EAI Endorsed Transactions
on Energy Web

Research Article EAI.EU

Energy-Efficient Design of Seabed Substrate Detection Model Leveraging CNN-SVM Architecture and Sonar Data

Keming Wang[1], Chengli Wang[2], Wenbing Jin[1]*, Liuming Qi[3]

[1]School of Internet of Things Technology, Hangzhou Polytechnic, Hangzhou, 311402, Zhejiang, PR China
[2]Hangzhou SECK Intelligent Technology Co., Ltd., Hangzhou, 311121, Zhejiang, PR China
[3]Hangzhou Institute of Applied Acoustics, Hangzhou, 310012, Zhejiang, PR China
*Corresponding author: WenbingJin@yeah.net

Abstract

This study introduces an innovative seabed substrate detection model that harnesses the complementary strengths of Convolutional Neural Networks (CNNs) and Support Vector Machines (SVMs) to analyze sonar data with a focus on energy efficiency. The model addresses the challenges of underwater sensing and imaging, including variable lighting conditions, backscattering effects, and acoustic sensor limitations, while minimizing energy consumption. By leveraging advanced machine learning techniques, the proposed model aims to enhance seabed classification accuracy, a crucial aspect for marine operations, ecological studies, and energy-intensive underwater applications.The introduced ShuffleNet-DSE architecture demonstrates significant improvements in both accuracy and stability for seabed sediment image classification, while maintaining energy-efficient performance. This robust tool offers a valuable asset for underwater exploration, research, and monitoring efforts, especially in environments where energy resources are limited.

Keywords: CNN, SVM, SONAR, ShuffleNet-DSE, backscattering

Received on 7 March 2024, accepted on 28 May 2024, published on 4 June 2024

doi: 10.4108/ew.6097

1. Introduction

The study of seabed is essential for a viable model as seas, deep oceans are the largest part of the earth ecosystem yet least explored. The depth of a deep ocean starts at approximately 200 metres and ends at around 11,000 metres. Subsequently the oceans have diversified ecosystems of plants and animals thus it is important to study and comprehend the geological as well as ecological attributes of the sea bed for the sake of numerous applications. However mostly conventional methods are utilised for the study of sea bed such as sea floor photography and towed and bottom samplers [1], but these methods have poor economic efficiency, rely on heavy machinery and high-intensity labor, and are time-consuming and labor-intensive. Therefore, in order to overcome the above limitations, it is urgent to propose new intelligent technologies. In view of this, research is being conducted to improve the accuracy and energy efficiency of seabed classification through modern underwater sensing technology and machine learning algorithms. The innovative model introduced in the study combines CNN and SVM, utilizing the ShuffleNet DSE architecture to improve the accuracy and stability of seabed sediment image classification, while also considering energy performance.

But with the advancement in technology such methods can now be mitigated by techniques like underwater sensing technology, various machine learning algorithms like convolution neural networks and support vector machines. Underwater sensing technology basically is the acquisition of the data from the deep water environments comprising of various sensors like single beam echosound, multi beam echosounders, side scan sound navigation and ranging where each sensors are designed for specific task and application such as navigation, object inspection, sea floor mapping etc. however for optical sensors. These sensors include underwater cameras, 3D underwater structured light sensors, and hyperspectral sensors. To explore large and deep areas of the sea, autonomous underwater vehicles (AUVs) and remotely operated vehicles (ROVs) are essential [2]. These mobile robots carry sensors, enabling real-time data collection and processing at depths beyond human reach. Recent research has focused on the deep sea due to valuable mineral resources, such as polymetallic nodules containing critical metals like nickel, copper, cobalt, manganese, and rare earth elements. These nodules hold potential as raw materials for various industries [3].

Another crucial application that can benefit from improved seabed classification is deep coral reef monitoring. Coral reefs play a vital role in ocean health, hosting a rich diversity of species . Unfortunately, due to human activities and climate change, coral species are facing endangerment . Recent studies highlight the rapid decline of certain coral species, with factors like bleaching and reef structure degradation contributing to their vulnerability [4]. Monitoring these species and conducting regular assessments of their status are essential steps toward achieving sustainable development and preserving these fragile ecosystems. And to classify a seabed following sensing technology are utlized as shown in the figure such as classification, detection and segmentation that also helps in collecting data for further research analysis.

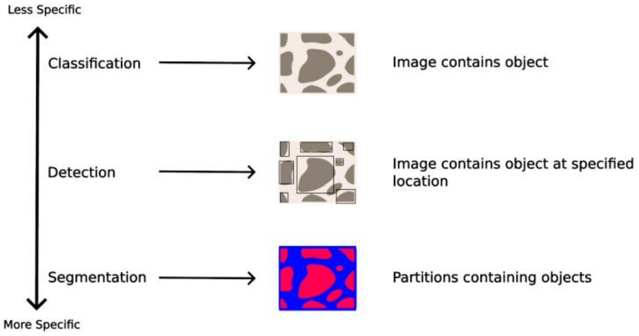


Figure 1. Seabed characterization

Beside the sensing techniques we have new advanced techniques like deep learning and machine learning that mostly use computational power to analyse the prior data to extract results such as from sonar data, commonly used techniques are support vector machine and convolution neural network. Support vector machines (SVM) are elegant and effective methods for classification. That is to classify objects into two or more categories, each object is represented as a point in an n-dimensional space, with its coordinates serving as features. SVMs perform classification by drawing a hyperplane (a line in 2D or a plane in 3D) such that all points of one category lie on one side of the hyperplane, and all points of the other category lie on the opposite side. As shown in the figure 2

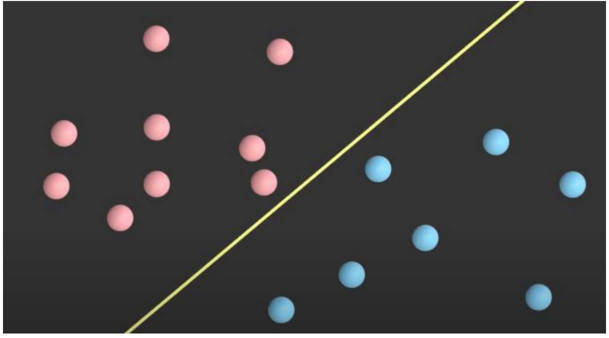


Figure 2. SVM hyper plane

The goal is to maximize the margin—the distance to the nearest points—between the two categories. The points falling exactly on the margin are called supporting vectors. To find this optimal hyperplane, SVMs require a labeled training set. In contrast SVM solve a convex optimization problem that maximizes the margin while ensuring that points from each category fall on the correct side of the hyperplane. As shown by the following equation

$$wTz + b \quad 1\ x \quad C1$$
$$wTz + b \quad -1\ x \quad C2$$

In practice, using SVMs is straightforward: load a Python library, prepare your training data, feed it to the fit function, and call predict to assign categories to new objects. The simplicity of SVMs makes them easy to understand and implement. However, in cases where points cannot be separated by a hyperplane, techniques like kernel tricks can be utilised that allow efficient handling of non-linear data.

Convolution Neural Network

Convolutional Neural Networks (CNNs) are a specialised type of neural network designed primarily for image processing tasks, though they can also be applied to natural language processing. Unlike regular neural networks that have fully connected layers, CNNs treat data as spatial, maintaining the spatial hierarchy of the input data. This is achieved through a unique architecture that includes convolutional layers, pooling layers, ReLU layers, and fully connected layers. The convolutional layer applies filters to the input image to create feature maps, simplifying the image for better processing. The pooling layer further reduces the data size, speeding up computation. ReLU layers introduce non-linearity, preserving the complexity of the data. Finally, the fully connected layer performs classification. Training CNNs can be done using unsupervised methods like autoencoders and GANs.

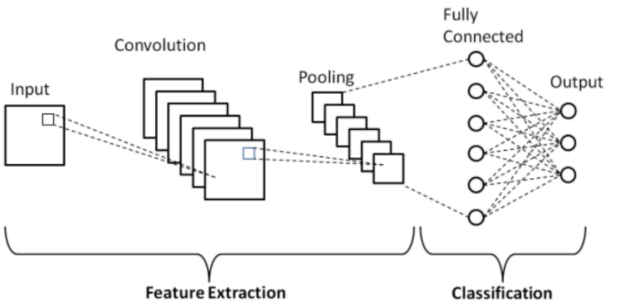


Figure 3. Schematic diagram of convolution neural network source [5]

In relation to Object detection architectures, it can be categorized into two main types: one-stage and two-stage approaches. One-stage approaches, exemplified by SSD (Single-Shot Multi-Box Detector) and YOLO (You Only Look Once), utilize a single pass of a Convolutional Neural Network (CNN) to predict bounding boxes and associated probabilities for object classes. SSD combines a backbone model (often a pre-trained image classification network) with an SSD head that interprets outputs as bounding boxes and classes. YOLO, on the other hand, generates multiple bounding boxes per object and suppresses redundant ones. In contrast, two-stage detectors, including Fast R-CNN, Faster R-CNN, and Mask R-CNN, involve a proposal stage to identify regions of interest (ROIs) containing potential objects, followed by a second stage for object classification. These architectures play a crucial role in tasks like object localization and recognition, finding applications in computer vision and robotics[6].

Characterization of Seafloor

Seafloor can be categorized based on the topographical, geological, biological and physical attributes of the seabed. And to discern among the properties various sensors can be utilized to inspect and collect data for processing, further knowledge about seabed is crucial for application like marine operation, underwater inspection, geological surveys object detecction etc. however in the time of inspection of the seabed various underwater challenges could encounter such as high pressure of the water on the machinery, execessive power draw, transmitting of data from the sensor in general a typical underwater computer vision system encoutner some of the followign challenges.

Lighting Conditions in Water

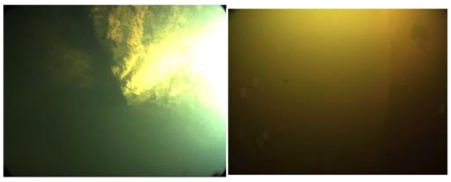
Water absorbs light across various wavelengths, with blue and green wavelengths attenuated less. Color changes due to light attenuation depend on factors like wavelength, dissolved organic compounds, salinity, and phytoplankton concentration. Image-enhancement algorithms fall into three categories: statistical, physical, and learning-based [7].

Challenges in Underwater Imaging

Challenges like backscattering, texture, vignetting effect and lac of ambient light creates problem for underwater sensing backscattering happens when reflection of light diffuse which cause blurring and noise similarly suspended particles scatter the light coming towards the camera, subsequently hence textures are indiscernible underwsater so identifgying various features of the environement are quiet challenging similarly vignetting effect reduces the brightness towards the image periphery compared to the center and since scarcity of ambient light tends to to use artificial lightning but since uneven distrubtion of light and close light sources pose difficulites.

Issues with Acoustic Sensors:

Underwater environments are noisy, affecting accurate signal transmission and reception thus leads to noisy signal also signal are atteneauted because of absorption and also spreading loss impact acoustic energy. Subsequently due to refraction in the water for single signal multiples path are available for transmission which mostly leads to false positive signal because the same signal can be received at different instances such phenomena is also called multiple path propagation [8].



a. Effect of artificial light b. Turbidity of the water

Figure 4. Example of limitation of underwater environment

Hydroacoustic sensing

Typical sensors that are used for sea mapping and geological surveys are based on data coming from SONAR which helps in to generate maps for seafloor, whereas some of teh common one are single beam and multi beam echo sounder. Single beam echosounder generally use piezo electric crystal like quartz for the transmission and receiving of the acoustic signals. It working mechanism is that it calculate the time taken by the signal in transmission and reception as shown by the fig. 5. Since speed of light denoted by c is constant the traveled distance is calculated as d = c x t/2.

The pros of this sensor is that it is fairly simple in operation and is low cost but con is that it cover a small portion of the seabed to be specific that is around 2 to 12 degrees.

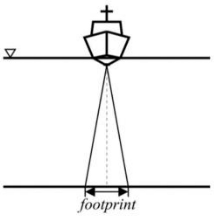


Figure 5. Single Beam Echo Sounder (SBES) source [9]).

Multi beam echosounder are advanced underwater mapping systems used to explore and understand the seafloor. As the name suggests, MBES operates by emitting multiple acoustic beams (sound waves) as shown by fig. 6 simultaneously. These systems consist of an array of transducers that send out sound waves into the water. When these waves encounter the seabed or other underwater features, they bounce back as echoes. MBES captures these echoes and processes them to create detailed bathymetric maps. Bathymetry refers to measuring water depth and understanding the underwater topography. MBES sensors can be categorized as narrow or wide systems. In narrow systems, high spatial resolution is achievable, but the coverage area per transmission is limited. By surveying the seafloor in swaths, MBES provides essential data for creating terrain models and studying underwater landscapes. Overall, MBES plays a crucial role in marine research, navigation, and environmental monitoring [9].

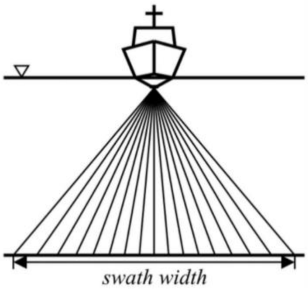


Figure 6. Multi Sound Echo Sounder (MBES) source [9]

Backscattering refers to the strength of the echo that bounces back from the seafloor or other underwater surfaces. When sound waves (acoustic signals) are emitted, they hit the seabed and reflect back. Different types of sediments and seabed structures cause varying intensities of backscattering. It is a useful tool for understanding what’s beneath the water. By analyzing the intensity of echoes, scientists can classify the seafloor based on its composition. To generate acoustic images, the backscatter intensity is converted into grayscale pixel values. This creates a visual representation called the backscatter mosaic. The strength of backscattering depends on several factors, including the angle at which sound waves hit the seafloor, the roughness of the detected object or material, and the properties of the water (such as salinity and temperature). While side-scan sonar provides high-resolution images of the seafloor, backscattering gives valuable information about what makes up the seabed based on echo intensity.

Follwing table 1. Represents data obtained from backcattering and bathymetry data. Bathymetry data which involves measuring water depth and underwater topography plays a critical role in understanding marine ecosystems. We have two specifive bathymetric derivates which are Bathymetric Position Index (BPI) and rugosity. BPI provides insights into benthic ecosystem processes, such as shelter availability and species abundance. Essentially, it helps us understand the vertical position of seafloor features. Rugosity, on the other hand, describes the roughness or irregularity of the seafloor. Areas with high rugosity may offer more habitats for marine organisms. Additionally, backscatter statistics are used to classify seabed sediments based on their textural properties. By analyzing the intensity of echoes (backscatter), researchers gain valuable information about the composition of the seabed.

Table 1. Statistics data of backscatter and bathymetrics derivate source [10]

	Statistics data of Backscatter
mode	Most frequently occurring value within a spatial neighborhood of cells
Median	Median backscatter value within a spatial neighborhood of cells
Mean	Average backscatter value within a spatial neighborhood of cells
Standard deviation	Dispersion from the mean backscatter value of the spatial neighborhood
	Bathymetric derivates
Rugosity	A measure of the irregularity of a surface
BPI	The relative topographic position of a central grid cell to a spatial neighborhood defined by a rectangular or circular annulus with an inner and outer radius
Profile curvature	Curvature parallel to the direction of maximum slope
slope	A measure of the rate of bathymetry change along a path
Planar curvature	Curvature perpendicular to the direction of maximum slope
Standard deviation	Dispersion from the mean depth of the spatial neighborhood

Literature Review a related work

With regard to deep leanrning and machine learning various novels are publishied by the authors which were deeply investigated in the perspective of convolution neural network Luo et al appraoches the convolution neural network for the classificatifcation of the sediment having the data collected from the small side sonar implemented in pearl river estuary in china.using the deep learning technique for the underwater sediment they employed two specific convolutional neural networks (CNNs that is a modified shallow CNN pre-trained on LeNet-5 and a deep CNN based on modifications to AlexNet. The dataset consisted of three sediment classes: reefs, sand waves, and mud. To facilitate classification, the acoustic images were divided into smaller-scale images. Specifically, there were 234 reef images, 217 mud images, and 228 sand wave images. For testing purposes, 20% of each sediment type's images were randomly selected as test samples (46 reef images, 43 mud images, and 45 sand wave images). The results indicated that the shallow CNN outperformed the deep CNN, achieving accuracy rates of 87.54%, 90.56%, and 93.43% for reefs, mud, and sand waves, respectively, on the testing dataset [11].

Subsequently, another study by Qin et al. [12] focused on classifying seabed sediments using common CNN architectures such as AlexNet-BN, LeNet-BN, and VGG16. The model was initially pre-trained on the grayscale CIFAR-10 dataset, which contained 10 classes of images unrelated to sonar data. To address the challenge of small datasets, the authors applied data enhancement techniques based on Generative Adversarial Networks (GANs). This approach expanded the dataset effectively. Additionally, they incorporated conditional batch normalization and modified the loss function to enhance GAN stability. Comparing their proposal with the SVM classifier, the results demonstrated that data optimization techniques could mitigate the limitations of small datasets in sediment classification. However, it's worth noting that using GANs can be computationally expensive. Thus it is viable to fine tune CNNs whilst working with limited data.

Whereas Aleem et al [13] worked on the Faster R-CNN to classify object underwater based on acoustic data Their work involved a forward-looking sonar dataset captured in a lab water tank. To enhance model performance, they introduced noise removal techniques such as the median filter and histogram equalization. Additionally, they combined Faster R-CNN with ResNet 50 for feature extraction. Another approach involved the use of YOLOv5s, a one-stage model, for real-time object detection in side-scan sonar images. The authors down-sampled image echoes while maintaining aspect ratios to improve recognition accuracy and reduce computational costs. Transfer learning played a crucial role by pre-training the model on the COCO dataset. Similarly, Zhang et al. proposed YOLOv5s pre-trained on the COCO dataset for target detection in a forward-looking sonar dataset. They fine-tuned the network using the k-means clustering algorithm. These studies highlight the importance of deep learning techniques in underwater object classification, emphasizing the need for careful preprocessing and model optimization. In the context of R-CNN Fulton et al. come up with the Trash-ICRA19, a dataset from marine litter to detect objects in the sea of japan, around 5700 images were collected for the process of detection and model training by dividing dataset into three categories that is rov (representing man made object), bio (including fish, plants etc) and plastc (such as marine debris), to evaluate such data Fulton used YOLOv2, tiny YOLOv2, SSSD and Faster R-CNN algorithms. These models were executed on an NVIDIA Jetson TX2 platform. The results suggest that these architectures have the potential to detect marine litter on the seafloor in real time. The evaluation metrics used were Mean Average Precision (mAP) and Intersection Over Union (IoU). Specifically, for litter detection whereas as YOLOv2 Achieved an mAP of 47.9% and an IoU of 54.7%, Tiny YOLOv2 achieved an mAP of 31.6% and an IoU of 49.8%, SSD Achieved an mAP of 67.4% and an IoU of 60.6% and Faster R-CNN achieved an mAP of 81.0% and an IoU of 53%. Additionally, a newer version of the dataset called TrashCan was introduced by Hong et al. This updated dataset is more diverse, containing additional images and classes. For baseline comparison, the Mask R-CNN and Faster R-CNN models were utilized.

Subsequently Tim et al. [14] approach to seabed sediment classification using side-scan sonar data through the application of Convolutional Neural Networks (CNNs). Recognizing the critical role of spatially high-resolution seabed information for oceanic and coastal engineering, habitat mapping, and other marine applications, the study addresses the need for efficient and accurate sediment analysis. Traditional methods, reliant on grain-size distribution from collected sediment samples, are enhanced by the CNN model, which automates the classification process. This model represents a significant leap from manual, labor-intensive methods, offering an end-to-end training capability that autonomously derives features during the training phase for subsequent classification. Their research evaluates the model's performance on real-world data, that were labeled into four distinct classes such as fine, sand, coarse, and mixed sediment as shown by the Fig. 7. The model then employs a patch-wise classification strategy coupled with ensemble voting to refine its accuracy.

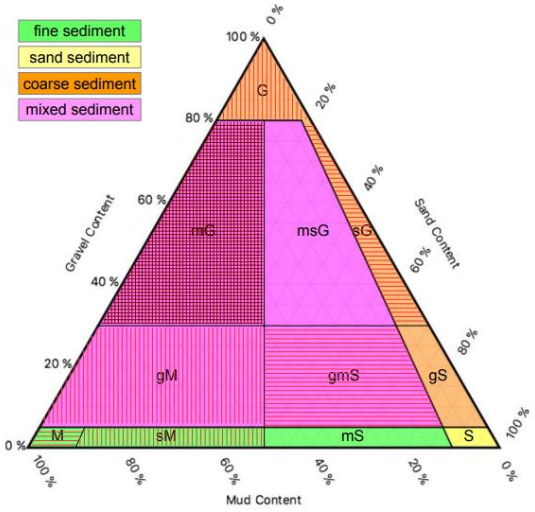


Figure 7. Classification of sediment source [14]

Notably, while the model achieves around 83 percent accuracy in sand sediment prediction, it encounters challenges with fine sediment, where accuracy drops to 11 percent. Their research work also delves into the complexities of label consistency and the factors influencing backscatter intensities, which are crucial for the sonar data interpretation. Where the author suggested future research directions, such as exploring varying levels of discretization and training data strategies to enhance the model's accuracy. It concludes by affirming the transformative potential of CNNs in seabed sediment classification, while also acknowledging the necessity for ongoing research to surmount the existing limitations.

Methodology

Numerous research work and technique were thoroughly studied in the literature review to carry out the analysis process every method have their pros and cons the manual methods were work heavy and uneconomical to run for the long run contrary to that the traditional approach to establishing a relationship model between sonar images and substrate types involves two key stages: sonar image feature extraction and classifier training. In the initial phase, due to limited sonar data availability, unsupervised learning methods are often used for data classification. Clustering algorithms, such as k-means, have been applied to submarine geological classification. However, the sensitivity of clustering methods to initial values can impact classification accuracy. Researchers have also explored Self-Organizing Maps (SOM) for sparse data classification. To achieve clear substrate types, supervised learning methods were utilized. Support Vector Machines (SVMs) and neural networks (such as Backpropagation networks) have been used to improve classification accuracy. Despite ongoing optimization efforts, challenges remain due to computational complexity and low-resolution, noisy sonar images. But with the rise in deep learning models such as convolutional neural networks and taking ideas from the traditional machine learning model a modified version of convolution neural network will be used in this proposed method for the accurate design of seabed substrate detection and classification. The modified CNN model which is used here is called ShuffleNet-DSE which is an extremely efficient convolutional neural network (CNN) architecture designed specifically for mobile devices with very limited computing power. The goal of ShuffleNet-DSE is to achieve high accuracy while operating within tight computation budgets (e.g., 10-150 MFLOPs) commonly found in mobile platforms such as drones, robots, and smartphones [15]. The architecture of ShuffleNet-DSE utilizes two key techniques that is Pointwise Group Convolution To reduce computation complexity and Channel Shuffle Operation: To address the side effects introduced by group convolutions, ShuffleNet-DSE introduces a novel operation called channel shuffle. This operation allows information to flow across feature channels effectively, thus enhancing the overall performance.

In the proposed method ShuffleNet-DSE is categorized into two stages that is Feature pre-extraction and channel information enhancement, the prior is going to improve the poor classification process whereas the later will enhance feature interaction, promote better representation learning which will contribute to the overall efficiency of the network as shown by the fig. 8 below.

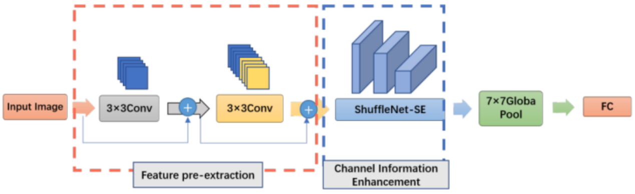


Figure 8. ShuffleNet-DSE network architecture

Extracting Features

Sonar data often contains noise, and its image resolution is poor, leading to challenges in feature extraction. Additionally, the simple data structure of sonar images isn't well-suited for complex network architectures. To address this, maximizing information extraction from each layer while maintaining depth becomes crucial. Dense connections and Swish activation functions can help mitigate these issues (as explained below), enhancing the effectiveness of sonar image analysis.

Wi = Hi([x0, x1, . . . , xi-1])

In a dense connection, the output of each layer (denoted as Wi as shown by the formula below) depends on the output of all previous layers. Specifically, Wi is formed by superimposing the feature maps from all preceding layers. By doing so, the model can fully utilize information from all feature maps during learning without significantly increasing the network depth. This method enhances the model’s learning capacity.

In scenarios with uneven quality of sonar images, traditional connections may blur feature information in certain layers, leading to obscured data for subsequent computations. However, dense connections address this issue. They stack the output feature matrices of each layer with those of all previous layers, ensuring that each layer considers the results from its predecessors. As a result, the risk of losing critical feature information due to specific layers is reduced. Dense connections improve model stability and, importantly, mitigate the vanishing gradient problem during backpropagation. Additionally, the Swish activation function replaces ReLU, providing a smoother transition between linearity and non-linearity. Swish is particularly well-suited for sonar image classification, overcoming ReLU’s limitations while maintaining model stability [16].

Structure of the network

The pre-extraction part of the network structure as illustrated in fig. 9 involves several steps. First, we control the number of channels using a 1x1 convolution. After calculations, the output feature matrix has twice the initial number of channels (e.g., if k = 32, we get 64 channels). Next, we apply a 3x3 convolution kernel for feature computation, ensuring effective information extraction. Superimposing the output feature matrix with the input feature matrix refines the image features. Subsequently the process is repeated to further enhance the data. Moreover after two superimpositions, we normalize the feature matrix. To handle large data dimensions resulting from overlapping feature matrices, we use a 1x1 convolution layer for dimensionality reduction. Finally, a 2x2 max-pooling layer compresses the data, retaining texture information while minimizing parameter errors from previous layers

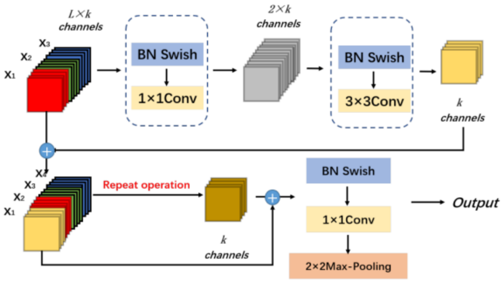


Figure 9. pre-extraction network framework

Next is to enhance the channel information which initiates from the calculation of the data coming from the sonar images, three techniques are utilized, that is DWConv, SE module to process the data and channel shuffle. These techniques greatly improve the accuracy as well as the stability of the model.

Network Framework

The channel information enhancement part in the context of ShuffleNet-V2. This technique builds upon improvements from its predecessor. The network structure is divided into two parts that is A and B as illustrated in Fig 10. the core process is the channel splitting process. Meaning channels are divided into two equal groups. Unlike ResNet’s addition operation, ShuffleNet-V2 uses concatenation (Concat). By splicing the left and right channel parts together, we ensure that the number of input channels matches the output channels. But the main work happens through the Channel Shuffle operation. This operation allows information exchange between the two channel groups. By shuffling channels, we enhance feature representation and improve overall performance. In essence, ShuffleNet-V2 leverages channel splitting, concatenation, and channel shuffling to enrich feature information and optimize channel communication.

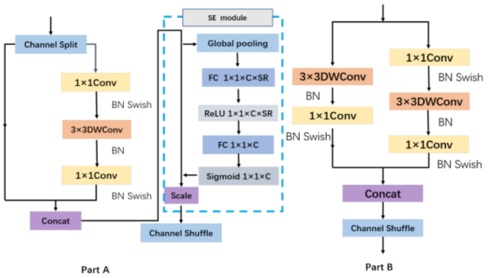


Figure 10. Channel information enhancement framework

Whereas the network level configuration is shown by the following table 2.

Table 2. ShuffleNet-DSE network design parameters

Module	Layer	K-Size	Output
input			1
	Conv1	3x3	32
	BN + Swish		
	Conv2	1x1	64
	BN+Swish		
	Conv3	3x3	32
	BN+Swish		
	Conv4	1x1	64
	BN+Swish		
	Conv5	1x1	64
	BN+Swish		
Dense Connection	Conv6	3x3	32
	BN+Swish		
	Conv7	1x1	48
	MaxPool	2x2	48

	A block		
	B block		48
	A block		
	B block		96
ShuffleNet-SE	A block		
	B block		192
	Conv5	1x1	1024
	Global Pool	7x7	
Summarize	FC		

Datasets for the model

The dataset utilised for the model training and analysis was comprised of the data coming from public sonar image dataset (US geological survey) and the Welhai real sonar dataset (WHDS). The public data was divided into six categories, having images of equal number of pixels(200x200), and to compensate for image scarcity, data augmentation techniques were used for dataset expansion. After random mixing, cropping and expanding a total of 2160 images were obtained. The dataset images are clear, and the features relevant for image classification are evident, making classification easier.

Contrary to that the WHDS dataset consists of real Weihai data. Compared to the SAS dataset, WHDS images exhibit fuzzier features, but they are more aligned to represent practical application scenarios. WHDS poses greater classification difficulty, making it the primary focus of the experiment. The images were cropped to a size of 200x200 pixels, resulting in a total of 516 pictures. The pictures were labelled and organised into three categories following 3096 images for random mixing, cropping and expanding. Further explanations of the data are shown by table 3 and real pic illustration. At the time of testing the data the accuracy slightly differed for each training recognition to rectify this issue the process was repeated numerously after that average was taken to evaluate data.

Table 3. Dataset information

Dataset	Category	Quantity	Enhancement
	Ripple Vertical	60	360
	Ripple 45°	60	360
SAS	Sand	60	360
	Rock	60	360
	Posidonia	60	360
	Silt	60	360
	1	203	1218
WHDS	2	153	918
	3	160	960

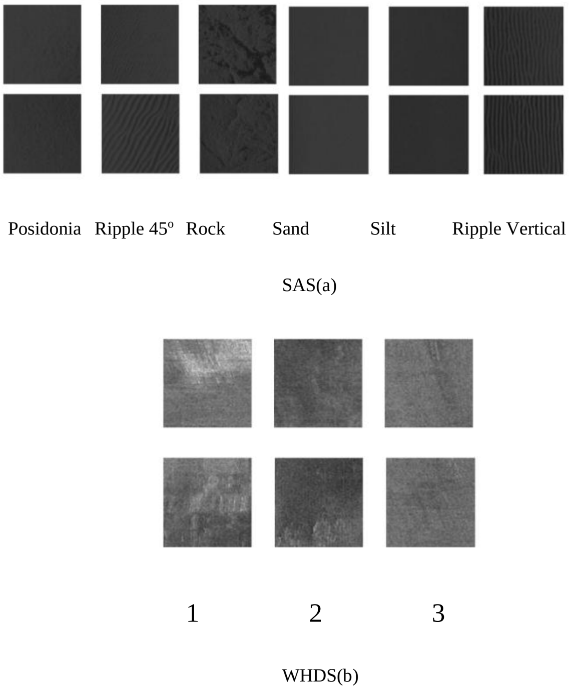


Figure 11. Illustration detail SAS a. WHDS b.

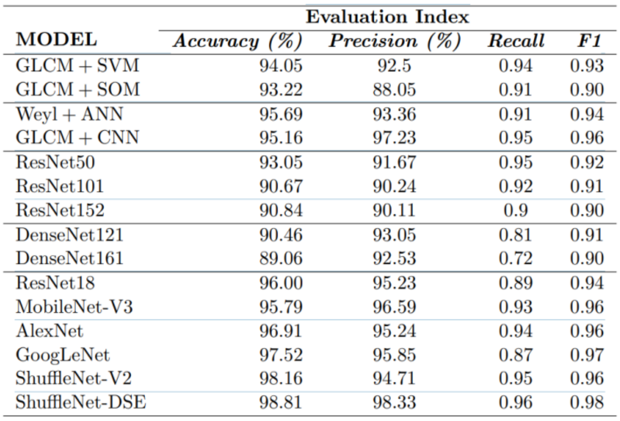
Performance metrics

The evaluation metrics for a classification model using a confusion matrix In the context of assessing model performance, we rely on several key metrics derived from the confusion matrix. These metrics help us understand how well our model is doing such as **Accuracy** provides an overall measure of correctness that is it calculates the ratio of correctly predicted instances (True Positives and True Negatives) to the total number of instances by the formula **Accuracy = (TP + TN) / m**, where TP represents True Positives, TN represents True Negatives, and m is the total number of instances. Next we have is **Precision**, it focuses on positive predictions. Where It quantifies the proportion of true positive predictions among all positive predictions made by the model via the formula **Precision = TP / (TP + FP)**, where FP represents False Positives. After the precision we have recall, F1 Score and inference time (T) in which recall assess how well the model identifies actual positive instances. It calculates the ratio of True Positives to the sum of True Positives and False Negatives by the formula **Recall = TP / (TP + FN)**, where FN represents False Negatives, and The **F1 score** balances precision and recall. It combines both metrics into a single value, especially useful for imbalanced datasets as shown by the equation **F1 = 2 * (Precision * Recall) / (Precision + Recall)** lastly **Inference Time (T)** measures how long the model takes to make predictions. It serves as an indicator of model complexity and efficiency. That is, lower inference time is desirable for real-time applications. In essence, these evaluation metrics help us gauge classification model performance, considering both accuracy and computational efficiency.

Result

The performance data of various models were evaluated using the SAS dataset, as depicted in Table 4. The dataset, which is a professional image with distinct texture attributes , allowed all models to maintain an accuracy of approximately 95%. However, the F1 score revealed that the comprehensive model and the deep learning model had better stability. From deep learning perspective, it was observed that the ResNet and DenseNet series, which have deep structures, struggled to deliver excellent performance. As the complexity of these models increased, their classification accuracy gradually declined. This was attributed to noticeable overfitting during the training process. Despite achieving 100% accuracy on the training set, their performance on the test set was unsatisfactory. Although the differences between various models on this dataset were minimal, the accuracy and other indicators showed that the deep learning network with lower complexity performed better. Among the models compared, ShuffleNet-DSE stood out with an accuracy of 98.81% and a precision of 98.33%. Its accuracy was 0.65% higher than that of ShuffleNet-V2, and its precision was significantly higher than other deep learning models. Moreover, ShuffleNet-DSE had the highest F1 score of 0.98 among the compared models. After considering all factors, ShuffleNet-DSE emerged as the model with the best performance. However, the uniqueness of the SAS data does not fully represent the superiority of the classification model's performance, indicating the need for further experimentation using the WHDS dataset.

Table 4. Comparision model from SAS dataset.



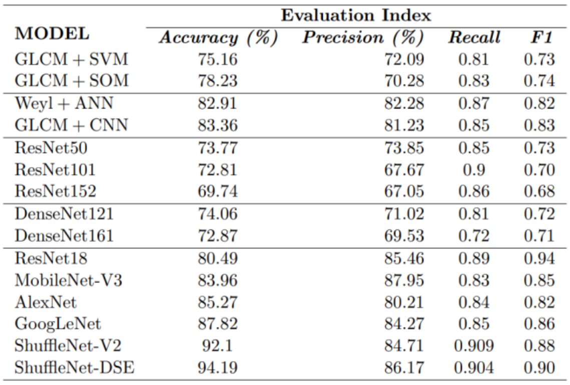
MODEL	Evaluation Index			
	Accuracy (%)	*Precision (%)*	*Recall*	*F1*
GLCM + SVM	94.05	92.5	0.94	0.93
GLCM + SOM	93.22	88.05	0.91	0.90
Weyl + ANN	95.69	93.36	0.91	0.94
GLCM + CNN	95.16	97.23	0.95	0.96
ResNet50	93.05	91.67	0.95	0.92
ResNet101	90.67	90.24	0.92	0.91
ResNet152	90.84	90.11	0.9	0.90
DenseNet121	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

WHDS data, a type of conventional sonar image data, is influenced by noise and has somewhat unclear texture features. After being manually labeled, it's divided into three categories. This data provides a real-world test of a model's performance. There's a significant difference in performance between various models on the WHDS dataset. The performance of each model is detailed in Table 5. Traditional machine learning struggles to adapt to sonar image data, with the accuracy of two models struggling to reach 80% and averaging around 72%, which is considered poor compared to all comparison models. In most public datasets, the ResNet and DenseNet series outperform the MobileNet and ShuffleNet series of lightweight networks. However, this trend is reversed with sonar imagery. Unlike other image types, the classification accuracy of deep learning networks decreases as the network's depth increases. For example, the accuracy of ResNet drops from 73.77% to 69.74% when the depth is increased from 50 to 152. Similarly, DenseNet's accuracy reduces from 74.06% to 72.87% when the depth is changed from 121 to 161. The deep network structures of the ResNet and DenseNet series tend to overfit when using sonar images as datasets, leading to a mismatch between the overall model performance and the model complexity.

Ignoring the model's inference time, the performance of the comprehensive model is superior and can match most lightweight deep learning models, with an accuracy of over 80%. However, the computational complexity of the synthetic model is much higher 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 highest accuracy, 94.19% and 92.1% respectively. Both these models use the techniques of DWConv and Channel-Shuffle, demonstrating the effectiveness of these techniques in processing sonar image data. The MobileNet-V3, which has SE modules in its network structure, has excellent precision, which is 3.24% higher than that of ShuffleNet-V2. The ShuffleNet-DSE model, which also has the SE module, successfully compensates for the poor precision of ShuffleNet-V2, improving the precision from 84.71% to 86.17%. The SE module can enhance the precision of the model.

The ShuffleNet-DSE model shows improvements in terms of accuracy and precision. Compared to the original model, the accuracy of ShuffleNet-DSE improves from 92.10% of ShuffleNet-V2 to 94.19%. Compared to the comprehensive learning model Weyl + ANN and GLCM + CNN, the accuracy improves by 11.28% and 10.83%, respectively. Considering the experimental results from two different datasets, the ShuffleNet-DSE demonstrates good accuracy and stability in processing sonar image data. 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 remains relatively low. The ShuffleNet-DSE has excellent classification accuracy while ensuring low model complexity.

Table 5. Performance of the model based WHDS data



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

Parameters for model training

The primary training parameters of the main models being compared are outlined here where the Momentum-Optimizer, with a momentum of 0.9, is predominantly used. The regular term primarily employs L2 with a coefficient of 0.00001. The learning rate for ResNet18 and AlexNet is adjusted using Piecewise Decay, and the learning rate changes at training rounds 30, 60, and 90.

For the ResNet18 model, the optimizer used is Momentum, and the learning rate is adjusted using Piecewise Decay. The learning rate values change at epochs 30, 60, and 90, with values of 0.1, 0.01, 0.001, and 0.0001 respectively. The regularizer used is L2. Subsequently the MobileNet-V3 model uses the Momentum optimizer with a learning rate of 1.3 and the L2 regularizer. Following by the AlexNet model uses the Momentum optimizer and the learning rate is adjusted using Piecewise Decay. The learning rate values change at epochs 30, 60, and 90, with values of 0.01, 0.001, 0.0001, and 0.00001 respectively. The regularizer used is L2. Moreover the GoogLeNet model uses the Momentum optimizer with a learning rate of 0.001 and the L2 regularizer. Both the ShuffleNet-V2 and ShuffleNet-DSE models use the Momentum optimizer with a learning rate of 0.0125 and the L2 regularizer.

Parameters for model training

This research work presents a solution to the challenge of accurately classifying seabed sediment in sonar images by designing a deep learning-based model specifically for this task. The model, known as ShuffleNet-DSE, is an enhancement of the ShuffleNet-V2 structure, incorporating the SE module, Swish activation function, and dense connection module. Despite a slight compromise on model efficiency, the ShuffleNet-DSE significantly boosts the accuracy and stability of seabed sediment image classification.

The model's effectiveness was demonstrated through experiments on the public SAS dataset and the Weihai dataset collected at sea, where it showed promising performance. However, the limited quantity of sample image data does impact the experimental results. Even though the dataset was expanded through data augmentation, the actual number of seabed sediment images remained the same, limiting the broad applicability of the experimental conclusions.

Looking ahead, the plan is to use a larger amount of real seabed sediment image data in future experiments to further refine the model structure. This will help to enhance the model's performance and applicability, making it a more robust tool for seabed sediment classification in sonar images. Which has the potential to greatly assist in underwater exploration, research and detection.

Acknowledgment

This work was supported by the Scientific Research Fund of Zhejiang Provincial Education Department (Y202248619).

References

[1] Boomsma, W.; Warnaars, J. Blue mining. In Proceedings of the 2015 IEEE Underwater Technology (UT), Chennai, India, 23–25 February 2015; pp. 1–4

[2] Cong, Y.; Gu, C.; Zhang, T.; Gao, Y. Underwater robot sensing technology: A survey. Fundam. Res. 2021, 1, 337–345.

[3] Hein, J.R.; Mizell, K. Deep-Ocean Polymetallic Nodules and Cobalt-Rich Ferromanganese Crusts in the Global Ocean: New Sources for Critical Metals. In Proceedings of the United Nations Convention on the Law of the Sea, Part XI Regime and the International Seabed Authority: A Twenty-Five Year Journey; Brill Nijhoff: Boston, MA, USA, 2022; pp. 177–197.

[4] de Oliveira Soares, M.; Matos, E.; Lucas, C.; Rizzo, L.; Allcock, L.; Rossi, S. Microplastics in corals: An emergent threat. Mar. Pollut. Bull. 2020, 161, 111810.

[5] A High-Accuracy Model Average Ensemble of Convolutional Neural Networks for Classification of Cloud Image Patches on Small Datasets - Scientific Figure on ResearchGate. Available from: https://www.researchgate.net/figure/Schematic-diagram-of-a-basic-convolutional-neural-network-CNN-architecture-26_fig1_336805909 [accessed 21 Apr, 2024]

[6] Domingos, L.C.; Santos, P.E.; Skelton, P.S.; Brinkworth, R.S.; Sammut, K. A survey of underwater acoustic data classification methods using deep learning for shoreline surveillance. Sensors 2022, 22, 2181.

[7] Hashisho, Y.; Albadawi, M.; Krause, T.; von Lukas, U.F. Underwater color restoration using u-net denoising autoencoder. In Proceedings of the 2019 11th International Symposium on Image and Signal Processing and Analysis (ISPA), Dubrovnik, Croatia, 23–25 September 2019; pp. 117–122.

[8] Gonçalves, L.C.d.C.V. Underwater Acoustic Communication System: Performance Evaluation of Digital Modulation Techniques. Ph.D. Thesis, Universidade do Minho, Braga, Portugal, 2012.

[9] Grall, P.; Kochanska, I.; Marszal, J. Direction-of-arrival estimation methods in interferometric echo sounding. Sensors 2020, 20, 3556.

[10] Stephens, D.; Diesing, M. A comparison of supervised classification methods for the prediction of substrate type using multibeam acoustic and legacy grain-size data. PLoS ONE 2014, 9, e93950.

[11] Luo, X.; Qin, X.; Wu, Z.; Yang, F.; Wang, M.; Shang, J. Sediment classification of small-size seabed acoustic images using convolutional neural networks. IEEE Access 2019, 7, 98331–98339.

[12] Qin, X.; Luo, X.; Wu, Z.; Shang, J. Optimizing the sediment classification of small side-scan sonar images based on deep learning. IEEE Access 2021, 9, 29416–29428.

[13] Aleem, A.; Tehsin, S.; Kausar, S.; Jameel, A. Target Classification of Marine Debris Using Deep Learning. Intell. Autom. Soft Comput. 2022, 32, 73–85.

[14] Berthold, Tim & Leichter, Artem & Rosenhahn, Bodo & Berkhahn, Volker & Valerius, Jennifer. (2017). Seabed sediment classification of side-scan sonar data using convolutional neural networks. 1-8. 10.1109/SSCI.2017.8285220. https://arxiv.org/pdf/1707.01083.pdf

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