fishes

MDPI

Supplementary Materials

Intelligent Diagnosis of Fish Behavior Using Deep Learning Method

Usama Iqbal [1,2,3,4,5], Daoliang Li [1,2,3,4,5,*] and Muhammad Akhter [1,2,3,4,5]

[1] College of Information and Electrical Engineering, China Agricultural University, Beijing 100083, China
[2] National Innovation Center for Digital Fishery, China Agricultural University, Beijing 100083, China
[3] Key Laboratory of Smart Farming Technologies for Aquatic Animal and Livestock, Ministry of Agriculture and Rural Affairs,, Beijing 100083, China
[4] Beijing Engineering and Technology Research Center for Internet of Things in Agriculture, , Beijing 100083, China
[5] Yantai Institute of China Agricultural University, Beijing 100083, China
* Correspondence: dliangl@cau.edu.cn

1. Convolutional Neural Network Architecture Description

1.1. CNN Model-1

The Convolutional Neural Network (CNN) filters can extract and classify core features. Implementing the regularization technique and adjustment of weights is essential for better training outcomes. The proposed CNN model-1 architecture is shown in figure 3(a). The input image given to CNN has a resolution of 224 × 224 × 3. This input image passes through six convolutional layers, six Relu functions, and two pooling layers. Before passing through two Fully Connected (FC) Layers, a feature map of 56 × 56 × 512 is obtained. The first three layers of CNN contain a sequence of 64, 128, and 256 filters. However, 3 × 3 kernel size and 1 × 1 stride size are used for all three layers. The output feature map size can be calculated using the following formula.

$$q_h, q_w, q_r = \left(\left\lfloor \frac{n_h+2e-k}{s} + 1 \right\rfloor, \left\lfloor \frac{n_w+2e-k}{s} + 1 \right\rfloor, n_k\right) \ldots\ldots (1)$$

n_h and n_w represent the height and width of the input image size, whereas k represents the kernel size, whereas $e, s \; and \; n_k$ Indicate the padding size, stride size, and the number of filters used, respectively. The output height, width, and channel number after each convolutional layer are represented by $q_h, q_w \; and \; q_r$ Respectively. The pool-1 layer performed the max-pooling function (kernel size 2 × 2, and stride 2 × 2), reducing the feature map size. The following formula can be used to calculate the output size of the image after each pooling layer

$$d_h, d_w, d_r = \left(\left\lfloor \frac{n_h+2e-k}{s} + 1 \right\rfloor, \left\lfloor \frac{n_w+2e-k}{s} + 1 \right\rfloor, n_c\right) \ldots\ldots (2)$$

Where n_c is the number of channels of the input. $d_h, d_w and \; d_r$ are the output height, width and number of channel after pooling layers respectively. The rest of the three layers consisted of 512 filters in each layer with the configuration of 3 × 3 kernel size and stride size 1 × 1, respectively. Sigmoid, Relu, and tanh are commonly used activation functions in CNN. The performance of the Relu is better than the other two non-linear activation functions and helps eliminate the propagation problem while in the training phase. Therefore, Relu is used for fast training and overcoming other issues in the proposed study. The SGD optimizer was used to optimize the model, with a learning rate, drop rate, and batch size of 0.00001, 0.3, and 1. During training, the dropout method is crucial in preventing the adaption of the same neurons having the same features repeatedly and is also helpful in forcing the network to learn robust features with different neurons. Consequently, it can enhance testing accuracy and reduce the loss of network capacity

and other computational resources. In the end, two FC layers with 4096 neurons connected with the final layer are employed to convert the output of the convolutional layer into two-dimensional, classify the images, and predict the category.

Later model-1 is implemented without using the max-pooling operation in the second part of the experiment, using the same architecture and configuration as before. The detailed structure of filter size, Rectified Linear Unit (Relu), stride, and pooling are explained in Tables 1 and 2.

Table 1. Configuration of CNN model-1 of the proposed method with the max-pooling operation.

Layer Config	Filters	Class Map Size	Kernel Size	Num of Stride
Input Image size		224 × 224 × 3		
Conv-1	64	224 × 224 × 64	3 × 3	1
Relu-1		224 × 224 × 64		
Conv-2	128	224 × 224 × 128	3 × 3	1
Relu-2		224 × 224 × 128		
Conv-3	256	224 × 224 × 256	3 × 3	1
Relu-3		224 × 224 × 256		
Pool-1		112 × 112 × 256		2
Conv-4	512	112 × 112 × 512	3 × 3	1
Relu-4		112 × 112 × 512		
Conv- 5	512	112 × 112 × 512	3 × 3	1
Relu - 5		112 × 112 × 512		
Conv - 6	512	112 × 112 × 512	3 × 3	1
Relu - 6		112 × 112 × 512		
Pool-2		56 × 56× 512		2
FC layer - I		4096 × 1		
FC layer - II		1024 × 1		
Response Layer		2 × 1		

Table 2. Configuration of CNN model-1 of the proposed method without the max-pooling operation.

Layer Config	Filters	Class Map Size	Kernel Size	Num of Stride
Input Image size		224 × 224 × 3		
Conv-1	64	224 × 224 × 64	3 × 3	1
Relu-1		224 × 224 × 64		
Conv-2	128	224 × 224 × 128	3 × 3	1
Relu-2		224 × 224 × 128		
Conv-3	256	224 × 224 × 256	3 × 3	1
Relu-3		224 × 224 × 255		
Conv-4	512	224 × 224 × 512	3 × 3	1
Relu-4		224 × 224 × 512		
Conv - 5	512	224 × 224 × 512	3 × 3	1
Relu - 5		224 × 224 × 512		
Conv - 6	512	224 × 224 × 512	3 × 3	1
Relu - 6		224 × 224 × 512		
FC layer - I		4096 × 1		
FC layer - II		1024 × 1		
Response Layer		2 × 1		

1.2. CNN Model-2

In the second phase of the experiment, we devised and implemented the Convolutional Neural Network (CNN) model-2 approach to improve classification performance. Model-2 has also consisted of six convolutional layers, six Relu functions, two pooling layers, and three Fully Connected (FC) layers. The strategy to add three FC layers to feed the diagnosis classifier fulfilled our need for classification. Model-2 is also implemented with and without using the max-pooling function. The optimizer, learning rate, drop rate, and batch size variables from Model-1 were kept same in this part of the experiment. The proposed model architecture is depicted in Figure 3 (b). Tables 3 and 4 discuss the filter size, stride, and pooling settings in detail.

Table 3. Configuration of CNN model-1 of the proposed method with the max-pooling operation.

Layer Config	Filters	Class Map Size	Kernel Size	Num of Stride
Input Image size		224 × 224 × 3		
Conv-1	64	224 × 224 × 64	3 × 3	1
Relu-1		224 × 224 × 64		
Conv-2	128	224 × 224 × 128	3 × 3	1
Relu-2		224 × 224 × 128		
Conv-3	256	224 × 224 × 256	3 × 3	1
Relu-3		224 × 224 × 256		
Pool-1		112 × 112 × 256		2
Conv-4	512	112 × 112 × 512	3 × 3	1
Relu-4		112 × 112 × 512		
Conv- 5	512	112 × 112 × 512	3 × 3	1
Relu - 5		112 × 112 × 512		
Conv - 6	512	112 × 112 × 512	3 × 3	1
Relu - 6		112 × 112 × 512		
Pool-2		56 × 56× 512		2
FC layer -I		4096 × 1		
FC layer -II		4096 × 1		
FC layer -III		1024 × 1		
Response Layer		2 × 1		

Table 4. Configuration of CNN model-1 of the proposed method without the max-pooling operation.

Layer Config	Filters	Class Map Size	Kernel Size	Num of Stride
Input Image size		224 × 224 × 3		
Conv-1	64	224 × 224 × 64	3 × 3	1
Relu-1		224 × 224 × 64		
Conv-2	128	224 × 224 × 128	3 × 3	1
Relu-2		224 × 224 × 128		
Conv-3	256	224 × 224 × 256	3 × 3	1
Relu-3		224 × 224 × 255		
Conv-4	512	224 × 224 × 512	3 × 3	1
Relu-4		224 × 224 × 512		
Conv - 5	512	224 × 224 × 512	3 × 3	1
Relu - 5		224 × 224 × 512		
Conv - 6	512	224 × 224 × 512	3 × 3	1
Relu - 6		224 × 224 × 512		
FC layer -I		4096 × 1		

FC layer -II	4096 × 1
FC layer -III	1024 × 1
Response Layer	2 × 1