (SLFN) to give the foundation for the design of OSELM. The OSELM algorithm was a sequential learning algorithm with higher versatility and achieved through four main operational principles. First, the training samples were presented in succession, one sample or one chunk at a time and the size was fixed or varied. Secondly, the learning process was done on the newly arrived single chunk or chunks of data samples at every single moment rather than on all the past data. Thirdly, a data sample or chunk was discarded immediately after the model completed learning from it. Lastly, the number of training observations was not known by the algorithm prior to training [23]. The OSELM was built on the operational basis of existing batch ELM [24].

Given a SLFN with x RBF or additive hidden nodes and P data samples, the representation of its output is given by:

$$f_x(l) = \sum_{i=1}^x \beta_i U(w_i, b_i, l), \quad l \in R^p, w_i \in R^p \quad (5)$$

where w_i and b_i are the learning parameters of hidden nodes, β_i denotes the weight that connects the i th node of the hidden layer to the output node and $U(w_i, b_i, l)$ denotes the i th node of the hidden layer output with respect to its input l . However, given additive hidden node with activation function $u(l): R \rightarrow R$ (such as sigmoid or threshold), $U(w_i, b_i, l)$ is given by:

$$U(w_i, b_i, l) = u(w_i \cdot l + b_i), \quad b_i \in R \quad (6)$$

where w_i is the weight vector that connects the input layer to the i th hidden node and b_i is the bias of the i th hidden node. $w_i \cdot l$ is the inner product of w_i and l vectors in R^p .

RBF network is a special kind of SLFN whose hidden layer consists of RBF nodes with a centroid and impact factor. The output of every node is given by a radially symmetric function of the distance between the input and the center of the node.

For the hidden node of RBF whose activation function is $u(l): R \rightarrow R$ (such as Gaussian), the hidden layer output with respect to its input l becomes:

$$U(w_i, b_i, l) = u(b_i \|l - w_i\|), \quad b_i \in R^+ \quad (7)$$

where w_i denotes the center of i th node of the RBF network and b_i is its impact factor. R^+ denotes a set of all positive real values.

The ELM is a form of supervised learning algorithm. Suppose a dataset consists of P arbitrary distinct data points $\{(l(i), t(i))\}_{i=1}^P$, such that $l(i)$ is an input vector of dimension $p \times 1$ and $t(i)$ is a target vector of $m \times 1$ dimension. The standard SLFN with x number of hidden nodes and activation function $u(l)$ can be modeled such that:

$$f_x(l_j) = \sum_{i=1}^x \beta_i u(l_j) w_i b_i = t_j, \quad j = 1, \dots, P \quad (8)$$

When the SLFN outputs are equal to the targets, the following compact formulation is obtained:

$$H\beta = T \quad (9)$$

where

$$H(w_1, \dots, w_x, b_1, \dots, b_x, l_1, \dots, l_p) = \begin{bmatrix} U(w_1, b_1, l_1) & \dots & U(w_x, b_x, l_1) \\ \vdots & \dots & \vdots \\ U(w_1, b_1, l_p) & \dots & U(w_x, b_x, l_p) \end{bmatrix}_{P \times x} \quad (10)$$

$$\beta = [\beta_1^T \dots \beta_x^T]^T, \text{ and } T = [t_1^T \dots t_p^T]^T_{P \times m} \quad (11)$$

H is the output matrix of the hidden layer and superscript T denotes the transpose of the matrix.

The hidden nodes parameters can be assigned randomly when the training samples (P) and hidden layer nodes (x) are

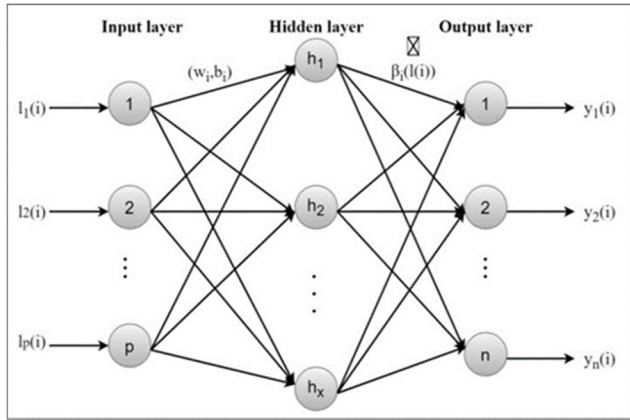


FIGURE 2. Architecture of ELM.

the same, i.e., when $P = x$. Then, output weights can be analytically calculated by merely inverting H which lowers the error rate to minimal value.

However, if the hidden layer nodes are less than the training samples, that is $x < P$, the hidden nodes parameters can nevertheless be randomly assigned. Conversely, the output weights can be calculated by using pseudoinverse of H which leads to a smaller training error $\epsilon > 0$. The training procedure is likewise reduced since the weights' calculation is done in a single step. The architecture of ELM is shown in Fig. 2.

The OSELM algorithm is a two-phase algorithm consisting of the initialization phase and sequential learning phase. The initialization phase involves filling up the appropriate matrix $(H_0) = x$ to be used in the learning phase. The least number of data samples needed for filling up H_0 should be at least equal to the number of hidden nodes.

Given an initial training chunk of data $\rho_0 = \{(l_i, t_i)\}_{i=1}^{P_0}$ and $P_0 \geq x$, the target is to minimize $\|H_0\beta - T_0\|$ achieved by $\beta^{(0)} = K_0^{-1}H_0^T T_0$ where $K_0 = H_0^T H_0$.

$$H_0 = \begin{bmatrix} U(w_1, b_1, l_1) & \cdots & U(w_x, b_x, l_1) \\ \vdots & \cdots & \vdots \\ U(w_1, b_1, l_{P_0}) & \cdots & U(w_x, b_x, l_{P_0}) \end{bmatrix}_{P_0 \times x}$$

$$T_0 = [t_1^T \cdots t_{P_0}^T]_{P_0 \times m}^T \quad (12)$$

Suppose another chunk of training data $\rho_1 = \{(l_i, t_i)\}_{i=P_0+1}^{P_0+P_1}$ is given, where the number of observations in this chunk is P_1 , the problem then requires to minimize (12) as follows.

$$\left\| \begin{bmatrix} H_0 \\ H_1 \end{bmatrix} \beta - \begin{bmatrix} T_0 \\ T_1 \end{bmatrix} \right\| \quad (13)$$

$$H_1 = \begin{bmatrix} U(w_1, b_1, l_{P_0+1}) & \cdots & U(w_x, b_x, l_{P_0+1}) \\ \vdots & \cdots & \vdots \\ U(w_1, b_1, l_{P_0+P_1}) & \cdots & U(w_x, b_x, l_{P_0+P_1}) \end{bmatrix}_{P_1 \times x}$$

$$T_1 = [t_{P_0+1}^T \cdots t_{P_0+P_1}^T]_{P_1 \times m}^T \quad (14)$$

Equation (13) holds true, and when we consider the two chunks ρ_0 and ρ_1 of the training data sets, the output weight

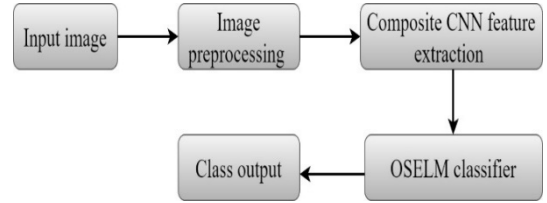


FIGURE 3. Process flow.

β evaluates to:

$$\beta^{(1)} = K_1^{-1} \begin{bmatrix} H_0 \\ H_1 \end{bmatrix}^T \begin{bmatrix} T_0 \\ T_1 \end{bmatrix} \quad (15)$$

where

$$K_1 = K_1^{-1} \begin{bmatrix} H_0 \\ H_1 \end{bmatrix}^T \begin{bmatrix} H_0 \\ H_1 \end{bmatrix} \quad (16)$$

For sequential learning, output weight $\beta^{(1)}$ is expressed as a function of $\beta^{(0)}$, K_1 , H_1 , and T_1 and not that of chunk ρ_0 of training data. K_1 is rewritten as

$$K_1 = \begin{bmatrix} H_0^T & H_1^T \end{bmatrix} \begin{bmatrix} H_0 \\ H_1 \end{bmatrix} = K_0 + H_1^T H_1 \quad (17)$$

$$\begin{aligned} \begin{bmatrix} H_0 \\ H_1 \end{bmatrix}^T \begin{bmatrix} T_0 \\ T_1 \end{bmatrix} &= H_0^T T_0 + H_1^T T_1 \\ &= K_0 K_0^{-1} H_0^T T_0 + H_1^T T_1 \\ &= K_0 \beta^{(0)} + H_1^T T_1 \\ &= (K_1 - H_1^T H_1) \beta^{(0)} + H_1^T T_1 \\ &= K_1 \beta^{(0)} - H_1^T H_1 \beta^{(0)} + H_1^T T_1 \end{aligned} \quad (18)$$

Combining (15) and (18), $\beta^{(1)}$ becomes

$$\begin{aligned} \beta^{(1)} &= K_1^{-1} \begin{bmatrix} H_0 \\ H_1 \end{bmatrix}^T \begin{bmatrix} T_0 \\ T_1 \end{bmatrix} \\ &= K_1^{-1} (K_1 \beta^{(0)} - H_1^T H_1 \beta^{(0)} + H_1^T T_1) \\ &= \beta^{(0)} + K_1^{-1} H_1^T (T_1 - H_1 \beta^{(0)}) \end{aligned} \quad (19)$$

where K_1 is given by

$$K_1 = K_0 + H_1^T H_1 \quad (20)$$

A recursive least squares algorithm is similar to the recursive updation algorithm for least squares solution. When the $(K + 1)$ th chunk, ρ_{k+1} is received, (21)–(23), as shown at the bottom of the next page. Equation (22) and (23) hold.

Instead of K_{k+1} , K_{k+1}^{-1} is used for the computation of $\beta^{(k+1)}$ from $\beta^{(k)}$ in (21). The derivation for the updation formula for K_{k+1}^{-1} is made using Woodbury formula [52].

$$\begin{aligned} K_{k+1}^{-1} &= (K_k + H_{k+1}^T H_{k+1})^{-1} \\ &= K_k^{-1} - K_k^{-1} H_{k+1}^T (I + H_{k+1} K_k^{-1} H_{k+1}^T)^{-1} \\ &\quad \times H_{k+1} K_k^{-1} \end{aligned} \quad (24)$$

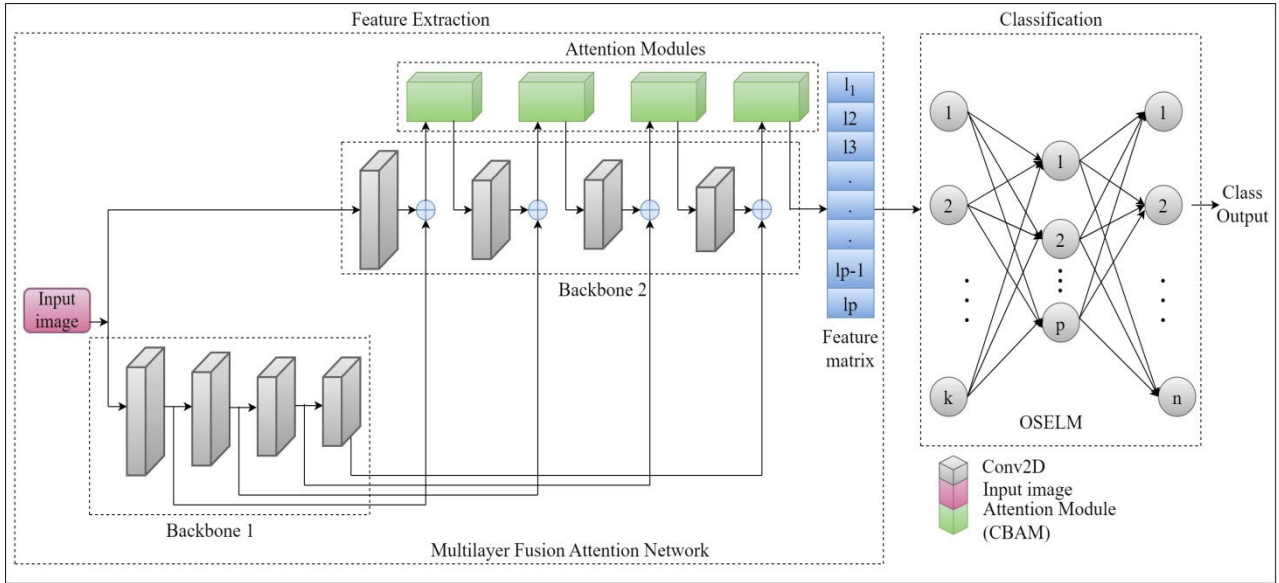


FIGURE 4. The proposed architecture.

Let $A_{k+1} = K_{k+1}^{-1}$, the updation equations for $\beta^{(k+1)}$ can then be rewritten as

$$\begin{aligned} A_{k+1} &= A_k - A_k H_{k+1}^T (I + H_{k+1} A_k H_{k+1}^T)^{-1} H_{k+1} A_k \\ \beta^{(k+1)} &= \beta^{(k)} + A_{k+1} H_{k+1}^T (T_{k+1} - H_{k+1} \beta^{(k)}) \end{aligned} \quad (25)$$

Equation (25) gives the recursive formula for $\beta^{(k+1)}$.

III. PROPOSED METHOD

The existing transfer learning models are not sufficient in fish disease classification because they do not have the ability to emphasize salient features and weaken meaningless regions which lowers the classification performance. In fish disease classification task, the idea is to focus on key regions that present discriminative features for the various fish diseases. Additionally, the deep sequential networks do not have the ability to integrate and enrich the feature maps during feature extraction. This paper therefore uses the keras functional API that allows inputs which are a list of tensors of the same shape other than the concatenation axis, and concatenates them to return a single tensor. This strengthens the feature maps and improves feature extraction.

Given an input image, preprocessing is done to remove noise and make the image consistent with the neural network input specifications. The multi-layer fusion network then performs feature extraction whereas the attention module suppresses the meaningless regions and strengthens regions of higher interest for feature refinement. The output features are then flattened and fed to the OSELM network for classification. This flow of processes is shown in Fig. 3.

A. MODEL NETWORK STRUCTURE

The proposed method comprises of convolutional blocks that consist of Conv2D layers, pooling layers, dropout and batch normalization layers, ReLU, softmax activation and OSELM classification layer. The preprocessed image data is input to two parallel backbone networks. The output feature map is obtained by:

$$v_i^p = f \left(\sum_{i \in M_k} v_i^{p-1} \cdot W_{ik} + b_i \right) \quad (26)$$

$$\begin{aligned} \rho_{k+1} &= \{(l_i, t_i)\}_{i=(\sum_{j=0}^k P_j)+1}^{\sum_{j=0}^{k+1} P_j}, \quad K_{k+1} = K_k + H_{k+1}^T H_{k+1}, \\ \beta^{(k+1)} &= \beta^{(k)} + K_{k+1}^{-1} H_{k+1}^T (T_{k+1} - H_{k+1} \beta^{(k)}) \end{aligned} \quad (21)$$

$$H_{k+1} = \begin{bmatrix} U(w_1, b_1, l_{(\sum_{j=0}^k P_j)+1}) & \cdots & U(w_x, b_x, l_{(\sum_{j=0}^k P_j)+1}) \\ \vdots & \cdots & \vdots \\ U(w_1, b_1, l_{\sum_{j=0}^{k+1} P_j}) & \cdots & U(w_x, b_x, l_{\sum_{j=0}^{k+1} P_j}) \end{bmatrix}_{P_{k+1} \times x} \quad (22)$$

$$T_{k+1} = \left[t_{(\sum_{j=0}^k P_j)+1}^T \cdots t_{\sum_{j=0}^{k+1} P_j}^T \right]_{P_{k+1} \times m}^{Transpose} \quad (23)$$

TABLE 2. Details of convolutional operations of the proposed method.

Layers	Backbone 1 Parameters	Backbone 1 Parameters
Input	256 × 256 × 3	256 × 256 × 3
Block 1	Conv2D_1, 2 Filters: 3 × 3 *32 Shape: 224×224×32 MaxPool: 2×2 Shape: 112, 112, 32	Conv2D_1 Filters: 3 × 3 *32 Shape: 224×224×32 MaxPool: 2×2 Shape: 112, 112, 32
Block 2	Conv2D_1, 2 Filters: 3 × 3 *1*64 Shape: 112, 112, 64 MaxPool: 2×2 Shape: 56, 56, 64	Conv2D_1 Filters: 3 × 3 *1*64 Shape: 112, 112, 64 MaxPool: 2×2 Shape: 56, 56, 64
Block 3	Conv2D_1, 2,3,4 Filters: 3 × 3 *1*128 Shape: 56, 56, 128 MaxPool: 2×2 Shape: 28, 28, 128	Conv2D_1 Filters: 3 × 3 *1*128 Shape: 56, 56, 128 MaxPool: 2×2 Shape: 28, 28, 128
Block 4	Conv2D_1, 2,3,4 Filters: 3 × 3 *1*256 Shape: 28, 28, 256 MaxPool: 2×2 Shape: 7, 7, 256	Conv2D_1 Filters: 3 × 3 *1*256 Shape: 28, 28, 256 MaxPool: 2×2 Shape: 7, 7, 256
FC-1	(None, 12544)	(None, 12544)
Softmax		Input vector size:
OSELM		12544
		Hidden nodes: 4,096
Output		Class Output

Algorithm 1 Algorithm 1. The proposed method

Input: RGB image set $v_i, t_i, i = 1, 2, \dots, N$
Output: CNN feature extractor $f(v_i)$, output matrix H_k that maps channel and spatial features M_{it} of image v_i to its corresponding class label t_i .

- 1) Input RGB image of size 224×224 .
- 2) Perform convolutional operations with attention to generate feature matrix $\{(l(i), t(i))\}_{i=1}^P$.
- 3) Remove the dense layers of feature extractor.
- 4) Initialize model learning using a small chunk of initial training sample matrix ρ_0 the training set.
- 5) Compute H_0 , through (12) the output matrix of the hidden layer.
- 6) Estimate the initial output weight $\beta^{(0)}$ and set $k=0$.
- 7) Calculate the output matrix H_{k+1} through (22) of for the $(k+1)$ th chunk of samples ρ_{k+1} .
- 8) Set $T_{k+1} = [t_{(\sum_{j=0}^k p_j)+1}, \dots, t_{(\sum_{j=0}^{k+1} p_j)}]^T$.
- 9) Calculate the output weight $\beta^{(k+1)}$ though (25).
- 10) Set $k = k + 1$.

return the CNN feature extractor $f(v_i)$

where p refers to the convolutional layers, W_{ik} is the convolutional kernel, b_i is the bias and V_i^{p-1} is the input channel map, and $f()$ represents the activation function. ReLU activation is used in the succeeding blocks since it is nonlinear and unsaturated. It can reduce training time, solve over fitting and provide fast convergence [46].

The proposed network builds from parallel multiple backbones that are identical to each other. This enables the integration of same level features of the identical backbones for feature refinement. The features output by blocks 1, 2, ..., n

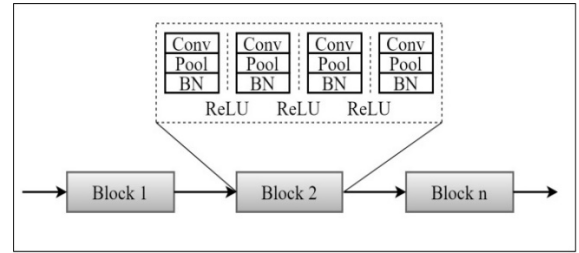


FIGURE 5. Detailed structure of CNN blocks.

in backbone 1 are concatenated by the output features which are of the same size from the corresponding blocks in backbone 2 and refined by the corresponding attention block. Conventionally, the design procedure of most CNNs convolves channel maps of the input images into their respective intermediate features whose resolution is monotonically lower. In our network, the $(s+1)$ th block of backbone 2 takes the refined feature map F_2^{att+1} resulting from the concatenation of output, x_1^s of the s -th block of backbone 1 and output x_2^s of the s -th block of backbone 2 and refined by attention block 1, that is:

$$F_2^{att+1} = F_1^s(x^s) \oplus F_2^s(x^s) \tag{27}$$

After the convolution operations, OSELM is then used for the classification of the 1-D vectors obtained after converting the output feature maps from the attention based multilayer convolutional network. The OSELM network randomly sets the input weights and biases of the hidden layer whereas only the output weights are updated. The input parameters are randomly generated and only the output weights are calculated during the model training stage. The entire process with no iteration operation generally improves the generalization ability of the neural network. The output 12544×1 of the CNN from the last layer is the input to the OSELM as elaborated in Table 2.

Although it has been proven theoretically in [23] that so long as ELM models have as many nodes in the hidden layer as possible, they are able to make approximation of any continuous target functions with any degree of accuracy. It has been experimented that random addition of hidden layer nodes did not guarantee better performance of ELM [54]. The recommendation is to automatically determine the initial hidden layer node number by using incremental constructive techniques such as incremental extreme learning machine (I-ELM) [51] and enhanced incremental ELM (EI-ELM) [14], among others and then optimizing the initial model in order to obtain optimal network. In this study we adopt the number of hidden nodes as 4096 because it has been experimented on similar output size in [54] and proven to perform well. This also saves the computation cost and training time. The proposed architecture is shown in Fig. 4 and the expanded structure of the CNN network blocks is shown in Fig. 5. The proposed method is shown in Algorithm 1.

B. PERFORMANCE EVALUATION METRICS

The classification performance has been determined using precision, accuracy, recall, and F-measure/score from the confusion matrix with TP, TN, FP, and FN, referring to true positive, true negative, false positive, and false negative, respectively. TP is prediction in which the predicted class label on the test data indeed correlates with the class of the ground truth image. FP refers to a prediction in which the predicted class does not belong to the class of the ground truth image. TN refers to a prediction in which the model predicts that a given test image does not belong to a given class and it is indeed true and FN refers to a prediction where the test image belongs to the ground truth image class but the model predicts it as not belonging to that class.

1) ACCURACY

This is one of the ways to evaluate model performance. It is the ratio of correctly predicted classifications to the overall number of predictions made.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (28)$$

2) PRECISION

This refers to the proportion of accurately classified samples in relation to the TP and FP values. In other words, it is a computation of the percentage of samples that are correctly classified.

$$Precision = \frac{TP}{TP + FP} \quad (29)$$

3) RECALL/SENSITIVITY

This refers to the ratio of TP to the sum of TP and FN.

$$Recall = \frac{TP}{TP + FN} \quad (30)$$

4) F1-SCORE

This is calculated as the symphonic mean of precision and recall.

$$F1 - score = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (31)$$

IV. DATASET AND PREPROCESSING

This section presents the details of the dataset used in the study and the preprocessing stages undertaken to make the data samples compliant with network specifications.

A. DATASET

The fish disease dataset is a custom dataset that has been collected from various internet sources. It includes a total of 649 low resolution images of infected fish and non-infected fish. The images have been captured in both aquariums and wild environments and are distributed into 6 classes, EUS, white spot disease, Diplopstomiasis, Argulus, hexamitiasis and healthy fish. The image samples have been augmented using rotation, flipping, and cropping to obtain a total of 5,165 image samples as shown in Table 3.

TABLE 3. Dataset details.

Class	Training (60%)	Validation (20%)	Test (20%)
EUS	451	151	151
Whitespot	648	216	216
Argulus	353	117	117
Diplopstomiasis	538	180	180
Hexamitiasis	295	98	98
Healthy	814	271	271
Total	3,099	1,033	1,033

The complex background environment poses a major challenge during feature extraction and disease identification by the learning algorithms. A source domain backbone to learn background information features was proposed and used to subtract the source domain information from the feature map of the entire image [19]. However, the presented method was not feasible without having a mechanism to initially separate the background from the entire image. In this paper, we eliminate the image background before passing the image data to the neural network in order to overcome noise and improve feature extraction as discussed in the next subsection.

The interferences caused by complex underwater environment makes the classification task very challenging due to the following issues:

1) LOW-RESOLUTION UNDERWATER IMAGES

The fish's texture feature information is lost due to low quality images which make it challenging in identifying fish diseases with similar characteristics such as Argulus and Diplopstomiasis.

2) COMPLEX SEABED BACKGROUND

The various images collected from the internet are from diverse sources including onshore and offshore aquariums with varying seabed and dynamic plant textures. This reduces the feature extraction ability of CNNs and make it difficult to identify fish disease.

3) CAMOUFLAGE COLOR

The appearance of some fish diseases on the body surfaces of the fish is identical with the color of the background. The fish skin affected by disease blends well with such environments and makes it hard to extract important features.

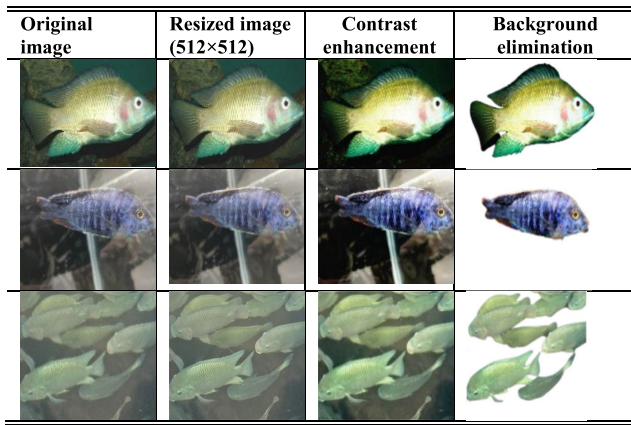
4) OCCLUSION

The fish's texture feature information is considerably lost due to obstruction from various objects and light reflection. This affects fish disease identification and increases the number of false positive predictions.

5) FISH DIVERSITY

The fish images collected have a variety of fish species which presents different patterns in the learning process. For example, there are many fish types in the collection of EUS disease. It would be easier to classify EUS among fish of the same species.

TABLE 4. Sample image preprocessing stages.



6) COMPLEX AND CHANGEABLE UNDERWATER ENVIRONMENT

The images collected from internet sources were collected at various times, scenes, and light intensities using cameras of various pixel sizes. The collected images vary in quality and light intensities which were not conducive for model learning that reduced the generalization ability of the model.

B. DATA PREPROCESSING

1) BACKGROUND ELIMINATION

The complex image backgrounds are a source of noise which over fits the model and lowers the classification performance. These backgrounds also give varying contrasts to the fish disease features and further interfere with the identification of important features for classification. In this paper, the image backgrounds have been eliminated to give uniform background to all the sample images for accurate model training. There exist a number of background removal techniques in image processing which are classified into edge-based techniques such as canny edge detection; foreground detection and machine learning based approaches such as thresholding and clustering, and deep learning approaches. These approaches have been reported to perform well on grayscale images but not on RGB images and the foreground pixels using the mean absolute deviation (MAD) tends to generate some isolated pixels (holes) in the target object interiors. We enhance the RGB color space algorithm for background subtraction by initially applying a foreground detection technique using machine learning selfie segmentation with media pipe [20], [55]. The selfie segmentation function assigns a float number in the range of [0.0, 1.0] to every pixel of the output mask. When the number assigned to each pixel is closer to 1.0, it indicates greater confidence for the pixel to represent the object of interest which in this case is fish. By contrast, the closer the value to 0, the more the confidence that the given pixel belongs to the background and is eliminated. The selfie segmentation then produces an output mask of the same image size as the input. However, due to the nature of underwater diseased fish images, the backgrounds are not completely eliminated. The output images of selfie

TABLE 5. Performance comparison of proposed method with pretrained models with background elimination.

Model	Acc. (%)	P (%)	R (%)	F (%)	Parameters
VGG19	92.04	90.60	91.00	90.80	143M
ResNet50	92.00	92.80	88.00	90.40	25.6M
EfficientNet	91.60	89.07	88.00	88.54	4.2M
Inceptionv3	90.55	89.00	88.82	88.91	22.2M
Xception	90.40	87.00	91.00	89.00	21.5M
Proposed	94.28	92.67	92.17	92.42	4.7M

Note: Acc.: accuracy, P: precision, R: recall and F: F1-score.

TABLE 6. Performance comparison of proposed method with attention-based pretrained models on dataset with background elimination.

Model	Acc. (%)	P (%)	R (%)	F (%)
VGG19+CBAM	93.38	91.00	91.83	91.50
ResNet50+CBAM	93.81	92.67	89.83	91.25
EfficientNet+ CBAM	92.28	91.00	89.00	90.00
Inceptionv3+ CBAM	91.40	90.00	88.50	89.25
Xception+ CBAM	91.99	89.50	91.00	90.25
Proposed Model	94.28	92.67	92.17	92.42

segmentation still have some sheds of the background. Therefore, these images are further input to the RGB color space algorithm for background elimination.

The RGB color space algorithm starts by assuming $u = (u, v)$, and $p(u)$ is a pixel whose intensity is expressed as $L(u)$:

$$I(u) = [L^R(u), L^G(u), L^B(u)]^K \tag{32}$$

where L^R, L^G and L^B refers to the intensities of the RGB color components (red, green, and blue, respectively). For simplifying the notations, superscript C is used to represent R, G or B .

Suppose $[L_1(u), L_2(u), \dots, L_T(u)]$ denotes K image frames, and $M(u)$ denotes the image that contains each pixel's temporal median, for $0 \leq \alpha < 1$, the α -metrically trimmed mean $\lambda_\alpha^C(u)$ of every RGB component for each pixel u is obtained by calculating temporal average of $I_k^C(u)$, disregarding the biggest $[\alpha K]$ deviations away from the median point. Note that $[.]$ denotes the largest integer function. Formally, taking into consideration an ordering of the differences $|I_k^C(u) - M^C(u)|$, and defining a function $1 \leq f(u, k) \leq K$ which is an integer function that returns the position of $|I_k^C(u) - M^C(u)|$ in such ordering, the α -metrically trimmed mean $\lambda_\alpha^C(u)$ is given by:

$$\lambda_\alpha^C(u) = \frac{1}{K - [\alpha K]} \sum_{t \in Z_\alpha(u)} I_k^C(u) \tag{33}$$

where $Z_\alpha(u) = \{k : f(u, k) \leq K - [\alpha K]\}$. When $\alpha = 0$, the trimmed mean becomes exactly the average and as α approaches 1, the trimmed mean tends towards the median value. The value of α is experimentally set to 0.3 [3]. Moving forward, $\lambda^C(u)$ then refers to the α -metrically trimmed value of mean for the pixel u using $\alpha = 0.3$.

It is necessary to evaluate the distribution of noise around the value of the actual background. This is done using scale

TABLE 7. Class-wise performance comparison of the proposed method with attention-based pretrained models on dataset with background elimination. (unit: %).

Model	Parameter	EUS	WSS	Argulus	Diplopstomiasis	Hexamitiasis	Healthy
VGG19+CBAM	Accuracy	92.73	98.80	82.00	92.00	95.94	98.80
	Precision	92.00	98.00	79.00	90.00	92.00	95.00
	Recall	95.00	96.00	75.00	92.00	95.00	98.00
	F1-score	93.50	97.50	77.00	91.00	93.50	96.50
ResNet50+CBAM	Accuracy	94.44	98.70	82.50	92.40	96.22	98.60
	Precision	93.00	99.00	80.00	93.00	92.00	99.00
	Recall	91.00	96.00	68.00	90.00	96.00	98.00
	F1-score	92.00	97.50	74.00	91.50	94.00	98.50
EfficientNet+ CBAM	Accuracy	90.09	98.30	81.70	91.00	95.30	97.30
	Precision	92.00	97.00	79.00	89.00	93.00	96.00
	Recall	93.00	96.00	67.00	89.00	92.00	97.00
	F1-score	92.50	96.50	73.00	89.00	92.50	96.50
Inceptionv3+ CBAM	Accuracy	92.05	97.40	80.33	89.90	92.70	96.00
	Precision	89.00	98.00	78.00	88.00	92.00	95.00
	Recall	93.00	92.00	70.00	89.00	93.00	94.00
	F1-score	91.00	95.00	74.00	88.50	92.50	94.50
Xception+ CBAM	Accuracy	92.00	98.40	80.35	90.16	95.02	96.00
	Precision	90.00	95.00	73.00	88.00	93.00	98.00
	Recall	93.00	96.00	77.00	90.00	94.00	96.00
	F1-score	91.50	95.50	75.00	89.00	93.50	97.00
Proposed Model	Accuracy	93.58	99.20	82.90	92.16	98.84	99.00
	Precision	92.00	98.00	80.00	91.00	96.00	99.00
	Recall	95.00	100.00	75.00	92.00	96.00	95.00
	F1-score	93.00	99.00	77.00	91.00	96.00	97.00

parameter such as the standard deviation. However, in the color space algorithm, a more robust scale estimator, called the mean absolute deviation (MAD), is used. MAD is defined as:

$$MAD^C(u) = median \left\{ \left| I_k^C(u) - M^C(u) \right| \mid k \in \{1, \dots, K\} \right\} \tag{34}$$

If we assume additive Gaussian noise, the relation $\sigma^C(u) = 1.4826MAD^C(u)$ gives the standard deviation estimate of each RGB component for every pixel. Pixels that belong to the foreground tend to be far from the distribution’s estimated mean. The foreground pixels normally appear in blobs and not isolated in the image. Therefore, we analyze close neighborhoods of each pixel. A pixel u is assigned to the foreground if

$$\sum_{x \in \Omega(u)} w(x) \left| I_k^C(x) - \lambda^C(x) \right| > h \sum_{x \in \Omega(u)} w(x) \sigma^C(x), \tag{35}$$

where $\Omega(u)$ denotes a small close neighborhood with center u , $w(x)$ is a weighting mask for every pixel $\mathbf{u} = (u, v)$, h determines the maximum possible deviation away from the mean value with respect to the standard deviation and the weighted average mask is denoted by w , with a central point whose weight is 4, weight of horizontal and vertical neighbors being 2, and 1 for diagonal neighbors. The neighborhood is set to 3×3 for Ω and $h = 3$.

The implementation has been done in opencv library and the sample preprocessing results are presented in Table 4.

TABLE 8. Performance comparison of proposed method with attention-based pretrained models on dataset without background elimination.

Model	Acc.	P	R	F
VGG19+CBAM	78.86	80.50	78.50	79.50
ResNet50+CBAM	81.25	83.33	74.50	77.50
EfficientNet+CBAM	78.79	77.50	82.50	79.17
Inceptionv3+CBAM	71.65	66.33	73.33	69.33
Xception+CBAM	69.59	66.17	67.50	66.50
Proposed	86.11	82.00	89.00	84.00

2) PIXEL ENHANCEMENT

This is applied to enhance the image contrast by increasing the domain of pixel intensity values.

3) IMAGE RESIZE

All sample images are resized to a dimension of $512 \times 512 \times 3$ to reduce the computational model complexity.

4) DATA AUGMENTATION

We augmented the data using rotation and flipping to increase the data samples.

5) LABELING

The image files were assigned class labels for the classification task.

V. RESULTS AND DISCUSSION

In this paper, 649 image samples from a custom dataset are augmented to obtain a total of 5,165 images and pre-processed for contrast enhancement, background elimination.

TABLE 9. Class-wise performance comparison of proposed method with attention-based pretrained models on dataset without background elimination. (unit: %).

Model	Parameter	EUS	WSS	Argulus	Diplopstomiasis	Hexamitiasis	Healthy
VGG19+CBAM	Accuracy	60.44	84.32	81.00	75.62	85.00	86.79
	Precision	55.00	96.00	79.00	71.00	87.00	95.00
	Recall	52.00	85.00	71.00	80.00	92.00	91.00
	F1-score	54.00	90.50	75.00	75.00	89.50	93.00
ResNet50+CBAM	Accuracy	82.47	82.96	79.58	80.00	80.30	82.24
	Precision	77.00	80.00	95.00	59.00	92.00	97.00
	Recall	91.00	82.00	55.00	88.00	71.00	60.00
	F1-score	83.00	81.00	70.00	71.00	81.50	78.50
EfficientNet+ CBAM	Accuracy	78.34	82.96	67.00	79.06	80.99	84.40
	Precision	69.00	89.00	67.00	82.00	71.00	87.00
	Recall	52.00	94.00	86.00	69.00	100.00	94.00
	F1-score	60.00	91.50	75.00	75.00	83.00	90.50
Inceptionv3+ CBAM	Accuracy	60.00	83.00	42.79	69.22	84.91	90.00
	Precision	56.00	85.00	31.00	55.00	82.00	89.00
	Recall	65.00	90.00	50.00	70.00	79.00	86.00
	F1-score	60.00	87.50	39.00	62.50	80.00	87.00
Xception+ CBAM	Accuracy	66.79	81.30	56.00	62.00	71.47	80.00
	Precision	49.00	93.00	52.00	47.00	71.00	85.00
	Recall	61.00	88.00	46.00	60.00	71.00	79.00
	F1-score	55.00	90.00	49.00	52.00	71.00	82.00
Proposed Model	Accuracy	89.88	90.48	80.10	78.62	83.59	94.00
	Precision	82.00	84.00	91.00	64.00	85.00	86.00
	Recall	96.00	98.00	52.00	90.00	98.00	100.00
	F1-score	88.00	91.00	66.00	75.00	91.50	93.00

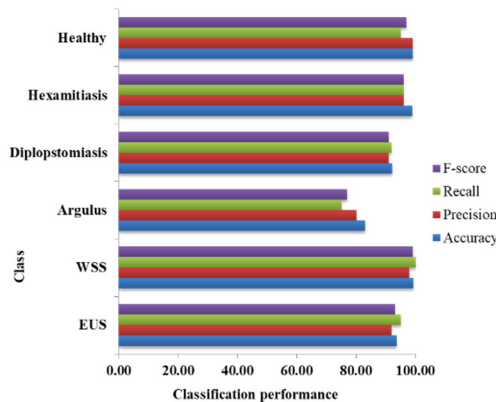


FIGURE 6. Class-wise performance of the proposed method with background elimination.

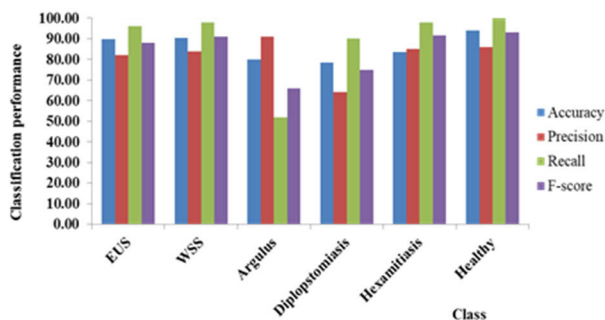


FIGURE 7. Class-wise performance of the proposed method without background removal.

The dataset is divided into training, validation and test sets in the ratio of 3:1:1, respectively. The training set uses 3,099 images, validation set uses 1,033 images and the test set uses

TABLE 10. Performance comparison of proposed method with ConvNeXt and Swin transformer models on dataset with background elimination.

Model	Acc. (%)	P (%)	R (%)	F (%)
Swin Transformer	95.55	91.50	92.50	92.00
ConvNeXt	95.95	91.17	89.00	90.10
Proposed Model	94.28	92.67	92.17	92.42

1,033 images as well. All image inputs to the neural networks are of size $224 \times 224 \times 3$. The same dataset is used for all the models in our experiments. The hardware and software specifications of the training environment include; Intel Core CPU i7-4790 of 3.6GHz, RAM of 16GB and supported by GPU (NVIDIA GeForce RTX 2080Ti), Tensorflow 2.4.0 and Keras 2.3.1 environment with the following parameters: batch size of 4, Adam optimizer, with default parameters $\beta 1 = 0.9$, $\beta 2 = 0.999$, $\alpha = 0.001$ and $\epsilon = 10^{-7}$ and cross entropy loss.

The proposed model is trained for 800 epochs on our dataset with eliminated background. The testing accuracy, precision, recall and F-measure are compared with state-of-the-art models used in image classification. We fine tune and train 5 deeper pretrained models for 300 epochs. The models are validated and tested on the same validation and test sets, respectively. The test performance comparisons are given in Table 5.

In comparison with the transfer learning models, the proposed model performs better with a top accuracy of 94.28% which is 2.2% higher than the accuracy of VGG19, the highest performing model in terms of accuracy among the pretrained models, precision of 92.67%, recall of 92.17% and F1-score of 92.42%. The proposed network is less deep

TABLE 11. Class-wise performance comparison of the proposed method with ConvNeXt and Swin transformer models on dataset with background elimination. (unit: %).

Model	Parameter	EUS	WSS	Argulus	Diplostomiasis	Hexamitiasis	Healthy
Swin Transformer	Accuracy	94.58	98.74	89.84	92.26	98.45	99.42
	Precision	92.00	98.00	80.00	89.00	91.00	99.00
	Recall	93.00	97.00	78.00	95.00	93.00	99.00
	F1-score	92.50	97.50	79.00	92.00	92.00	99.00
ConvNeXt	Accuracy	95.26	96.90	91.58	95.26	98.16	98.55
	Precision	91.00	94.00	87.00	86.00	90.00	99.00
	Recall	93.00	91.00	74.00	95.00	85.00	96.00
	F1-score	92.00	93.00	80.50	90.50	88.00	97.00
Proposed Model	Accuracy	90.09	98.30	81.70	91.00	95.30	97.30
	Accuracy	93.58	99.20	82.90	92.16	98.84	99.00
	Precision	92.00	98.00	80.00	91.00	96.00	99.00
	Recall	95.00	100.00	75.00	92.00	96.00	95.00
	F1-score	93.00	99.00	77.00	91.00	96.00	97.00

which reduces feature information loss and reduces the vanishing gradient problem and learns well on limited data. The attention modules further suppress regions of low interest and focus on more meaningful regions to enhance the feature extraction process. This intimates that more deeper and important features are easily extracted. The addition of the OSELM in the place of the fully connected layer further strengthens the classification process by reducing the training time and loss. This is achieved because the input nodes to hidden node weights and biases are generated randomly and the output weights are determined analytically on the basis of the training samples arriving sequentially. The proposed network has much less parameters compared to the pretrained models. To further ascertain the impact of our composite CNN-OSELM in comparison to other attention-based networks, the deep pretrained models have been enhanced by integrating attention mechanism after the last convolutional layer. The models are trained for 500 epochs, validated and tested on the same dataset. The performance comparison with the proposed method is shown in Table 6.

The proposed model still outperforms the pretrained-attention models by over 0.47% accuracy. The multilayer fusion network is less deep but wider, with two backbones that fuse features from the same level blocks. This refines the feature maps further and reduces loss that arises due to diminishing gradient problem commonly experienced in very deep networks. The OSELM further improves the classification performance and reduces the number of parameters. This is because only the number of hidden layers is selected and other parameters are automatically determined. To better appreciate the performance of the proposed method in comparison with the pretrained attention models, we present a detailed summary of the class-wise performance in Table 7.

The class-wise classification performance results presented in Table 8 show that our proposed model achieves the best results on overall for all the 6 classes, followed by attention-based ResNet50 and VGG19. However, in comparison with performance results of the baseline models depicted in Table 6, it shows that CBAM considerably improves the performance of the baseline models by an average of 1% on accuracy, 1.8% on precision, and 0.5% and 1.3% on recall and

TABLE 12. Performance comparison of proposed method with ConvNeXt and Swin transformer models on dataset without background elimination. (UNIT: %).

Model	Acc.	P	R	F
Swin Transformer	89.68	87.00	84.83	85.92
ConvNeXt	88.86	82.33	83.50	82.67
Proposed	86.11	82.00	89.00	84.00

F1-score, respectively. The class-wise performance in Table 7 reveals that the classification performance of whitespot disease was highest with an accuracy of 99.20% and F1-score of 99% followed by the healthy class with an accuracy of 99% and F1-score of 97%. This pattern is true in both the proposed method and the attention-based pretrained models. This is attributed to the physical characteristics of these two classes which makes them very distinct and easily identified. In Table 6, attention-based ReNet50 generally out competes its counterparts due to its ability to maintain a low error rate with skip connections that help to regularize layers in case of performance degradation. However, the classification performance of Argulus was poorest with the proposed model achieving an accuracy of 82.90% and F1-score of 77%. Moreover, all the attention-based pretrained models also indicate a similar performance on the Argulus class despite hexamitiasis class having the least number of training samples. The poor classification performance of Argulus is associated with the nature of the disease. The Argulus disease causing agent, Argulidae, tends to attach itself on the body of its host. However, the host will bear physical characteristics of the infestation after the disease has adversely progressed. The proposed approach bases the recognition ability on the presence of the Argulidae on the fish body which tends to resemble the color of its host making it difficult to identify. The class-wise performance of the proposed method with background elimination is graphically illustrated in Fig. 6.

The image background are initially subtracted in the RGB color space from the training samples to reduce noise brought about by the unstable water environment due to multiple underwater scenes, complex underwater seabed, reflections, low contrast, low-resolution and blurring of images. To illustrate the relevancy of image background elimination in our

TABLE 13. Class-wise performance comparison of the proposed method with ConvNeXt and Swin transformer models on dataset with background elimination. (unit: %).

Model	Parameter	EUS	WSS	Argulus	Diploptomiasis	Hexamitiasis	Healthy
Swin Transformer	Accuracy	92.26	94.90	81.58	83.95	88.16	97.20
	Precision	78.00	93.00	84.00	76.00	92.00	99.00
	Recall	93.00	91.00	63.00	86.00	85.00	91.00
	F1-score	85.50	92.00	73.50	81.00	88.50	95.00
ConvNeXt	Accuracy	91.87	85.09	77.44	92.74	94.10	91.97
	Precision	84.00	88.00	74.00	66.00	83.00	99.00
	Recall	79.00	93.00	74.00	89.00	79.00	87.00
	F1-score	81.50	90.50	74.00	76.00	81.00	93.00
Proposed Model	Accuracy	78.34	82.96	67.00	79.06	80.99	84.40
	Accuracy	89.88	90.48	80.10	78.62	83.59	94.00
	Precision	82.00	84.00	91.00	64.00	85.00	86.00
	Recall	96.00	98.00	52.00	90.00	98.00	100.00
	F1-score	88.00	91.00	66.00	75.00	91.50	93.00

study, we further trained, validated and tested the proposed model and the attention-based pretrained models on the original dataset without background elimination. In other words, the models were trained, tested and validated on the same number of data samples with their original background as opposed to the previous experiments. The summarized results are shown in Table 8 and the class-wise performance results are shown in Table 9. The results attained by all the models show a poor uniform pattern of accuracy, precision, recall and F1-score though the proposed method still achieves better results compared to the attention based pretrained models.

The performance of the proposed method without background removal is demonstrated graphically in Fig. 7 for a clearer overview. The class-wise performance without background elimination generally reveals an over-fitting problem and is generally poor with some classes having very low values of F1-score and accuracy and others having fairly higher accuracy values but with low precision and recall.

We further compare the performance of our proposed model on two competing models, ConvNeXt [56] and Swin transformer [57]. ConvNeXt uses depth-wise convolution, which is a special case of grouped convolution in which the number of groups and channels are equal. The depth-wise convolution is closely similar to weighted sum operation used in self-attention. This operates on a per-channel basis; the information is mixed only in the spatial dimension. On the other hand, Swin transformer is a hierarchical transformer that uses shifted windows to compute its representation.

The Swin transformer was proposed by [57] to overcome the challenges that arise in adapting transformer models from natural language processing (NLP) to vision tasks. These challenges are associated with the differences that occur between NLP and vision tasks such as high pixel resolution in comparison with words and the scale of visual entities. The Swin transformer and the ConvNeXt are trained, validated and tested on the same dataset (with background elimination) as the proposed model. The overall test performance of these models in comparison with the proposed model is shown in Table 10.

The proposed model slightly outperforms ConvNeXt and Swin transformer with F1-score higher than both models.

However, ConvNeXt achieves the highest classification accuracy of 95.95%. Additionally, the class-wise performance shown in Table 11 shows a more stable classification performance of the proposed model compared to the ConvNeXt and Swin transformer. The variations between accuracy, precision and recall are smaller in the proposed model compared to the ConvNeXt and Swin transformer. ConvNeXt achieves the highest overall recognition performance of Argulus class with a classification accuracy of 91.58% and harmonic mean of 80.50% which is the highest classification performance for Argulus. However, its recall value is very low, 74.00% which indicates a high false negative ratio. This is not appropriate for fish disease identification because misclassifying positive cases as negative can lead to widespread of the disease in the aquarium before it is detected.

Our proposed model has the ability to detect both positive and negative cases at a similar rate, as indicated by the class-wise performance in Table 11. We further evaluated the performance of ConvNeXt and Swin transformer in comparison with the proposed method on the dataset without background elimination. The overall results are shown in Table 12.

Swin transformer performed slightly better than our model on noisy data indicating its resiliency to noise. The class-wise performance of ConvNeXt and Swin transformer on noisy data in comparison with the proposed model is shown in Table 13. The results show a generally deteriorating performance pattern on the dataset without background elimination across the three models. The ConvNeXt and Swin transformer models are more unstable considering the variations in precision and recall for the respective classes. Though Swin transformer tends to achieve a higher accuracy value on this dataset, the ability to make positive and negative predictions correctly is averagely lower compared to the proposed model.

VI. CONCLUSION AND FUTURE WORK

This study proposed a multilayer fusion attention CNN-OSELM network for fish disease recognition in aquaculture. We build a more powerful feature extractor that integrates the output features from same level blocks to enhance feature extraction using SLC and multilayer fusion with CBAM.

We incorporate OSELM at the fully connected layer to reduce the training time and improve classification performance. The dataset used in the study was collected from various internet sources and the image backgrounds were initially subtracted in the RGB color space from the training samples to reduce noise brought about by interferences from multiple underwater scenes, complex underwater seabed background, low contrast, reflections, low-resolution and blurring of images. The network was trained, tested and validated on both the preprocessed dataset and the original dataset to appreciate the impact of background elimination from underwater fish disease images on classification performance. The pretrained backbone networks with and without attention mechanism, Swin transformer and ConvNeXt were also trained, validated and tested on the same dataset for performance comparisons. The trained models were evaluated on the basis of accuracy, precision, recall and F1-score. The proposed method achieves superior test results on the dataset with background elimination compared to the rest of the models. Though Swin transformer performs slightly better than our method, the ability to make positive and negative predictions correctly is averagely lower compared to the proposed model.

The class-wise performance also indicated that whitespot disease and healthy had higher classification performance than the rest of the classes due to the distinctive and more visible nature of these infections. ConvNeXt also achieved higher F1-score on Argulus class though with a lower recall value which implies higher false negative predictions. Such a scenario is not good in the case of fish disease identification. This implies the model could classify positive cases as negative which is very dangerous during disease outbreak. Instead, a higher false positive rate, that is a lower precision value could be tolerated.

In future work, more training samples will be collected and used for further training in order to improve the generalization ability and overall performance of our model. Further, fish disease recognition performance of deep learning models will be compared with fuzzy logic applied on segmented images. The experimental results of this paper show a promising step towards automatic fish disease identification in smart aquaculture.

REFERENCES

- [1] D. Qu, "The state of world fisheries and aquaculture report," Food Agricult. Org. (FAO), United Nations, Rome, Italy, Tech. Rep., Mar. 2022, pp. 1–73.
- [2] D. Persson, A. Nødtvedt, A. Aunsmo, and M. Stormoen, "Analysing mortality patterns in salmon farming using daily cage registrations," *J. Fish Diseases*, vol. 45, no. 2, pp. 335–347, Feb. 2022.
- [3] M. Tavares-Dias and M. L. Martins, "An overall estimation of losses caused by diseases in the Brazilian fish farms," *J. Parasitic Diseases*, vol. 41, no. 4, pp. 913–918, Dec. 2017.
- [4] E. J. Noga, *Fish Disease: Diagnosis and Treatment*, 2nd ed. Hoboken, NJ, USA: Wiley, Jun. 2010, pp. 1–544.
- [5] G. Osman, M. M. M. Alam, S. M. I. Khalil, S. M. Bari, A. Hamom, M. Parven, and M. A. A. Mamun, "Identification of pathogenic bacteria from diseased stringing catfish *Heteropneustes fossilis* with their sensitivity to antibiotics," *Int. J. Fisheries Aquatic Stud.*, vol. 8, no. 1, pp. 291–301, Dec. 2019.
- [6] N. Steckler and R. P. E. Yanong, "Argulus (fish louse) infections in fish," Univ. Florida, Gainesville, FL, USA, Tech. Rep. FA184, Jan. 2022, pp. 1–4, vol. 184, no. 1.
- [7] R. Wang, J. Feng, Y. Su, L. Ye, and J. Wang, "Studies on the isolation of photobacterium damsela subsp. piscicida from diseased golden pompano (*Trachinotus ovatus* Linnaeus) and antibacterial agents sensitivity," *Veterinary Microbiol.*, vol. 162, nos. 2–4, pp. 957–963, Mar. 2013.
- [8] H. A. Alturkistani, F. M. Tashkandi, and Z. M. Mohammedsleh, "Histological stains: A literature review and case study," *Global J. Health Sci.*, vol. 8, no. 3, pp. 72–79, May 2016.
- [9] A. E. Toranzo, B. Magariños, and J. L. Romalde, "A review of the main bacterial fish diseases in mariculture systems," *Aquaculture*, vol. 246, nos. 1–4, pp. 37–61, May 2005.
- [10] M. Sun, X. Yang, and Y. Xie, "Deep learning in aquaculture: A review," *J. Comput.*, vol. 31, no. 1, pp. 294–319, Jan. 2020.
- [11] J. Wang, S. Lee, Y. Lai, C. Lin, T. Wang, Y. Lin, T. Hsu, C. Huang, and C. Chiang, "Anomalous behaviors detection for underwater fish using AI techniques," *IEEE Access*, vol. 8, pp. 224372–224382, 2020.
- [12] K. P. R. A. Haq and V. P. Harigovindan, "Water quality prediction for smart aquaculture using hybrid deep learning models," *IEEE Access*, vol. 10, pp. 60078–60098, 2022.
- [13] S. P. Khabusi and Y. Huang, "A deep learning approach to predict dissolved oxygen in aquaculture," in *Proc. Int. Conf. Adv. Robot. Intell. Syst. (ARIS)*, Taipei, Taiwan, Aug. 2022, pp. 1–6.
- [14] L. Kuang, P. Shi, C. Hua, B. Chen, and H. Zhu, "An enhanced extreme learning machine for dissolved oxygen prediction in wireless sensor networks," *IEEE Access*, vol. 8, pp. 198730–198739, 2020.
- [15] D. Li, J. Sun, H. Yang, and X. Wang, "An enhanced naive Bayes model for dissolved oxygen forecasting in shellfish aquaculture," *IEEE Access*, vol. 8, pp. 217917–217927, 2020.
- [16] W. Hu, L. Chen, B. Huang, and H. Lin, "A computer vision-based intelligent fish feeding system using deep learning techniques for aquaculture," *IEEE Sensors J.*, vol. 22, no. 7, pp. 7185–7194, Apr. 2022.
- [17] H. Yang, X. Wang, J. Sun, and D. Li, "Dissolved oxygen prediction using RBF network based on improved conjugate gradient method," in *Proc. IEEE 11th Int. Conf. Softw. Eng. Service Sci. (ICSESS)*, Beijing, China, Oct. 2020, pp. 515–518.
- [18] W. Lin, J. Zhong, S. Liu, T. Li, and G. Li, "ROIMIX: Proposal-fusion among multiple images for underwater object detection," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Barcelona, Spain, May 2020, pp. 2588–2592.
- [19] Z. Zhao, Y. Liu, X. Sun, J. Liu, X. Yang, and C. Zhou, "Composited Fish-Net: Fish detection and species recognition from low-quality underwater videos," *IEEE Trans. Image Process.*, vol. 30, pp. 4719–4734, 2021.
- [20] C. R. Jung, "Efficient background subtraction and shadow removal for monochromatic video sequences," *IEEE Trans. Multimedia*, vol. 11, no. 3, pp. 571–577, Apr. 2009.
- [21] Y. Liu, Y. Wang, S. Wang, T. Liang, Q. Zhao, Z. Tang, and H. Ling, "CBNet: A novel composite backbone network architecture for object detection," in *Proc. AAAI Conf. Artif. Intell.*, Apr. 2020, vol. 34, no. 7, pp. 11653–11660.
- [22] T. Liang, X. Chu, Y. Liu, Y. Wang, Z. Tang, W. Chu, J. Chen, and H. Ling, "CBNet: A composite backbone network architecture for object detection," *IEEE Trans. Image Process.*, vol. 31, pp. 6893–6906, 2022.
- [23] N.-Y. Liang, G.-B. Huang, P. Saratchandran, and N. Sundararajan, "A fast and accurate online sequential learning algorithm for feedforward networks," *IEEE Trans. Neural Netw.*, vol. 17, no. 6, pp. 1411–1423, Nov. 2006.
- [24] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: Theory and applications," *Neurocomputing*, vol. 70, nos. 1–3, pp. 489–501, Dec. 2006.
- [25] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, and X. Tang, "Residual attention network for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 6450–6458.
- [26] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 7132–7141.
- [27] M. S. Ahmed, T. T. Aurpa, and M. A. K. Azad, "Fish disease detection using image based machine learning technique in aquaculture," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 34, no. 8, pp. 5170–5182, Sep. 2022.

- [28] S. Malik, T. Kumar, and A. K. Sahoo, "Image processing techniques for identification of fish disease," in *Proc. IEEE 2nd Int. Conf. Signal Image Process. (ICSIP)*, Singapore, Aug. 2017, pp. 55–59.
- [29] V. Lyubchenko, R. Matarneh, O. Kobylin, and V. Lyashenko, "Digital image processing techniques for detection and diagnosis of fish diseases," *Int. J. Adv. Res. Comput. Sci. Softw. Eng.*, vol. 6, no. 7, pp. 79–83, Jul. 2016.
- [30] Y. Wang, H. Ye, and B. Li, "A research based on recognition algorithm of characteristics of body surface of infected fish," in *Proc. World Automat. Congr.*, Kobe, Japan, Sep. 2010, pp. 155–160.
- [31] J. Liu, H. Yu, W. Yang, and C. Sun, "Combining active learning and semi-supervised learning based on extreme learning machine for multi-class image classification," in *Proc. Int. Conf. Intell. Sci. Big Data Eng.*, Suzhou, China, Jun. 2015, pp. 163–175.
- [32] Y. Zeng, X. Xu, Y. Fang, and K. Zhao, "Traffic sign recognition using deep convolutional networks and extreme learning machine," in *Proc. Int. Conf. Intell. Sci. Big Data Eng.*, Suzhou, China, Jun. 2015, pp. 272–280.
- [33] R. Dwivedi, T. Dutta, and Y.-C. Hu, "A leaf disease detection mechanism based on L1-norm minimization extreme learning machine," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, no. 1, Aug. 2022, Art. no. 8019905.
- [34] M. Nahiduzzaman, M. O. F. Goni, M. S. Anower, M. R. Islam, M. Ahsan, J. Haider, S. Gurusamy, R. Hassan, and M. R. Islam, "A novel method for multivariant pneumonia classification based on hybrid CNN-PCA based feature extraction using extreme learning machine with CXR images," *IEEE Access*, vol. 9, pp. 147512–147526, 2021.
- [35] X. Zhang and L. Qin, "An improved extreme learning machine for imbalanced data classification," *IEEE Access*, vol. 10, pp. 8634–8642, 2022.
- [36] Q. Lv, X. Niu, Y. Dou, J. Xu, and Y. Lei, "Classification of hyperspectral remote sensing image using hierarchical local-receptive-field-based extreme learning machine," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 3, pp. 434–438, Mar. 2016.
- [37] Y. Bazi, N. Alajlan, F. Melgani, H. AlHichri, S. Malek, and R. R. Yager, "Differential evolution extreme learning machine for the classification of hyperspectral images," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 6, pp. 1066–1070, Jun. 2014.
- [38] G. Huang, H. Zhou, X. Ding, and R. Zhang, "Extreme learning machine for regression and multiclass classification," *IEEE Trans. Syst. Man, Cybern. B, Cybern.*, vol. 42, no. 2, pp. 513–529, Apr. 2012.
- [39] Y. Zhou, J. Peng, and C. L. P. Chen, "Extreme learning machine with composite kernels for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 6, pp. 2351–2360, Jun. 2015.
- [40] Y. Yang and Q. M. J. Wu, "Extreme learning machine with subnetwork hidden nodes for regression and classification," *IEEE Trans. Cybern.*, vol. 46, no. 12, pp. 2885–2898, Dec. 2016.
- [41] X. Liu, Q. Hu, Y. Cai, and Z. Cai, "Extreme learning machine-based ensemble transfer learning for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 3892–3902, Jun. 2020.
- [42] H. Li, H. Zhao, and H. Li, "Neural-response-based extreme learning machine for image classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 2, pp. 539–552, Feb. 2019.
- [43] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Aug. 2017, pp. 2117–2125.
- [44] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," Cornell Univ., Ithaca, NY, USA, Tech. Rep. 1807.06521, Jul. 2018, pp. 1–17.
- [45] P. K. Wong, C. M. Vong, X. H. Gao, and K. I. Wong, "Adaptive control using fully online sequential-extreme learning machine and a case study on engine air-fuel ratio regulation," *Math. Problems Eng.*, vol. 1, no. 1, pp. 1–11, Apr. 2014.
- [46] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, Lake Tahoe, NV, USA, Dec. 2012, pp. 1097–1105.
- [47] N. Wang, M. J. Er, and M. Han, "Generalized single-hidden layer feedforward networks for regression problems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 6, pp. 1161–1176, Jun. 2015.
- [48] S. Tamura and M. Tateishi, "Capabilities of a four-layered feedforward neural network: Four layers versus three," *IEEE Trans. Neural Netw.*, vol. 8, no. 2, pp. 251–255, Mar. 1997.
- [49] G.-B. Huang, "Learning capability and storage capacity of two-hidden-layer feedforward networks," *IEEE Trans. Neural Netw.*, vol. 14, no. 2, pp. 274–281, Mar. 2003.
- [50] C. R. Rao and S. K. Mitra, "Generalized inverse of a matrix and its applications," in *Proc. 6th Berkeley Symp. Math. Statist. Probab.*, Berkeley, CA, USA, Jul. 1971, pp. 601–620.
- [51] G.-B. Huang, L. Chen, and C.-K. Siew, "Universal approximation using incremental constructive feedforward networks with random hidden nodes," *IEEE Trans. Neural Netw.*, vol. 17, no. 4, pp. 879–892, Jul. 2006.
- [52] G. H. Golub and C. F. V. Loan, *Matrix Computations*, 3rd ed. Baltimore, MD, USA: The Johns Hopkins Univ. Press, 1996, pp. 1–723.
- [53] G.-B. Huang, N.-Y. Liang, H.-J. Rong, P. Saratchandran, and N. Sundararajan, "On-line sequential extreme learning machine," in *Proc. Int. Conf. Comput. Intell.*, Calgary, AB, Canada, Jul. 2005, pp. 1–6.
- [54] J. Sharma, O.-C. Granmo, and M. Goodwin, "Deep CNN-ELM hybrid models for fire detection in images," in *Proc. Artif. Neural Netw. Mach. Learn. (ICANN)*, Rhodes, Greece, Sep. 2018, pp. 245–259.
- [55] J. Ye, T. Gao, and J. Zhang, "Moving object detection with background subtraction and shadow removal," in *Proc. 9th Int. Conf. Fuzzy Syst. Knowl. Discovery (FSKD)*, Chongqing, China, May 2012, pp. 1859–1863.
- [56] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A ConvNet for the 2020s," Cornell Univ., Ithaca, NY, USA, Tech. Rep. 2201.03545, Mar. 2022, pp. 1–15.
- [57] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 9992–10002.



YO-PING HUANG (Fellow, IEEE) received the Ph.D. degree in electrical engineering from Texas Tech University, Lubbock, TX, USA. He was a Professor and the Dean of Research and Development, the Dean of the College of Electrical Engineering and Computer Science, and the Department Chair with Tatung University, Taipei, Taiwan. He is currently the President of the National Penghu University of Science and Technology, Penghu, Taiwan. He is also a Chair

Professor with the Department of Electrical Engineering, National Taipei University of Technology, Taipei, where he was the Secretary-General. His current research interests include fuzzy system design and modeling, deep learning modeling, intelligent control, medical data mining, and rehabilitation systems design. He is a fellow of IET, CACS, TFSA, and the International Association of Grey System and Uncertain Analysis. He received the 2021 Outstanding Research Award from the Ministry of Science and Technology, Taiwan. He serves as the IEEE SMCS VP for Conferences and Meetings and the Chair for the IEEE SMCS Technical Committee on Intelligent Transportation Systems. He was the IEEE SMCS BoG, the President of the Taiwan Association of Systems Science and Engineering, the Chair of the IEEE SMCS Taipei Chapter, the Chair of the IEEE CIS Taipei Chapter, and the CEO of the Joint Commission of Technological and Vocational College Admission Committee, Taiwan.



SIMON PETER KHABUSI (Graduate Student Member, IEEE) received the B.S. degree in computer engineering from Busitema University, Tororo, Uganda, in 2016, and the M.Tech. degree in computer science and engineering from Delhi Technological University, Delhi, India, in 2020. He is currently pursuing the Ph.D. degree in electrical engineering with the National Taipei University of Technology, Taipei, Taiwan. Before joining the Ph.D. program, he was a Systems Administrator with the Uganda Heart Institute, Mulago National Referral Hospital, Kampala, Uganda, and a part-time Assistant Lecturer with the Ernest Cook Ultrasound Research and Education Institute, Mengo Hospital, Kampala. His research interests include machine learning, deep learning, and computer vision and their applications in aquaculture, network and information security, medical image analysis, and health informatics.