Computing and Informatics, Vol. 41, 2022, 714–738, doi: 10.31577/cai_2022_3_714

SEABED SEDIMENT CLASSIFICATION FOR SONAR IMAGES BASED ON DEEP LEARNING

Yuhong Shi, Mo Chen

School of Information Science and Engineering
Dalian Polytechnic University
Dalian, China
e-mail: wcwcwwwc621@163.com, 995176560@qq.com

Chunlong Yao, Xu Li, Lan Shen

Dalian Polytechnic University
Dalian, China
e-mail: yaocl@dlpu.edu.cn

Abstract. Along with the development of sonar technology, the detection accuracy and stability of sonar have been improved. A large amount of seabed sediment information can be obtained through sonar detection. However, this information is often accompanied by noise interference, resulting in poor quality of the generated images. Moreover, sonar images are different from conventional images. There are single-channel images. The model needs to classify the images according to the texture features in the images. Coupled with the scarcity of sonar data, this makes it difficult to accurately classify the seabed sediment. According to the characteristics of sonar images, we propose the ShuffleNet-DSE which is a classification model based on deep learning. The ShuffleNet-DSE network is based on ShuffleNet-V2, while ensuring the lightweight of the network, it incorporates feature dense connection and Squeeze-and-Excitation (SE) structure channel self-attention. And combined with the sonar image's characteristics, the partial activation function of the model is changed to the Swish. The experimental results show that compared with the traditional machine learning classification method, ShuffleNet-DSE has greatly improved the classification accuracy and the computational cost. Compared with excellent neural network models such as AlexNet, MobileNet-V3, GoogLeNet and ResNet, it is more suitable for sonar image processing.

Keywords: Image classification, sediment classification, sonar image, deep learning, ShuffleNet

1 INTRODUCTION

The sea area is vast, and the sea area contains extremely rich resources. Mankind's cognition of the sea is still in the exploratory stage. The seabed sediment mainly refers to the constituent materials on the surface of the seabed, and the common ones are silt, sand and rock [1]. With the development and application of sonar technology, humans can detect seabed sediment through sonar, thereby obtaining seabed sediment images and inverting the seabed topography based on the images [2]. For people's life, seabed topography not only directly affects the navigation and anchoring of ships [3], but also affects the success of submarine pipeline laying. For the utilization of seabed resources, the distribution of seabed minerals can be obtained through the analysis of seabed sediment, so as to better develop and use seabed resources [4]. For the military, the visualization of seabed environment is particularly important [5], they need to adopt different strategies to solve the problem according to different seabed environments. Therefore, the study of more efficient seabed sediment classification methods can provide an information basis for further exploration of the ocean.

Sonar is currently the most appropriate means to explore the seabed on a large scale. Compared with optical means, sonar has better penetration and wider detection range [6]. However, the detection effect of sonar is often affected by external factors, most of which are related to the complex marine environment [7]. As a result, the generated sonar images often have more noise interference. Moreover, the sonar data produced by sonars of different performances are quite different [8]. The sonar image is a picture generated by mapping the intensity of the echo to the gray value. It is a single-channel picture, and the image features are mainly texture features [9]. Due to the confidentiality and diversity of storage methods of sonar data, there is a lack of clear and standard sonar image data on the seabed sediment [10]. All of these have brought great difficulty to the subsequent classification of sediment.

With the development of deep learning technology, new technical support is provided for the classification of seabed sediment. At present, for normal RGB image classification, large-scale CNNs models have been able to perform very well. However, these models cannot effectively adapt to the characteristics of the sonar image. The processing of sonar images is accompanied by severe over-fitting, and the texture features of the images cannot be extracted effectively, resulting in low prediction accuracy. Their complex network structure limits the classification performance. This paper starts from three aspects: feature extraction, inter-channel information processing and activation function selection, and optimizes the deep learning network. Among them, the ShuffleNet-V2's Depthwise-Separable-Convolution (DWConv) and

Channel-Shuffle are more suitable for processing the inter-channel information of sonar images through experimental comparison. Using ShuffleNet-V2 as the core, and integrates the Squeeze-and-Excitation (SE) module into it. The SE module filters according to the channel weight and selects more effective channel information as the model input, reducing the complexity of the model as much as possible. The SE module can reduce noise interference and make the overall model more stable. Use Swish instead of ReLU activation function to make the model improved by more comprehensive learning features. And because the depth of the model should not be too large, the feature dense connection is adopted in the feature extraction part which can transfer the features more effectively without increasing the depth of the model. After comprehensive consideration, the improved model has greatly improved operation efficiency and classification accuracy.

The rest of the paper is organized as follows. Section 2 presents the literature review. Section 3 depicts the design of the method in detail. Section 4 presents the experimental part, which evaluates the effectiveness of the proposed method by comparing the performance of different classification models. Finally, Section 5 concludes this work and discusses the future direction.

2 RELATED WORK

Research on the classification of sonar images can be divided into two categories: traditional machine learning and deep learning. Therefore, articles related to these two methods are described separately.

2.1 Traditional Machine Learning

The traditional research idea is to divide the establishment of the relationship model between sonar images and substrate types into two stages, namely sonar image feature extraction and classifier training. Traditional machine learning research is divided into unsupervised learning and supervised learning.

At the initial stage, due to the lack of sonar data, some scholars tend to use unsupervised learning methods for data classification in order to solve the situation of small data samples. Clustering is a typical algorithm for unsupervised learning. The k-means clustering algorithm is a commonly used clustering algorithm. Lu et al. [11] proposed the use of k-means clustering algorithm in submarine geological classification. However, the clustering method is affected by the initial value. If the initial value is not appropriate, the classification accuracy will be poor. So, some scholars turned their attention to the Self-Organizing Maps (SOM). The SOM was proposed by Kohonen [12] in 1981. The SOM network is more suitable for sparse data than other methods. Tang et al. [13] use the combination of Gray Level Co-Occurrence Matrix (GLCM) and SOM to realize the classification of various types of seabed sediment. This framework of using algorithms to extract features and then classify them has been widely used since then.

However, since unsupervised learning cannot obtain specific types of substrates, in order to obtain clear types of substrates, supervised learning is usually used to achieve classification.

Xiong et al. [14] performed principal component analysis by extracting feature vectors, selecting standard deviation, contrast and other parameters as training feature vectors, and using support vector machine (SVM) for classification, which verifies the feasibility of using supervised learning methods to achieve classification. On this basis, Xu et al. [15] proposed a support vector machine based on a radial basis kernel function, which effectively improves the classification accuracy compared to traditional SVM. The SVM method often means a lot of calculations, so it is gradually replaced by the BP network. Yang et al. [16] combined genetic algorithm and BP network to achieve better accuracy for multi-substrate type recognition scenarios, this study provides a knowledge base for multi-type classification of seabed sediments, and also proves that BP network has research potential in the field of sonar data processing. Xiong et al. [17] combined the characteristics of genetic algorithm, wavelet analysis and neural network, they use genetic algorithm to optimize the initial weights and wavelet parameters of wavelet neural network, and combine the multi-resolution and local refinement of wavelet analysis. When compared with the previous, the BP neural network model has achieved better accuracy in the recognition of the three types of substrates.

Although traditional machine learning is being continuously optimized, the problem of large amount of model computation has not been changed. At the same time, due to the low resolution and serious interference of sonar images, the design of the feature extraction algorithm and the optimal choice of the classification algorithm have always been the focus of debate on the intelligent classification of seabed sediments.

2.2 Deep Learning

With the rise of deep learning, related research scholars have also begun to consider applying deep learning to the classification of seabed sediment. Compared with the research ideas of traditional machine learning, deep learning can automatically extract the intrinsic features of the target through the internal network structure, and establish a stable feature combination through the abstraction process from low-level to high-level [18], which weakens the subjectivity of artificial selection of features and saves workload.

Xue [19] combined Scale-Invariant Feature Transform (SIFT) with Convolutional Neural Network (CNN) and proved through experiments that CNN has a higher recognition accuracy rate than traditional machine learning algorithms. Fu [20] combines the GLCM with CNN, and the accuracy rate is significantly better than traditional machine learning algorithms in multi-class image classification scenarios. Zhao et al. [21] used the combination of Weyl transform and statistical features to extract the features and input the two-layer neural network to classify the seabed sediment, and obtained better accuracy.

But these studies are still affected by traditional machine learning. They still divided the classification model into two parts: image feature extraction and classifier training, and did not take full advantage of the characteristics of the deep learning network.

Han et al. [22] applied CNN to marine target recognition, and this research classifies sonar images only by building a CNN network. Although the study was to classify seafloor targets, it was not seafloor sediments. But the excellent classification accuracy of its scheme is enough to prove that deep learning technology can correctly process sonar data. The study also describes how simple deep learning models tend to perform better than more complex models for the characteristics of sonar data.

At present, the application of deep learning methods in the classification of seabed sediment is still in the exploratory stage. Therefore, this paper proposes the ShuffleNet-DSE network model to provide a new deep learning solution for the research of seabed sediment classification.

3 PROPOSED METHOD

The ShuffleNet-DSE is a method that can effectively extract sonar image features and analyze inter-channel information more comprehensively. The ShuffleNet-DSE consists of two stages:

1. Feature pre-extraction, and
2. Channel information enhancement.

The pre-extraction part will improve the problem of poor classification accuracy caused by image noise and low image resolution to a certain extent, and perform effective feature extraction on the input sonar image.

The sonar image is a single-channel image, which leads to less information interaction between the channels, and the channel information enhancement part can solve this problem. Figure 1 shows the overall structure of the ShuffleNet-DSE. The details of the model will be explained in this section below.

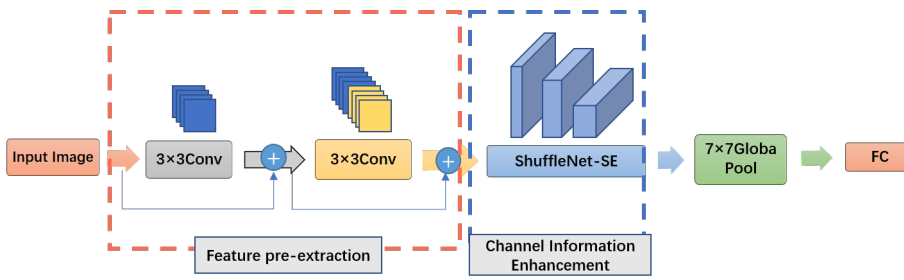


Figure 1. The network structure of ShuffleNet-DSE

3.1 Feature Pre-Extraction

The sonar data contains the noise, and the image resolution is poor, which is often accompanied by great difficulties in the feature extraction process. And the data structure of sonar image is simple, sonar data is not suitable for complex network structure, so it is necessary to extract as much information as possible from the network structure of the image while ensuring the depth. Therefore, it becomes particularly important to make full use of the feature maps output by each layer. These problems can be improved by using the dense connection and the Swish activation functions.

3.1.1 Dense Connection

The output W_i formula of dense connection is as follows:

$$W_i = H_i([x_0, x_1, \ldots, x_{i-1}]). \tag{1}$$

W_i depends on the output of all previous layers. The output of each previous layer is superimposed through the channel as the input of the current layer. It is not difficult to see from the formula that the input of the dense connection is the superposition of the output feature maps of each previous layer. The dense connection allows the model to make full use of all feature maps for learning. Without increasing the depth of the network, this method can increase the learning ability of the model. In the face of uneven quality of sonar images, the conventional connection method is likely to blur the feature information in the convolution calculation of a certain layer. This results in all subsequent layers' computations being based on data whose feature information is obscured. The characteristic information of sonar images is difficult to transmit effectively. These issues can cause large fluctuations in the classification accuracy of the model. However, the dense connection method will superimpose the output feature matrix of each layer with each previous layer, and each layer will consider the calculation results of each previous layer, thereby reducing the possibility of loss of feature information due to a certain layer. the dense connections can improve model stability. And since this approach reduces the depth of the network model, it mitigates the vanishing gradient phenomenon during backpropagation.

3.1.2 Activation Function

The ShuffleNet-DSE uses Swish [23] as the activation function.

ReLU [24] is a commonly used activation function, but it has some flaws in processing sonar data. The calculation formula of ReLU is:

$$f(x) = \begin{cases} 0, & x < 0, \\ x, & x \geq 0. \end{cases} \tag{2}$$

ReLU will make the network sparse, reduce the dependence between parameters, and inhibit the occurrence of overfitting to a certain extent. In addition, when ReLU is backpropagated, the calculation is simple, and it is not easy to cause the gradient to disappear. But the biggest disadvantage of ReLU is that the output has no negative value. ReLU will force the value to be set to 0 when dealing with negative values to achieve the purpose of network sparsity. However, this operation will also cause a large number of features to be shielded, which reduces the model learning feature information. The sonar image is generated as an arrangement of pixel gray values. The sonar image will have negative data as the feature depth increases. If ReLU is used, this part of data will be set to 0, thus losing feature data. The Swish makes up for the shortcoming that the ReLU has no negative value, and it is a non-linear function with a lower bound and no upper bound. Swish is an improved form of ReLU. The calculation formula of Swish is:

$$f(x) = \frac{x}{(1 + \exp(-\beta * x))}. \tag{3}$$

Swish will perform calculations on negative numbers, which also means that Swish's calculation amount is greater than that of ReLU. However, since the sonar image data is not complicated and the network structure of the overall model is relatively simple, the model that selects Swish as the activation function will not increase the amount of computation too much. The nature of the Swish function will change due to different β values, the images of the two activation functions are shown in Figure 2. When

$$\begin{cases} f(x) = \frac{x}{2}, & \beta < 0, \\ f(x) \approx \mathrm{Re}LU, & \beta \to \infty. \end{cases} \tag{4}$$

So, Swish can be regarded as a smooth function between the linear function and the ReLU function. Swish has greater flexibility in the face of diverse data. This is exactly what is expected in sonar image classification. Swish is more suitable for sonar image calculation than ReLU.

3.1.3 Network Structure

The network structure of the pre-extraction part is shown in Figure 3.

The parameter L in Figure 3 is the layer number of the current Block. k is the growth rate, which represents the number of output feature map channels passing through each block.

First, the number of channels is controlled by 1×1 convolution. After calculation, the number of channels of the output feature matrix is $2 \times k$ ($k = 32$). Then a convolution kernel of size 3×3 is used for feature computation. The stride and padding of the convolution kernel are both 1. Effective feature information can be extracted after calculation. The padding is used here to ensure that the input feature matrix and the output feature matrix have the same size. Finally, the feature matrix with the number of channels k is obtained. The output feature matrix is

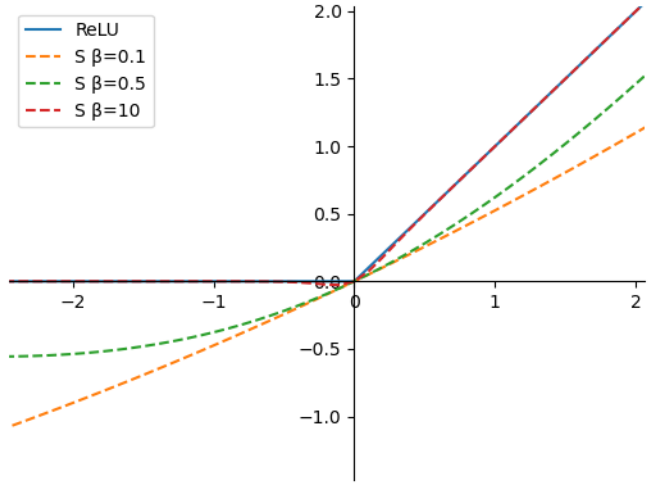


Figure 2. The function diagram of ReLU and Swish

superimposed with the input feature matrix as the input for the next calculation. Repeat the above process to further refine the image feature information. The feature matrix after two superpositions is normalized. And the 1×1 convolution layer is used to reduce the dimensionality of the data. This is to solve the problem that the data dimension is too large due to overlapping feature matrices. Finally, the data is compressed by a max-pooling layer with the size of 2×2 and the stride

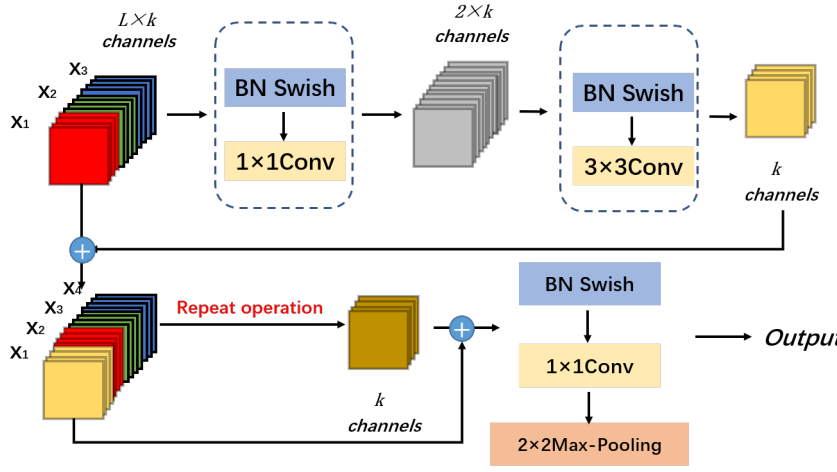


Figure 3. The network structure of pre-extraction part

of 2. The max-pooling layer will reduce the feature map size to 1/2 of the previous size. Max-pooling can reduce the deviation of the estimated mean caused by the parameter error of the convolutional layer, and retain more texture information.

3.2 Channel Information Enhancement

The channel information enhancement part mainly starts from improving the channel information calculation of the sonar image, and uses the three technologies of DWConv, Channel-Shuffle and SE module to process the channel information, which enhances the correlation between different channel information and can filter the noisy channel information. The accuracy and stability of the classification model are greatly increased by the channel information enhancement part.

3.2.1 Depthwise-Separable-Convolution

One convolution kernel of the Depthwise-Separable-Convolution (DWConv) is responsible for one channel, and one channel is convolved by only one convolution kernel. It is equivalent to processing the input feature map as a single-channel image. Due to the single-channel nature of sonar images, DWConv are more suitable than standard convolutions (Conv).

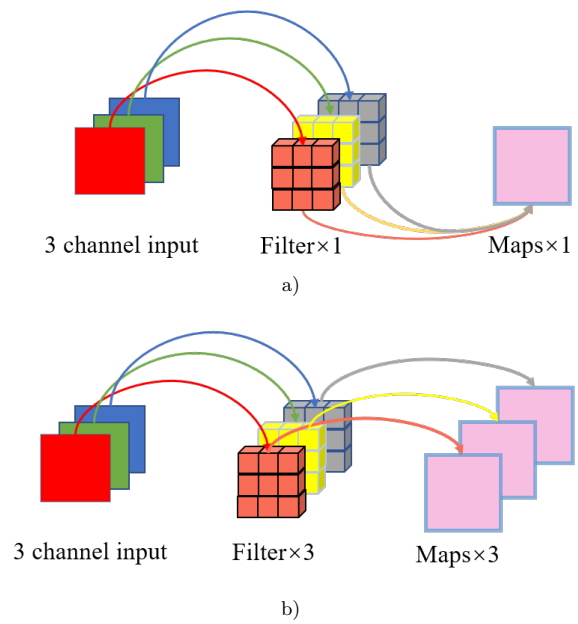


Figure 4. Compare with the Conv and the DWConv

The DWConv can be seen from part a of Figure 4 a), the standard convolution will slide on the input image through a convolution kernel, and the value obtained

by multiplying the weight of the original data and the corresponding position of the convolution kernel will be mapped to the corresponding position of the feature maps. That is to say, each convolution kernel will process the data of all input channels.

In standard convolution, each convolution kernel will process the data of all input channels and map it into a feature map. The calculation formula of the parameter quantity P_F of the standard convolution is:

$$P_F = F_W * F_H * C_N * C_N. \tag{5}$$

Among them, F_W,F_H,C_N are the number of convolution kernels, the scale of the convolution kernels, and the number of input data channels, respectively. As can be seen from Figure 4 b), the principle of DWConv is that each convolution kernel processes an input channel separately to generate a map. In this way, the convolution kernel only processes 2-dimensional spatial information, which reduces the processing of information between different channels. Thus, the calculation formula of the parameter quantity P_D of the DWConv can be obtained as:

$$P_D = F_W * F_H * C_N. \tag{6}$$

Compared with the standard convolution method, DWConv reduces the number of parameters required for calculation and improves the operating efficiency of the model to a certain extent. However, because DWC ignores the information between channels, which will lead to the loss of effective information, it needs to be compensated by the method of the Channel-Shuffle.

3.2.2 Channel-Shuffle

The Channel-Shuffle can make up for the shortcomings of the DWConv. As shown in Figure 5, each convolution processes the data of the same channel group. This will result in the loss of communication between channels, and thus loss of part of the effective information. The concept of Channel-Shuffle is to shuffle and reorder the channels in different channel groups, so that the information of each group is fully integrated without increasing the amount of calculation, thereby solving the previous problems. This technique avoids the phenomenon that each output feature map is independent of each other, and strengthens the learning ability of the model. The implementation method of Channel-Shuffle is as follows:

1. Assuming that the input layer is divided into C groups, the number of channels in each group is N, the total number of channels is $C \times N$, and the reshape operation is performed to output a feature matrix of (C, N) dimensions.
2. Transpose the feature matrix and change the matrix dimension to $(C,\ N)$.
3. Regroup the matrix by row.

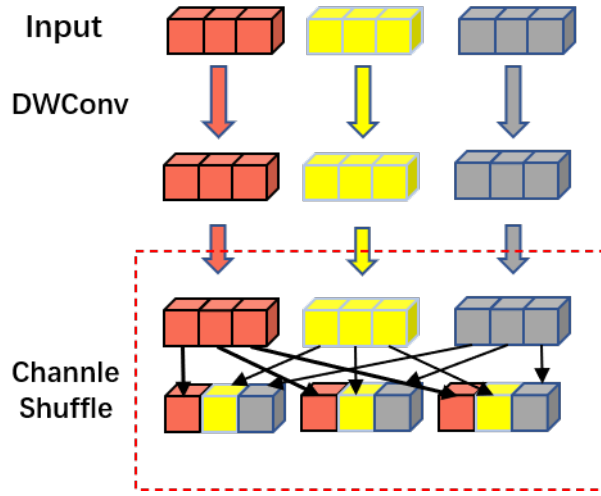


Figure 5. The demonstration of Channel-Shuffle

3.2.3 Squeeze-and-Excitation Module

In 2017, Hu et al. [25] proposed the SE module. The self-attention mechanism of SE module is to obtain the weight factor of each channel by calculating the importance of feature channels. Neural network models focus on learning feature channel data with large weight factors. Sonar images contain complex information, and noise information is more common. Not all channel information helps the classifier to make a correct decision. Therefore, it is necessary to analyze the importance of all channels through the SE module before the Channel-Shuffling to remove redundant channel information. This can increase the anti-interference of the model while improving the classification accuracy.

The SE module structure is shown in Figure 6. The SE module is divided into two parts: Squeeze and Excitation.

The Squeeze part converts the input data of $W \times H \times C$ into $1 \times 1 \times C$ data through the Global pooling layer. Where W, H and C are the characteristic length, width and channel number, respectively.

The main body of the Excitation part is composed of two Fully Connected layers (FC). The first FC layer will reduce the number of channels through $SR \in (0, 1)$ to achieve the purpose of reducing the amount of calculation. Through the second FC layer to restore the number of channels, and through the Sigmoid function to control the weight factor value between (0 1). The lower the weight factor, the closer it is to 0, the higher the weight factor, the closer it is to 1.

Finally, the module will perform a Scale operation. Multiply the input $W \times H \times C$ data and the $1 \times 1 \times C$ weighting factor output by Excitation correspondingly to obtain the feature data with re-calibrated weights.

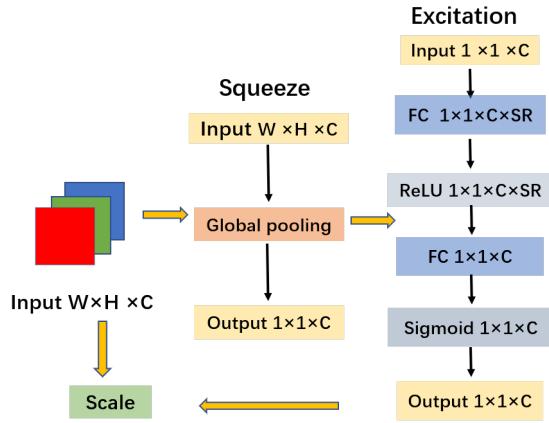


Figure 6. The network structure of SE module

3.2.4 Network Structure

The channel information enhancement part is based on improvements made by ShuffleNet-V2 [26]. The network structure is divided into two parts, A and B, as shown in Figure 7.

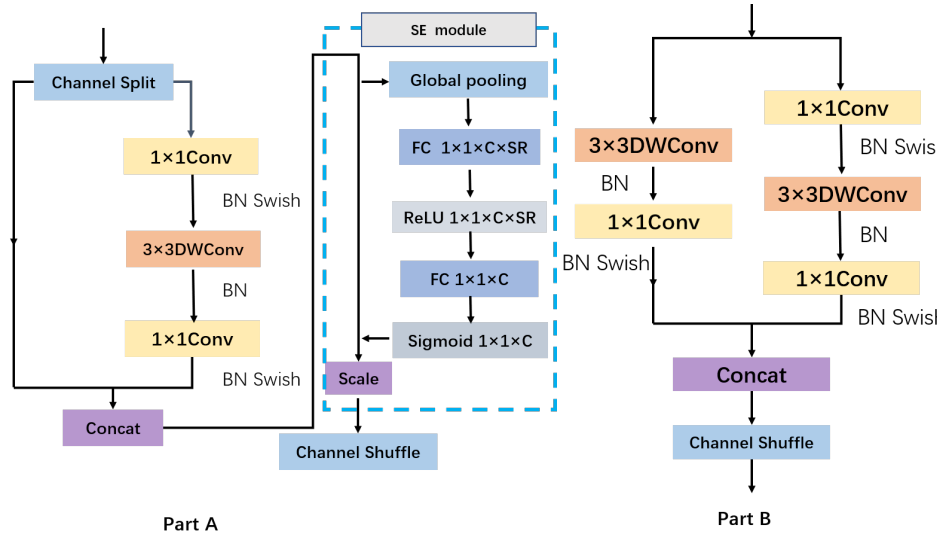


Figure 7. The network structure of channel information enhancement part

Where the channel splitting divides the channel into two equal parts. The left and right parts are spliced by concatenated (Concat). This is different from ResNet's

add. The add is the feature value addition, the Concat is the feature dimension addition. Concat can ensure that the number of input channels is the same as the number of output channels through splicing. The spliced feature matrix will be fused through the Channel Shuffle operation.

3.3 Method Configuration

The network level design of ShuffleNet-DSE is shown in Table 1.

Module	**Layer**	**K-Size**	**Output Channels**
Input			1
	Conv1	3×3	32
Dense Connection	BN + Swish		
	Conv2	1×1	64
	BN + Swish		
	Conv3	3×3	32
	BN + Swish		
	Conv4	1×1	64
	BN + Swish		
	Conv5	1×1	64
	BN + Swish		
	Conv6	3×3	32
	BN + Swish		
	Conv7	1×1	48
	MaxPool	2×2	48
ShuffleNet-SE	A block		
	B block		48
	A block		
	B block		96
	A block		
	B block		192
Summarize	Conv5	1×1	1 024
	Global Pool	7×7	
	FC		

Table 1. Design of ShuffleNet-DSE

4 EXPERIMENT

The traditional machine learning algorithms in the comparative experiment are built by MATLAB, and the deep learning framework is built by paddlepaddle2.0 framework. The CPU running the hardware is Intel Core i7-11800h. The GPU is Tesla V100.

4.1 Contrast Model

The traditional machine selects models GLCM + SOM and GLCM + SVM, and the comprehensive model selects GLCM + CNN and Weyl + ANN.

Deep learning models are divided into two groups, which are:

1. Neural network models with complex network structures: DenseNet [27] and ResNet50 [28].
2. A set of lightweight models with low model complexity, including ResNet18, AlexNet [29], MobileNet-V3 [30] and GoogLeNet [31].

4.2 Datasets

The dataset uses the public sonar image dataset (SAS) of the US Geological Survey and the Weihai real sonar dataset (WHDS) for testing. SAS is divided into 6 categories, each with 60 pictures. All images are 200 × 200 in size. Since there are fewer data images, data augmentation operations are used to expand the dataset. After random flipping, random stretching and random cropping, 2160 images were obtained [32]. The dataset images are clear, and the image classification features are obvious, making it easier to classify.

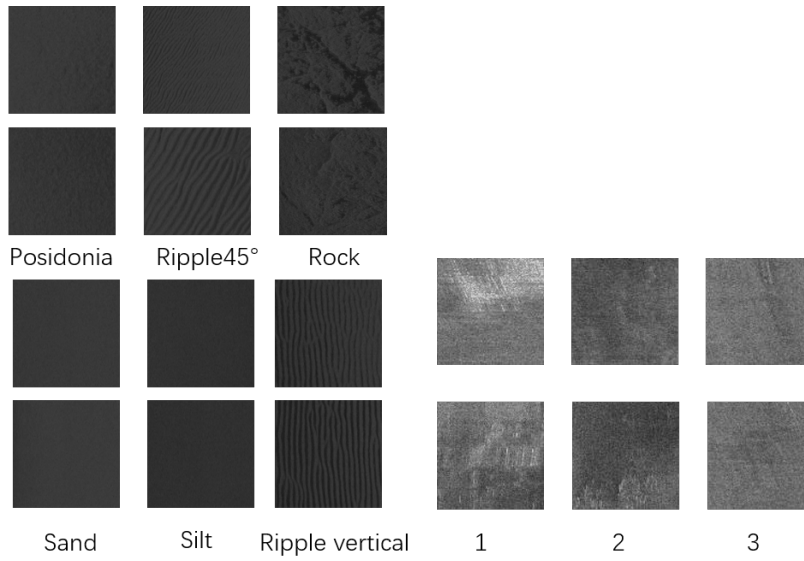
The WHDS dataset is real Weihai data. Compared with the SAS dataset, the image features are fuzzier, but it is more in line with the actual application scenarios. WHDS has greater difficulty in classification, so it is the dataset that the experiment focuses on verification. Crop the picture to 200 × 200 size, generating 516 pictures in total. After labeling, the pictures are divided into three categories. Subsequently, 3 096 images were obtained after random flipping, random stretching and random clipping.

The datasets details are shown in Figure 8 and Table 2.

Dataset	Category	Quantity	Enhancement
SAS	Posidonia	60	360
	Ripple45°	60	360
	Rock	60	360
	Sand	60	360
	Silt	60	360
	Ripple vertical	60	360
WHDS	1	203	1 218
	2	153	918
	3	160	960

Table 2. Datasets quantity information

All datasets divide 80 % of the data into the training set, and 20 % of the data into the test set.



a) SAS b) WHDS

Figure 8. Dataset category details

During the test, it was found that the accuracy of the algorithm differed slightly with each training recognition. In order to reduce the fluctuation of the test results, the test was repeated 5 times. The average data of 5 times is used as the evaluation data.

4.3 Evaluation Measures

Use confusion matrix to obtain *Precision*, *Accuracy*, *Recall rate*, *F1* index to evaluate the performance of classification model. The specific evaluation formula is as follows:

$$Accuracy = \frac{TP + TN}{m}, \tag{7}$$

$$Precision = \frac{TP}{TP + FP}, \tag{8}$$

$$Recall = \frac{TP}{TP + FN}, \tag{9}$$

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall}. \tag{10}$$

At the same time, the inference time T of the model is also one of the evaluation indicators, which is used to measure the complexity of the model.

4.4 Experimental Result

In the case of using dataset SAS, the performance data of all models are shown in Table 3. As the SAS data is a professional image with obvious texture features, the accuracy of all models can be maintained at around 95 %. But through F1, it can be seen that the stability of the comprehensive model and the deep learning model is better. In the deep learning method, it can be seen that the ResNet series and DenseNet series with deep structure are difficult to achieve excellent performance, and as the model complexity increases, the model classification accuracy gradually decreases. This is because they have obvious overfitting during the training process. Although the training set achieves 100 % accuracy, the performance on the test set is not satisfactory. In this dataset, the gap between various models is small, it can still be seen through the accuracy and other indicators that the deep learning network with small complexity has better performance. The ShuffleNet-DSE has an accuracy of 98.81 % and a precision of 98.33 %. The accuracy of Shufflenet-DSE is the highest among comparison models. The accuracy of ShuffleNet-DSE is 0.65 % higher than that of ShuffleNet-V2. The precision of ShuffleNet-DSE is 98.33 %, which is much higher than other deep learning models. At the same time, the F1 value of ShuffleNet-DSE is 0.98, which is also the best among many compared models. Under comprehensive consideration, the performance of ShuffleNet-DSE is the most excellent model.

The particularity of SAS data does not completely represent the superiority of the classification model with performance, so it is necessary to continue the experiment using WHDS.

MODEL	**Evaluation Index**			
	Accuracy (%)	***Precision (%)***	***Recall***	***F1***
GLCM + SVM	94.05	92.5	0.94	0.93
GLCM + SOM	93.22	88.05	0.91	0.90
Weyl + ANN	95.69	93.36	0.91	0.94
GLCM + CNN	95.16	97.23	0.95	0.96
ResNet50	93.05	91.67	0.95	0.92
ResNet101	90.67	90.24	0.92	0.91
ResNet152	90.84	90.11	0.9	0.90
DenseNet121	90.46	93.05	0.81	0.91
DenseNet161	89.06	92.53	0.72	0.90
ResNet18	96.00	95.23	0.89	0.94
MobileNet-V3	95.79	96.59	0.93	0.96
AlexNet	96.91	95.24	0.94	0.96
GoogLeNet	97.52	95.85	0.87	0.97
ShuffleNet-V2	98.16	94.71	0.95	0.96
ShuffleNet-DSE	98.81	98.33	0.96	0.98

Table 3. Performance comparison of classification models in SAS dataset

WHDS data is conventional sonar image data, which has the influence of noise. The texture features of the data are also fuzzy. After manual labeling, the data is divided into three categories. WHDS can really test whether the model can show satisfactory performance in practical applications. On the WHDS dataset, there is a huge gap between different models.

The performance data of each model are shown in Table 4. It can be seen that traditional machine learning is difficult to adapt to sonar image data, and the accuracy of the two models is difficult to reach 80 %. And the accuracy is also around 72 %, which is a poor level among all comparison models.

In most public datasets, ResNet series and DenseNet series perform better than MobileNet series and ShuffleNet series of lightweight networks. However, with its characteristic sonar imagery, this has been reversed. Also, unlike other image types, the classification accuracy of deep learning networks decreases as the number of layers in the network increases. Comparing ResNet with three depths of 50, 101, and 152, the accuracy dropped from 73.77 % to 69.74 %. DenseNet also reduces the accuracy from 74.06 % to 72.87 % when the depth is changed from 121 to 161. The deep network structures of the ResNet series and DenseNet series are prone to overfitting when using sonar images as datasets, resulting in a disproportion between the overall model performance and the model complexity.

Without considering the model's inference time T, the performance of the comprehensive model is superior and can be on par with most lightweight deep learning models, with an accuracy of over 80 %. However, the computational complexity of the synthetic model is far greater than that of the lightweight deep learning model.

Lightweight deep learning models are suitable for processing sonar image data. When ResNet uses 18 as the depth configuration, the accuracy improves from 73.77 % to 80.49 %. Among the lightweight deep learning models, the ShuffleNet-DSE and the ShuffleNet-V2 have the best accuracy, 94.19 % and 92.1 %. Both the ShuffleNet-DSE and the ShuffleNet-V2 adopt the techniques of DWConv and Channel-Shuffle, which also shows the superiority of these two techniques in processing sonar image data. The MobileNet-V3 has SE modules in its network structure. The MobileNet-V3 has excellent precision, which is 3.24 % higher than that of ShuffleNet-V2. The ShuffleNet-DSE model, which also has the SE module, successfully makes up for the poor precision of ShuffleNet-V2. The precision of ShuffleNet-DSE improves from 84.71 % of ShuffleNet-V2 to 86.17 %. The SE module can improve the precision of the model.

The ShuffleNet-DSE is improved in terms of accuracy and precision. Compared with the original model, the accuracy of ShuffleNet-DSE is improved from 92.10 % of ShuffleNet-V2 to 94.19 %. Compared with the comprehensive learning model Weyl + ANN and GLCM + CNN, the accuracy is improved by 11.28 % and 10.83 %, respectively.

Combined with the experimental results of two different datasets, the ShuffleNet-DSE has good accuracy and stability in processing sonar image data.

Compare the accuracy of several groups of models and the model inference time T. The result is shown in Figure 10.

MODEL	Evaluation Index			
	Accuracy (%)	***Precision (%)***	***Recall***	***F1***
GLCM + SVM	75.16	72.09	0.81	0.73
GLCM + SOM	78.23	70.28	0.83	0.74
Weyl + ANN	82.91	82.28	0.87	0.82
GLCM + CNN	83.36	81.23	0.85	0.83
ResNet50	73.77	73.85	0.85	0.73
ResNet101	72.81	67.67	0.9	0.70
ResNet152	69.74	67.05	0.86	0.68
DenseNet121	74.06	71.02	0.81	0.72
DenseNet161	72.87	69.53	0.72	0.71
ResNet18	80.49	85.46	0.89	0.94
MobileNet-V3	83.96	87.95	0.83	0.85
AlexNet	85.27	80.21	0.84	0.82
GoogLeNet	87.82	84.27	0.85	0.86
ShuffleNet-V2	92.1	84.71	0.909	0.88
ShuffleNet-DSE	94.19	86.17	0.904	0.90

Table 4. Performance comparison of classification models in WHDS dataset

As can be seen from Figure 9, the deep learning models perform better than the comprehensive model when making inferences. The deep learning models eliminate the need for an additional feature extraction process, since they use their own network structure to extract features, what dramatically reduces the computational costs. When the *batchsize* = 4, the inference time of the Weyl + ANN and the GLCM + CNN is around 12 ms, but the lightweight deep learning models can control the inference time within 6 ms. Despite some sacrifices in model lightweighting, the ShuffleNet-DSE is still able to complete inference within 4.12 ms. The model complexity of ShuffleNet-DSE is still at a relatively low level. The ShuffleNet-DSE has excellent classification accuracy under the premise of ensuring low model complexity.

4.5 Comparative Analysis

Experiments will explore the impact of different techniques on the performance of the final model. The experimental dataset is WHDS.

The results of the four groups of experiments are shown in Figure 10. There are five models in the figure:

Original model: The original model of ShuffleNet-DSE, the activation function of the model is Swish.

ReLU: Replace all Swish in the ShuffleNet-DSE with ReLU.

Convolution: Change the combination of the DWConv and the Channel-Shuffle in ShuffleNet-DSE to standard convolution kernel.

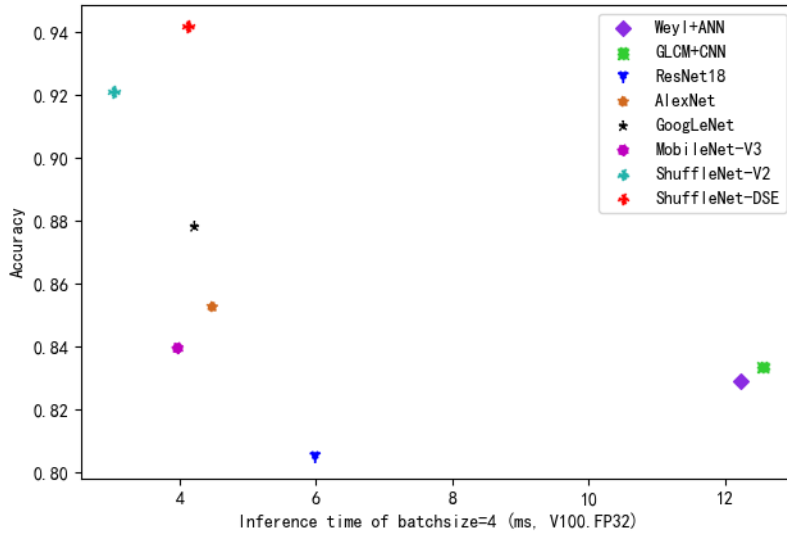


Figure 9. Accuracy and inference time of different models

Without SE: Delete the SE module in ShuffleNet-DSE.

Normal Connection: Change the dense connection in ShuffleNet-DSE to the normal connection.

When the original model uses Swish as the activation function, the accuracy is between 93.83 % and 94.67 %, the average accuracy of ten experiments is 94.22 %, and the accuracy fluctuation range is 0.84 %. When Swish is replaced by ReLU, the model's accuracy is between 92.37 % and 92.89 %, the average accuracy of ten experiments is 92.69 %, and the fluctuation range of the accuracy is 0.52 %. Since ReLU uniformly sets negative features to 0, ReLU has better stability, the degree of fluctuation is small. But compared to ReLU, the average accuracy of Swish is improved by 1.53 %. The Swish has better accuracy, so Swish is more suitable for processing sonar image data.

After replacing the DWConv and the Channel-Shuffle with standard convolutions, the accuracy of the model ranged from 87.85 % to 88.46 %, the average accuracy of ten experiments is 88.16 %, and the accuracy fluctuation range is 0.61 %. Compared to using the DWConv and the Channel-Shuffle, the accuracy drops significantly, with an average accuracy drop of 6.06 %. It can be seen that the DWConv and the Channel-Shuffle technology are the key technology to ensure the classification accuracy, which also shows that the problem that sonar images are difficult to be accurately classified can be solved from the information between channels.

After removing the SE module in ShuffleNet-DSE, the model accuracy is between 92.83 % and 94.25 %, the average accuracy of ten experiments is 93.36 %, and the accuracy fluctuation range is 1.42 %. After removing the SE module, the ac-

curacy decreased slightly, the average accuracy decreased by 0.86 %, the accuracy fluctuation increased greatly, and the fluctuation increased by 0.6 %. It can be seen that the main function of the SE module is to increase the stability of the model, reduce the performance fluctuation, and improve the accuracy slightly.

After using the normal connection method, the model accuracy is between 92.86 % and 93.99 %, the average accuracy of ten experiments is 93.57 %, and the accuracy fluctuation range is 1.13 %. Compared with the dense connection method, the accuracy decreases slightly, and the average accuracy decreases by 0.65 %. At the same time, the accuracy fluctuation range increased slightly, and the fluctuation range increased by 0.29 %. Using the dense connection can enhance the noise resistance of the model and reduce the impact of feature calculation errors on the final result.

Using the Swish, the DWConv and the Channel-Shuffle will mainly improve the accuracy of the model. Swish increases the calculation of negative features and reduces the loss of feature values. The combination of the DWConv and the Channel-Shuffle can use different channel information more effectively. The SE module and the dense connection method can reduce the adverse effects of image noise and low image resolution, thereby increasing the stability of the model. By combining the advantages of each technique, the ShuffleNet-DSE is able to better classify sonar images of seafloor sediments.

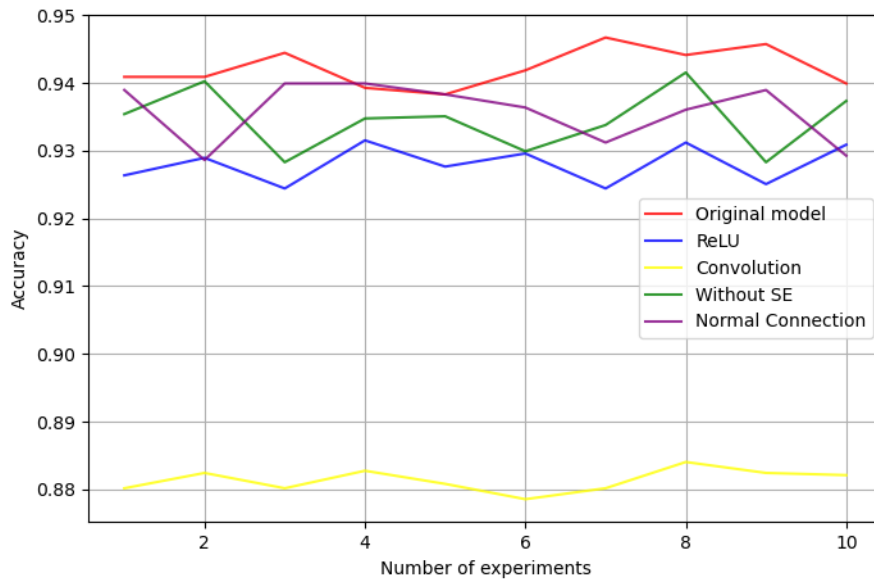


Figure 10. Accuracy of different models in ten experiments

4.6 Model Training Parameters

The training parameters of the main comparison models in this experiment are shown in Table 5. Mainly use the Momentum-Optimizer ($momentum = 0.9$). The regular term mainly uses L2 ($coeff = 0.00001$). The learning rate of ResNet18 and AlexNet adopts Piecewise Decay, and the learning rate is changed when the training rounds are 30, 60 and 90, respectively.

MODEL	Optimizer		
	Technology	***Learning rate***	***Regularizer***
ResNet18	Momentum	Piecewise Decay decay_epochs: [30, 60, 90] values: [0.1, 0.01, 0.001, 0.0001]	L2
MobileNet-V3	Momentum	1.3	L2
AlexNet	Momentum	Piecewise Decay decay_epochs: [30, 60, 90] values: [0.01, 0.001, 0.0001, 0.00001]	L2
GoogLeNet	Momentum	0.001	L2
ShuffleNet-V2	Momentum	0.0125	L2
ShuffleNet-DSE	Momentum	0.0125	L2

Table 5. Main training parameters

5 CONCLUSION AND FUTURE WORK

To solve the issue of difficulty to accurately classify the seabed sediment of the sonar image, this paper designed a seabed sediment classification model based on deep learning for sonar images. Based on the improvement of the ShuffleNet-V2 unit structure, combined with the SE module, Swish activation function and dense connection module, constructs a ShuffleNet-DSE network model. The ShuffleNet-DSE architecture is designed at the expense of minimal model efficiency, and greatly improves the accuracy and stability of the model in processing seabed sediment image classification. Experimental results on the public dataset SAS and Weihai dataset collected at sea demonstrated that this proposed model can attain the favorable performance achievable.

Due to the limitation of the amount of sample image data, it has a certain impact on the experimental results. Although the dataset is expanded by means of data enhancement, the actual seabed sediment image is not increased. Therefore, the experimental conclusion lacks universal applicability to a certain extent.

Future work will use more real seabed sediment image data for experiments to further improve the model structure.

Acknowledgements

This work was supported in part by Scientific Research Projects of the Education Department of Liaoning Province (No. LJKZ0537, No. J2020113).

REFERENCES

[1] Shih, C. C.—Horng, M. F.—Tseng, Y. R.—Su, C. F.—Chen, C. Y.: An Adaptive Bottom Tracking Algorithm for Side-Scan Sonar Seabed Mapping. 2019 IEEE Underwater Technology (UT), IEEE, 2019, pp. 1–7, doi: 10.1109/UT.2019.8734291.

[2] Bu, J. W.—Yan, T. N.—Chang, Z. J.: Introduction to the Status Quo and Operating Principle of Seabed Samplers – Part 1 of the Subject on Seabed Sampling. Exploration Engineering (Drilling and Tunneling), Vol. 2, 2001, pp. 44–48, doi: 10.3969/j.issn.1672-7428.2001.02.021 (in Chinese).

[3] Geng, X. Q.—Xu, X.—Liu, F. L.—Zhang, Z. G.—Chen, Q.: The Current Status and Development Trends of Marine Sampling Equipment. Equipment for Geotechnical Engineering, Vol. 10, 2009, No. 4, pp. 11–16, doi: 10.3969/j.issn.1009-282X.2009.04.002 (in Chinese).

[4] Lu, X. L.—Zhang, L. T.—Wang, F.—Su, J.: Summary of Submarine Acoustic Detection Technology and Equipment. Ocean Development and Management, Vol. 35, 2018, No. 6, pp. 91–94, doi: 10.3969/j.issn.1005-9857.2018.06.020 (in Chinese).

[5] Jiang, Y. M.—Chapman, N. R.—Gerstoft, P.: Estimation of Geoacoustic Properties of Marine Sediment Using a Hybrid Differential Evolution Inversion Method. IEEE Journal of Oceanic Engineering, Vol. 35, 2010, No. 1, pp. 59–69, doi: 10.1109/JOE.2009.2025904.

[6] Qin, X.—Luo, X.—Wu, Z.—Shang, J.: Optimizing the Sediment Classification of Small Side-Scan Sonar Images Based on Deep Learning. IEEE Access, Vol. 9, 2021, pp. 29416–29428, doi: 10.1109/ACCESS.2021.3052206.

[7] Zhao, J.—Wang, X.—Zhang, H.—Wang, A.: A Comprehensive Bottom-Tracking Method for Sidescan Sonar Image Influenced by Complicated Measuring Environment. IEEE Journal of Oceanic Engineering, Vol. 42, 2016, No. 3, pp. 619–631, doi: 10.1109/JOE.2016.2602642.

[8] Frederick, C.—Villar, S.—Michalopoulou, Z. H.: Seabed Classification Using Physics-Based Modeling and Machine Learning. The Journal of the Acoustical Society of America, Vol. 148, 2020, No. 2, pp. 859–872, doi: 10.1121/10.0001728.

[9] Jin, X. L.: The Development of Research in Marine Geophysics and Acoustic Technology for Submarine Exploration. Progress in Geophysics, Vol. 22, 2007, No. 4, pp. 1243–1249, doi: 10.3969/j.issn.1004-2903.2007.04.034 (in Chinese).

[10] Nissen, P.—Dame, R.—Carothers, J.—Suarez, G.—Pratt, C. et al.: Side-Scan_Sonar Backscatter Tiles for Hudson River, NY (.xtf) from 2010-06-15 to 2010-08-15. Data Set. 2012, `https://www.fisheries.noaa.gov/inport/item/47922`.

[11] Lu, L.—Jin, S. H.—Bian, G.—Cui, Y.—Xia, W.: The Application of K-Means Clustering Analysis Algorithm in Multibeam Seafloor Classification. Hydrographic

Surveying and Charting, Vol. 38, 2018, No. 3, pp. 64–68, doi: 10.3969/j.issn.1671-3044.2018.03.016 (in Chinese).

[12] Kohonen, T.: Self-Organization and Associative Memory. Springer, Berlin, Heidelberg, Springer Series in Information Sciences, Vol. 8, 2012, doi: 10.1007/978-3-642-88163-3.

[13] Tang, Q. H.—Liu, B. H.—Chen, Y. Q.—Zhou, X. H.—Ding, J. S.: Acoustic Seafloor Classification Using Self-Organizing Map Neural Network. Technical Acoustics (Shengxue Jishu), Vol. 26, 2007, No. 3, pp. 380–384 (in Chinese).

[14] Xiong, M.—Wu, Z.—Li, S.—Luo, X.—Tang, Q.: Seafloor Sonar Sediment Image Recognition with the Support Vector Machine. Marine Science Bulletin, Vol. 31, 2012, No. 4, pp. 409–414 (in Chinese).

[15] Xu, C.—Li, H. S.—Wang, C.—Zhao, X. L.: Seabed Classification of Multibeam Seabed Acoustic Image Based on Composite Kernel SVM. Progress in Geophysics, Vol. 29, 2014, No. 5, pp. 2437–2442, doi: 10.6038/pg20140567 (in Chinese).

[16] Yang, F.—Li, J.—Zhao, J.—Du, Z.: Seabed Classification Using BP Neural Network Based on GA. Science of Surveying and Mapping, Vol. 36, 2006, No. 2, pp. 111–114, doi: 10.3771/j.issn.1009-2307.2006.02.038.

[17] Xiong, M. K.—Wu, Z. Y.—Li, S. J.—Shang, J. H.: Wavelet Neural Network Identification and Classification of Sediment Seabed Sonar Images Based on Genetic Algorithms. Acta Oceanologica Sinica, Vol. 36, 2014, No. 5, pp. 90–97, doi: 10.3969/j.issn.0253-4193.2014.05.010 (in Chinese).

[18] Yang, F.—Choi, W.—Lin, Y.: Exploit All the Layers: Fast and Accurate CNN Object Detector with Scale Dependent Pooling and Cascaded Rejection Classifiers. Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 2129–2137, doi: 10.1109/CVPR.2016.234.

[19] Xue, J.: Classification of Seabed Sediment and Terrain Complexity Based on Multibeam Data. Ph.D. Thesis. Qingdao, First Institute of Oceanography, Ministry of Natural Resources of China, 2017 (in Chinese).

[20] Fu, N.: Research on Classification Method of Submarine Substrate Type Based on Characteristics of Sonar Image. Ph.D. Thesis. Harbin, Harbin Engineering University, 2019 (in Chinese).

[21] Zhao, T.—Lazendić, S.—Zhao, Y.—Montereale-Gavazzi, G.—Pižurica, A.: Classification of Multibeam Sonar Image Using the Weyl Transform. In: Choraś, M., Choraś, R. (Eds.): Image Processing and Communications (IP & C 2019). Springer, Cham, Advances in Intelligent Systems and Computing, Vol. 1062, 2019, pp. 206–213, doi: 10.1007/978-3-030-31254-1_25.

[22] Han, S.—Mao, H.—Dally, W. J.: Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding. 2015, doi: 10.48550/arXiv.1510.00149.

[23] Ramachandran, P.—Zoph, B.—Le, Q. V.: Searching for Activation Functions. 2017, doi: 10.48550/arXiv.1710.05941.

[24] He, J.—Li, L.—Xu, J.—Zheng, C.: ReLU Deep Neural Networks and Linear Finite Elements. Journal of Computational Mathematics, Vol. 38, 2020, pp. 502–527, doi: 10.4208/jcm.1901-m2018-0160.

[25] Hu, J.—Shen, L.—Sun, G.: Squeeze-and-Excitation Networks. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 7132–7141, doi: 10.48550/arXiv.1709.01507.

[26] Zhang, X.—Zhou, X.—Lin, M.—Sun, J.: ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 6848–6856, doi: 10.48550/arXiv.1707.01083.

[27] Huang, G.—Liu, Z.—Van Der Maaten, L.—Weinberger, K. Q.: Densely Connected Convolutional Networks. Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 2261–2269, doi: 10.1109/CVPR.2017.243.

[28] He, K.—Zhang, X.—Ren, S.—Sun, J.: Deep Residual Learning for Image Recognition. Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[29] Krizhevsky, A.—Sutskever, I.—Hinton, G. E.: ImageNet Classification with Deep Convolutional Neural Networks. Advances in Neural Information Processing Systems 25 (NIPS 2012), 2012, pp. 1097–1105.

[30] Howard, A.—Sandler, M.—Chu, G.—Chen, L. C.—Chen, B.—Tan, M.—Wang, W.—Zhu, Y.—Pang, R.—Vasudevan, V.—Le, Q. V.—Adam, H.: Searching for MobileNetV3. Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 1314–1324, doi: 10.48550/arXiv.1905.02244.

[31] Szegedy, C.—Liu, W.—Jia, Y.—Sermanet, P.—Reed, S.—Anguelov, D.—Erhan, D.—Vanhoucke, V.—Rabinovich, A.: Going Deeper with Convolutions. Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1–9, doi: 10.1109/CVPR.2015.7298594.

[32] Cubuk, E. D.—Zoph, B.—Mane, D.—Vasudevan, V.—Le, Q. V.: AutoAugment: Learning Augmentation Policies from Data. 2018, doi: 10.48550/arXiv.1805.09501.

Yuhong Shi received his B.Sc. degree in communication engineering at the Jiangsu University. Currently he is studying for his Master's degree at the School of Information Science and Engineering, Dalian Polytechnic University. His research focuses on deep learning.

Mo Chen received her B.Sc. degree in software engineering at the Qingdao University. Currently she is studying for her Master's degree at the School of Information Science and Engineering, Dalian Polytechnic University. Her research focuses on deep learning and NLP.

Chunlong Yao received his B.Eng. degree in computer and its application from the Northeast Heavy Machinery Institute, Qiqihar, China, in 1994; his M.Eng. degree in computer application technology from the Northeast Heavy Machinery Institute, Qiqihar, China, in 1997; and his Ph.D. degree in computer software and theory from the Harbin Institute of Technology, Harbin, China, in 2005. He is currently Professor and Supervisor of postgraduate students at the Dalian Polytechnic University, Dalian, China. His current research interests include data mining and intelligent information system.

Xu Li received her Ph.D. degree in computer science from the Yanshan University, China. She is currently Associate Professor with the Dalian Polytechnic University, Dalian, China. She has published more than 20 articles in national and international journals and conference proceedings. Her research interests include natural language processing and deep learning.

Lan Shen received her M.Eng. degree in computer software and theory from the Dalian Maritime University, Dalian, China, in 2007. She is currently Associate Professor and Supervisor of postgraduate students at the Dalian Polytechnic University, Dalian, China. Her current research interests include application of internet of things technology and edge computing.