IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received January 21, 2020, accepted February 29, 2020, date of publication March 6, 2020, date of current version March 18, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2978880

Underwater Object Classification in Sidescan Sonar Images Using Deep Transfer Learning and Semisynthetic Training Data

GUANYING HUO[1,2], (Member, IEEE), ZIYIN WU[1], AND JIABIAO LI[1]

[1]Key Laboratory of Submarine Geosciences, Second Institute of Oceanography, Ministry of Natural Resources, Hangzhou 310012, China
[2]College of IoT Engineering, Hohai University, Changzhou 213022, China

Corresponding authors: Guanying Huo (huoguanying@hhu.edu.cn), Ziyin Wu (zywu@vip.163.com), and Jiabiao Li (jbli@sio.org.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 41830540, Grant 41876097, and Grant 41576099, and in part by the Scientific Research Fund of the Second Institute of Oceanography, Ministry of Natural Resources (MNR), under Grant JB1803.

ABSTRACT Sidescan sonars are increasingly used in underwater search and rescue for drowning victims, wrecks and airplanes. Automatic object classification or detection methods can help a lot in case of long searches, where sonar operators may feel exhausted and therefore miss the possible object. However, most of the existing underwater object detection methods for sidescan sonar images are aimed at detecting mine-like objects, ignoring the classification of civilian objects, mainly due to lack of dataset. So, in this study, we focus on the multi-class classification of drowning victim, wreck, airplane, mine and seafloor in sonar images. Firstly, through a long-term accumulation, we built a real sidescan sonar image dataset named *SeabedObjects-KLSG*, which currently contains 385 wreck, 36 drowning victim, 62 airplane, 129 mine and 578 seafloor images. Secondly, considering the real dataset is imbalanced, we proposed a semisynthetic data generation method for producing sonar images of airplanes and drowning victims, which uses optical images as input, and combines image segmentation with intensity distribution simulation of different regions. Finally, we demonstrate that by transferring a pre-trained deep convolutional neural network (CNN), e.g. VGG19, and fine-tuning the deep CNN using 70% of the real dataset and the semisynthetic data for training, the overall accuracy on the remaining 30% of the real dataset can be eventually improved to 97.76%, which is the highest among all the methods. Our work indicates that the combination of semisynthetic data generation and deep transfer learning is an effective way to improve the accuracy of underwater object classification.

INDEX TERMS Object classification, underwater search and rescue, sidescan sonar image, semisynthetic data generation, deep transfer learning.

I. INTRODUCTION

Sidescan sonars can provide high resolution images of the seabed even in zero-visibility water, which makes it very useful in a variety of military and civilian applications such as mine-countermeasures, ocean mapping, offshore oil prospecting, and underwater search and rescue [1]–[3]. For underwater search and rescue, sidescan sonars have been widely used to detect drowning victims, wrecks and airplanes lying on the seabed or lakebed.

For long searches, such as the search for Malaysian Airlines Flight 370, the sonar operators, who stare at the display screen to see if there is the desired object, may feel very tired after a period of work and miss potential target object. Therefore, it is essential to use some intelligent image processing methods to help people find possible objects and correctly identify the desired objects. Sidescan sonar image segmentation methods [4]–[10], which can distinguish the highlight object region, the accompanied shadow region and the remaining sea-bottom reverberation region, can be used to mark all the suspicious objects and alert the operator. Although some of the objects marked by the segmentation methods are of interest to people, most are meaningless for a special search task. Therefore, object classification or recognition is another important issue for underwater search.

Currently, underwater object classification or recognition methods are mainly focusing on mine classification, i.e., detecting all possible mine-like objects (MLOs)

The associate editor coordinating the review of this manuscript and approving it for publication was Zhan-Li Sun.

and classifying each of the objects as mine or not-mine [11]. Before the rise of deep learning [12], many model-based approaches using a prior knowledge or data-driven approaches have been proposed for identifying MLOs. By comparing an extracted set of features from an MLO to a set of training data, some approaches can work well when the test data are very similar to the training dataset [13]–[16]. Using the information from both highlight region shadow regions or multi-view templates can help to improve the classification accuracy [16]. Reed *et al.* [2] combined the Hausdorff distance of the synthetic shadows to the real object shadow with highlight and size information to produce a membership function, and then adopted Dempster–Shafer information theory to classify the objects using both mono-view and multi-view analysis. Cho *et al.* [17] tried to improve the recognition accuracy by using multi-angle view mine simulation and template matching. Away from model-based approaches, local feature descriptors without prior knowledge, such as the Haar-like feature [18], the Haar-like and local binary pattern (LBP) features [3], the combination of Haar features and learned features from a human operator's brain electroencephalogram (EEG) [19] have also been proposed for mine recognition. The extracted features are usually combined with some state-of-the-art machine learning approaches, such as boosting [18] and support vector machines (SVMs) [20]. The features must be carefully selected, and the classifier should be adjusted when the training data is imbalanced [21].

Deep learning, which uses neural networks (NNs) involving more than two layers, can exploit feature representations learned exclusively from data and therefore provide an end-to-end scheme for a special task instead of hand-crafting features [22]. Since Hinton [23] integrated the restricted Boltzmann machine into a deep neural network, deep learning has seen a wave of success in many fields of application. For instance, by interleaving multiple convolutional and pooling layers, convolutional neural networks (CNNs), have shown great advantages in object classification [24]–[26], object detection [27], [28], and semantic segmentation [29], [30], sometimes even able to surpass human ability. Recently, CNNs have also been applied to MLOs or marine vessels detection [31]–[37] and seafloor classification [38]–[40], and have been proven to be more effective than traditional methods.

The classification methods mentioned above are mainly concerned with MLOs or seafloor classification. However, there is an increasing demand for humanitarian searching and rescuing drowning victims, wrecks and airplanes. Motivated by this, in this paper, we focus on a new classification task of multi-class objects including drowning victims, wrecks, airplanes, mines and seafloor. Because both the wrecks and airplanes have much higher shape complexity than MLOs and seafloor, this classification task is more challenging. It is natural to choose powerful CNNs for this challenging task, but we must first solve the problem of lack of dataset. For object classification of optical images, the ImageNet dataset has played a very important role; for object classification of SAR images, the MSTAR dataset can be used, and a ship dataset has recently been released on Github by Wang *et al.* [41], [42]. However, *no public dataset is available for underwater object classification, which severely restricts the development of underwater object classification*. That is why we first build the real sidescan sonar image dataset named *SeabedObjects-KLSG*, which currently contains 385 wreck, 36 drowning victim, 62 airplane, 129 mine and 578 seafloor images.

Although our dataset contains as many data as possible, it is still a small one. It is believed that training a CNN from scratch usually needs more than 5000 samples per class to achieve a satisfactory result [12]. Transfer learning can play an important role when the training dataset is small. It is to be expected that the most used fine-tuning strategy in transfer learning can perform well on the SeabedObjects-KLSG dataset. Moreover, can we further improve the classify accuracy using some easy-to-implement methods, such as generating semisynthetic data for training?

Based on the above analysis, the main contributions and work of this paper are given as follows.

1) To urgently promote underwater objects classification in sidescan sonar images, especially civilian objects classification, a real sidescan sonar image dataset named *SeabedObjects-KLSG* is built, which can be used to identify wrecks, drowning victims, airplanes, mines and seafloor.
2) Considering the real dataset is imbalanced, we proposed a semisynthetic data generation method for producing sonar images of airplanes and drowning victims, which uses optical images as input, and combines image segmentation with intensity distribution simulation of different regions.
3) By transferring a pre-trained deep convolutional neural network (CNN), e.g. VGG19, and fine-tuning the deep CNN using 70% of the real dataset and the semisynthetic data for training, the overall accuracy on the remaining 30% of the real dataset can be eventually improved to 97.76%, which is higher than fine-tuning using only the real dataset and other three methods.

II. DATASET AND METHODS

A. UNDERWATER OBJECT DATASET OF SIDESCAN SONAR IMAGE

For humanitarian underwater search and rescue, drowning victims, wrecks and airplanes are concerned by people. With the great support from several sonar equipment suppliers including Lcocean, Hydro-tech Marine, Klein Marine, Tritech, and EdgeTech, and through a decade of accumulation, we have eventually built a real object dataset of sidescan sonar images, in which each image consists of a single object, namely drowning victim, wreck, airplane, mine, or pure seafloor. The dataset is named *SeabedObjects-KLSG*, and currently includes 385 wreck images, 36 drowning victim images, 62 airplane images, 129 mine images, and

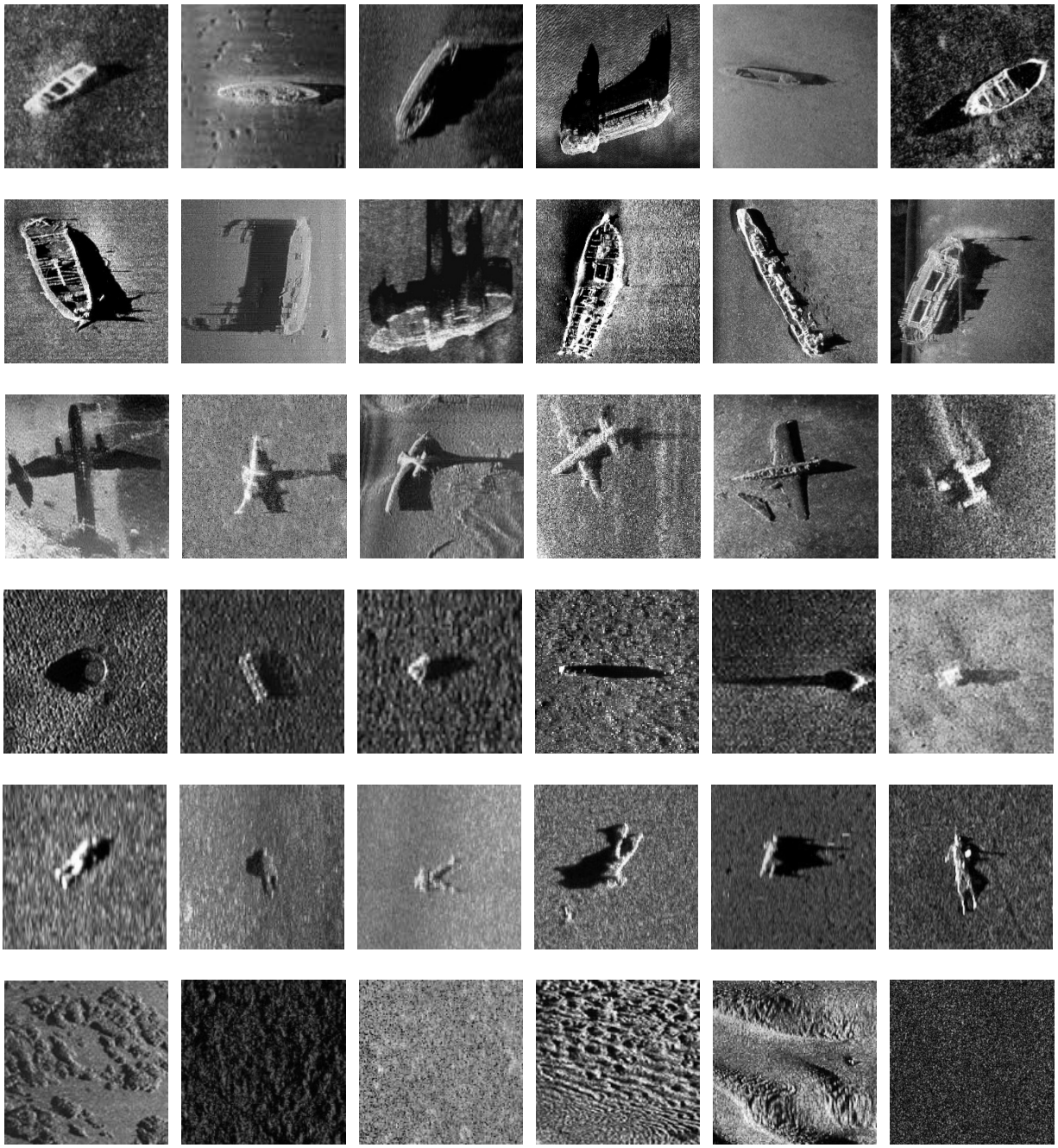

FIGURE 1. Samples from the *SeabedObjects-KLSG* dataset. Samples in the first and second rows, in the third row, in the fourth row, and in the fifth row are wrecks, airplanes, mines, drowning victims, and seafloors, respectively.

578 seafloor images. All the images are cut from original big sidescan sonar images, without any processing.

Some samples in the *SeabedObjects-KLSG* dataset are shown in Fig. 1. It is worth noting that each type of objects has its appearance diversity in Fig. 1, e.g. wrecks of different size and appearance, mines from different manufacturers and with different shapes (the first three are Manta, KMD-II, and Rockan, with the shapes of truncated cone, cylinder, and wedge, respectively), seafloor of different types (the seafloor images from left to right are rock, mud, sand, sand waves, sand ridges, and clay, respectively). Because wrecks and airplanes have more complex appearance, mines have different shapes, and seafloor have different types, the feature representation used for our task must be powerful enough to highly represent each major class. Thrilled by the success of deep learning, it is natural to use CNNs in our task.

B. CNN AND TRANSFER LEARNING

1) PRINCIPLES OF A CNN

CNN is the leading model in image classification, object detection, and semantic segmentation. Since LeNet was proposed by LeCun *et al.* [43], many excellent CNNs

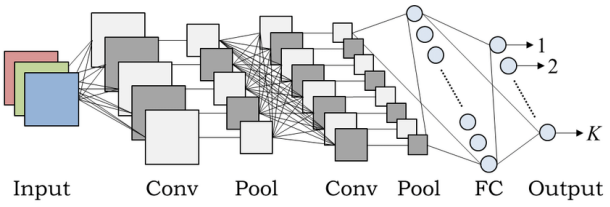

FIGURE 2. Structure of a CNN used for classification of K classes. Conv, Pool, and FC mean convolutional layer, pooling layer, and fully connected layer, respectively.

have been proposed in recent years, such as AlexNet [24], VGGNets [25], GoogLeNet [44], and ResNet [26]. A typical CNN consists of convolutional layers, pooled layers, and fully connected layers. Fig. 2 gives a structure of a shallow CNN, which includes 2 convolutional layers, 2 pooling layers, 2 fully connected layers (output layer is also a fully connected layer).

Convolutional layers have the properties of sparse connectivity, weight sharing and translation equivariance, because each convolution kernel is local and very small, e.g., with a size of 3×3 or 5×5, and is shared across the same input-output connection. The convolutional layers can be calculated according to the following:

$$C_j^l = \sigma(\sum_{i=1}^{M} D_i^{l-1} * K_{ij}^l + b_j^l) \quad j = 1, 2, ...N \qquad (1)$$

where C_j^l is the feature map with index j (totally N feature maps) in the current layer l; D_i^{l-1} is the channel with index i (totally M channels) in the previous layer $l-1$; K_{ij}^l is the convolution kernel used to convolve with D_i^{l-1} to generate C_j^l, and b_j^l is the bias; $*$ denotes the convolution operation; and σ is the activation function.

CNNs usually use the rectified linear unit (ReLU) as the activation function σ, which can strengthen the nonlinear representation of features. The pooling layers, which often use max-pooling, i.e., choosing the maximum value in a local window as the output value, are used to reduce the size of the representation to save the computation as well as to make features a little more robust.

After a series of convolution and pooling operations, multiple feature maps can be extracted from the input image, which are flattened and then input to a fully connected layer. Dropout operation, which makes a neuron ineffective with probability p in each training iteration, is often adopted in the fully connected layers of a CNN to prevent over-fitting. A Softmax is combined with the final output layer to give the probability of the classification results, and the Cross Entropy is usually used as the loss function. For the error backpropagation and more details of CNNs, we invite the reader to consider the book by Goodfellow [12].

Because many parameters need to be learned in a CNN (LeNet, the shallowest one among the above mentioned, has about 60,840 parameters; other CNNs have even more parameters), a small training dataset is far from enough. An empirical rule is that if training a CNN from scratch, 5000 training samples per class is usually necessary to achieve an acceptable result. Obviously, we donnot have that many data.

2) TRANSFER LEARNING

In case of small samples, we can resort to transfer learning. By transferring the knowledge from the source domain to the target domain, transfer learning can help to overcome the difficulty of insufficient data. If a domain is represented by $\mathcal{D} = \{\chi, P(X)\}$, where χ is the feature space and $P(X)$ is the edge probability, and a task is represented by $\mathcal{T} = \{y, f(x)\}$, where y is the label space and $f(x)$ is the target prediction function, the definition of transfer learning can be formally defined as follows [45]:

Given a learning task $\mathcal{T}_t$ on domain $\mathcal{D}_t$, we can get the help from for another learning task $\mathcal{T}_s$ on domain $\mathcal{D}_s$. Transfer learning aims to improve the performance of predictive function f_t for the task $\mathcal{T}_t$ by discovering and transferring knowledge from $\mathcal{D}_s$ and $\mathcal{T}_s$, where $\mathcal{D}_s \neq \mathcal{D}_t$ and/or $\mathcal{T}_s \neq \mathcal{T}_t$.

Considering the relationship between the source domain and the target domain, transfer learning methods can be divided into four major categories [45]: instance-based, feature-based, parameter/model-based, and relation-based. Model-based deep transfer learning [46]–[49] is the most popular one, and fine-tuning pre-trained models learned from large benchmark datasets in source domains, has been proven to be more effective than direct transfer learning [41], [49]. The key to the success of model-based deep transfer learning is that low-level and middle-level features represented by a deep CNN is generic for different tasks [48], [49].

In this paper, to solve the problem of insufficient data, we also use fine-tuning. After pre-training a CNN model on the large dataset of ImageNet, we transfer all the trained layers except the final fully connected layer with Softmax classification and add a new output layer with 5 outputs to construct a new CNN model; then we fine-tune the whole model on the training dataset of *SeabedObjects-KLSG*. After fine-tuning, we can finally test the performance of the fine-tuned CNN. The whole process is shown in Fig. 3. Because VGGNet is more suitable for transfer learning, VGG19 is used in this paper, the structure of which is given in Fig. 4.

3) SEMISYNTHETIC DATA GENERATION

It is very rare that we can obtain sidescan sonar images containing meaningful objects such as drowning victims, airplanes, etc. Therefore, despite our persistent efforts for more than 10 years, the *SeabedObjects-KLSG* dataset, which we believe contains the largest numbers of wrecks, drowning victims and airplanes, is still small and imbalanced. Although transfer learning of a pre-trained CNN using ImageNet can compensate data inadequacy, imbalance of real training data may still cause more misclassification. Considering this, we try to simulate more sonar images of drowning victims, airplanes and mines, and then add these synthetic data to the training dataset, to see if they can help to improve the classification performance of a CNN.

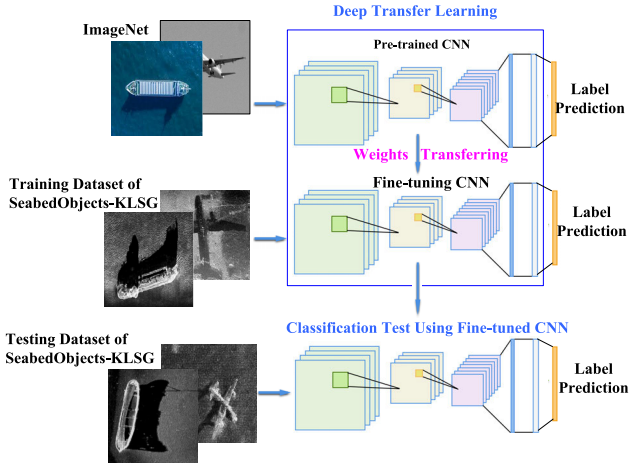

FIGURE 3. Deep transfer learning used in this paper. We first pre-train a CNN model on the dataset of ImageNet, then transfer most layers with weights and fine-tune the pre-trained CNN on the training dataset of *SeabedObjects-KLSG*, and finally test the fine-tuned CNN on the testing dataset of *SeabedObjects-KLSG* after fine-tuning.

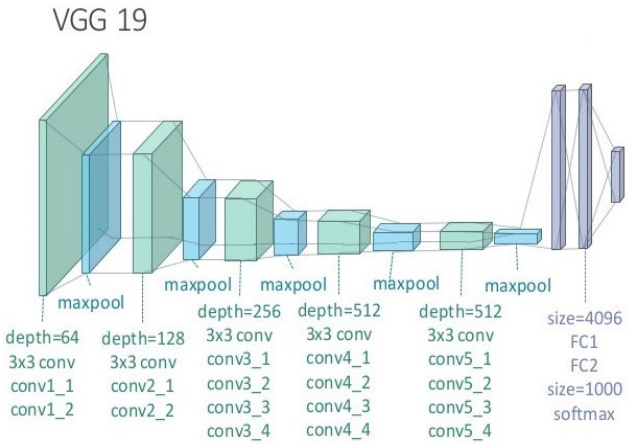

FIGURE 4. Structure of VGG19. VGG19 includes 16 convolutional layers (in five groups) with 3 × 3 kernel, and 3 fully connected layers, totally 19 layers with parameters.

Several excellent sonar image simulation models [50]–[55] have been proposed for producing complete synthetic sonar images, which usually include three-dimensional modeling of objects and seabed, ray tracing, waves scattering calculation using the Kirchhoff approximation, etc. A precise model should also take into consideration transducer directivity and motion characteristics, and we recommend the work by Bell to the reader for sidescan sonar image simulation [50]. It will be difficult and take a long time to implement such a precise simulation model. Considering this, a simple semisynthetic data creation scheme to generate MLO sidescan images has been proposed by Barngrover [3], and proven to be effective.

The above methods, except the method proposed by Barngrover, are all based on simulation, and require complicated calculation. Although the current simulation methods can be well used in the simulation of mine targets with several simple and regular shapes, it is difficult for them to simulate airplanes with different complex shapes. Considering that the major shape information of an object is the most important for underwater object recognition, a semisynthetic data generation method is proposed and adopted here, the processing flow chart of which is given in Fig. 5. Different form the method proposed by Barngrover which moves the mine object in a sonar image to different positions, the proposed method uses an optical image to extract the major shape of an object, such as an airplane, simulate the object and shadow regions according to the probability distribution function matching of a reference sonar image, and randomly select the background from the reference image. Firstly, the input optical object image is segmented and the object is highlighted, then the corresponding shadow region is created by translation. Meanwhile, after the segmentation of a reference sonar image, the intensity distributions of the object and shadow regions of the reference image can be modelled by Weibull probability distribution function (PDF), which is defined by:

$$\mathcal{W}_X(x; min, C, \alpha) = \frac{C}{\alpha}\left(\frac{(x-min)}{\alpha}\right)^{C-1} exp\left(-\frac{(x-min)^C}{\alpha^C}\right) \quad (2)$$

where C and α are the shape and scale parameters, respectively; min is the minimum value of X. Although Rayleigh and other distribution models can be adapted to sonar images, the Weibull PDF has been proven to very effective in describing the intensity x within the shadow and reverberation regions [5].

Intensity distributions of object and shadow regions of the synthetic image are then simulated according to the two Weibull PDFs generated by the object and shadow regions of the reference image, respectively. Finally, the background of the simulated sonar image is randomly selected and copied from the reference image, and the semisynthetic image can be downsampled in the azimuth direction to take into account ship speed. For object segmentation of the object image, many famous segmentation methods can be used, e.g., Chan-Vese model; for segmentation of noisy sonar images, several sonar image segmentation methods can be used, such as [1], [4]–[9] and our previous work in [10].

Some semisynthetic sidescan sonar images of airplanes and drowning victims are given in Fig. 6. The semisynthetic sonar images have the major shapes of airplanes or drowning victims, and look like real sidescan sonar images. For simulating sonar images of mines, the shapes of which are regular and simple, the first two steps in Fig. 5 are replaced by three-dimensional modeling and ray tracing. Some simulated mine images are also given in Fig. 6.

III. EXPERIMENTAL RESULTS

A. EXPERIMENT SETTINGS

To demonstrate the effectiveness of the proposed method using both deep transfer learning and semisynthetic training data, the experimental results of the proposed method will

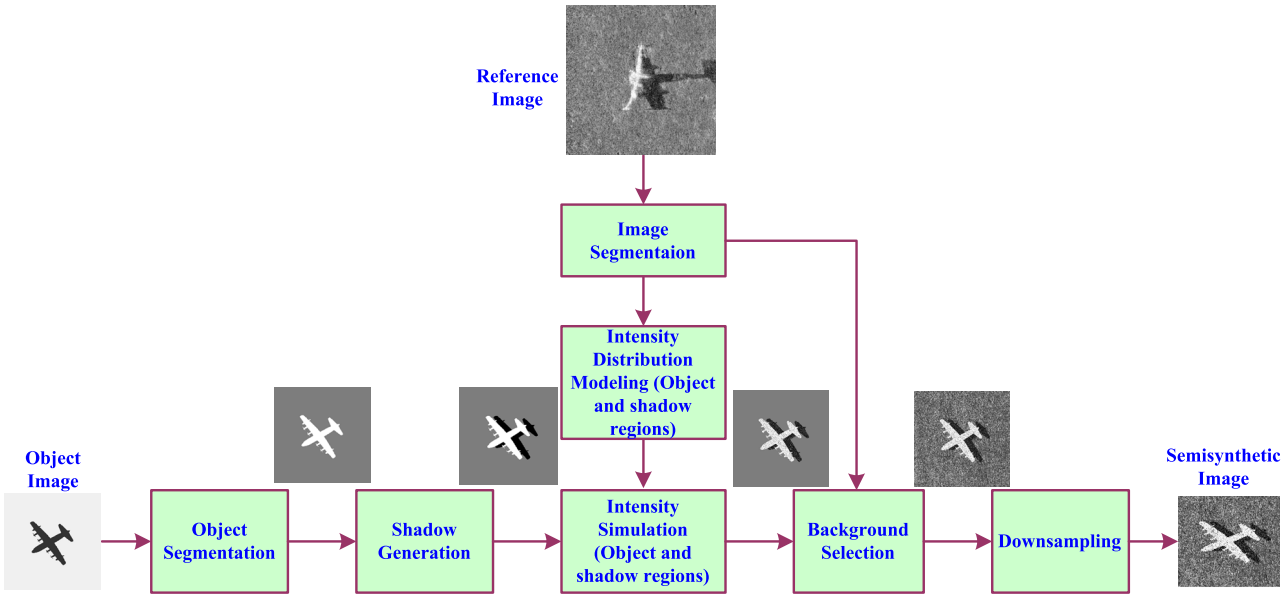

FIGURE 5. Flow chart of the generation of semisynthetic sidescan sonar images. Shape of an object is from an object image, intensity distributions of object and shadow regions are simulated in accordance with those of a reference image, respectively, and the background is randomly selected and copied from the background of the reference image.

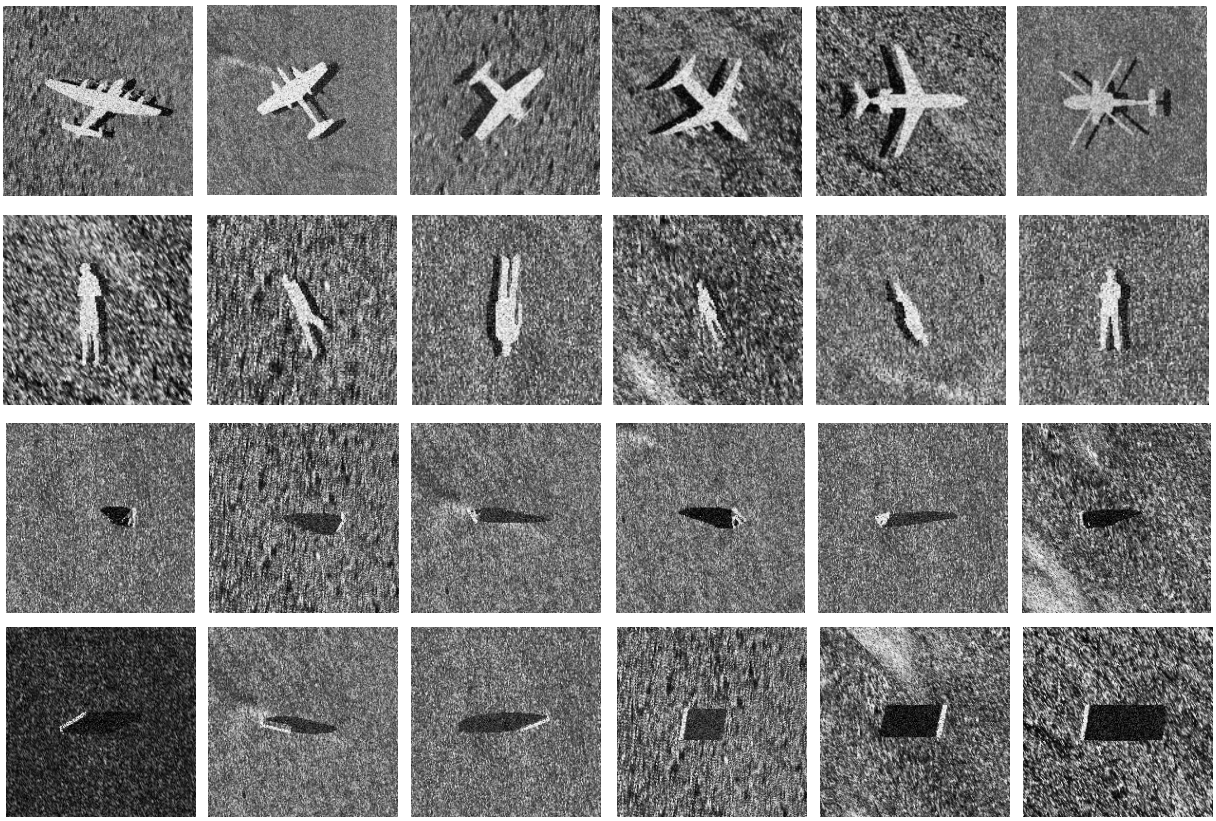

FIGURE 6. Some semisynthetic sidescan sonar images of airplanes, drowning victims and mines. The first and second rows are airplanes and drowning victims, respectively; and the third and fourth rows are wedge and cylinder mines, respectively.

be compared with the method using BOF descriptor [56] on SIFT features [57] and SVM classification, the method using a shallow CNN trained from scratch [40], the gcForest method using Deep Forest [58], the method using deep learning of small datasets [59]. All the methods used for comparison have been demonstrated to be effective on small-scale

training data: Before the rise of deep learning, the SIFT-based descriptors usually perform best among various local descriptors [60], and SVM can work well with only small samples [61]; the gcForest method is highly competitive to deep neural networks and can work well even when there are only small-scale training data; the method in [59] transfers all the layers of a pretrained VGG16 network except the last two full connect layers, add a new fully connected layer, and then fine-tunes it to achieve good performance on small datasets.

All the programs except that for the gcForest method are written in MATLAB R2018b environment with the Deep Learning Toolbox, and run on a GPU with GTX1080Ti. The computer used is installed with Intel Core i7-7800X 3.5 GHz CPU, 64G RAM, and Ubuntu18.04 operating system. Among all the CNNs, VGG19, which has been proven to be more effective in transfer learning, is preferred by the proposed method. To significantly save the training time of CNNs, several pre-trained deep learning toolbox models on ImageNet can be downloaded from the MATLAB Central, thanks to the work of the MATLAB Deep Learning Toolbox team. The program for the gcForest method is developed with Python 2.7.

For each class in the *SeabedObjects-KLSG* dataset, 70% and 30% of the images are randomly selected as training samples and testing samples, respectively. Therefore, the numbers of training wreck, drowning victim, airplane, mine and seafloor images are 270, 25, 43, 90, and 405, respectively; and the numbers of testing wreck, drowning victim, airplane, mine and seafloor images are 115, 11, 19, 39, and 173, respectively. To eliminate possible influence of sample partitioning on the performance of the classifier, a hold-out scheme was used to randomly create 10 datasets to test the classifier. To minimize the impact of random initialization of parameters, repeated tests of 10 times are conducted on each dataset and the average is taken as the result of the classification on this dataset. The average of the results on the 10 datasets is taken as the final result. The numbers of the semisynthetic drowning victim, airplane and mine images are 170, 198 and 207, respectively. All the semisynthetic sonar images will also be used for training to see if they can improve the classification accuracy.

Training options are also very important, and must be carefully selected for better results. For the proposed method, by trail-and-error, the solver of stochastic gradient descent with momentum (sgdm) is used, and the momentum is set to 0.9; the initial learning rate $\alpha = 0.0001$, and the rate is multiplied by a factor of 0.1 every time 5 epochs has passed; a batch size of 32 is adopted; a dropout scheme with probability $p = 0.5$ is used in the two fully connected layers before the final output layer. To accelerate the parameter learning of the newly added final fully connected layer, the multiplier for the learning rate of the weights and that for the for the learning rate of the biases are 10 and 20, respectively. For the shallow CNN, by trail-and-error, a batch size of 16 is adopted for better results. For the method in [59], we use the same parameters as in [59]: fine-tuning is learned by a stochastic gradient descent, with a learning rate of 0.0001 and a moment of 0.99; and only one fully connected layer with 32 hidden units is used with a dropout probability $p = 0.5$. For the gcForest method, the number of estimators and the number of maximum layers are both set to 100, and four typical types of classifiers are used, which are RandomForestClassifier, XGBClassifier, ExtraTreesClassifier, and LogisticRegression. For the method using BOF descriptor on SIFT features and SVM classification, the open-source VLFeat 0.9.21 (www.vlfeat.org) can be used and the size of BOF is set to 300. The test dataset is also used for validation in the training process, and the training will be terminated when the validation accuracy is stable.

TABLE 1. Overall accuracy of different methods.

Methods	OA (%) Using only real sonar images for training	OA (%) Using both the real and semisynthetic sonar images for training
BOF+SIFT +SVM	84.31%	87.96%
Shallow CNN	83.19%	85.52%
gcForest	86.76%	90.05%
Transferred VGG16 (no FC layers)	92.87%	96.08%
Transferred VGG19	93.83%	**97.76%**

B. EXPERIMENTAL RESULTS

The overall accuracy (OA), which is the percentage of all the correct positive classifications and can represent the overall classification performance, is first used to evaluate the performance of different methods. The OA results on the test datasets of all the five methods are given in Table 1, with each method has two OA results: one is achieved by using only the real sonar images for training, and the other is achieved by using both the real and the semisynthetic sonar images for training.

From Table 1, we can see that:

1) among all the five methods, fine-tuning a transferred VGG19 deep learning model performs the best, which can finally achieve an OA of 97.76% if both the real and semisynthetic sonar images are used for training;
2) fine-tuning a transferred VGG16 model without transferring the last two fully connected layers is the second best, which can achieve an OA of 96.08%;
3) for our classification task which involves objects with complex shapes, training a shallow CNN from scratch and then using it for classification will not perform very well, the OA result of which is the lowest among the five methods, and lower than the traditional method using BOF descriptor on SIFT features and SVM classification;
4) the gcForest method, which also have a deep structure, does perform a little better than training a shallow CNN

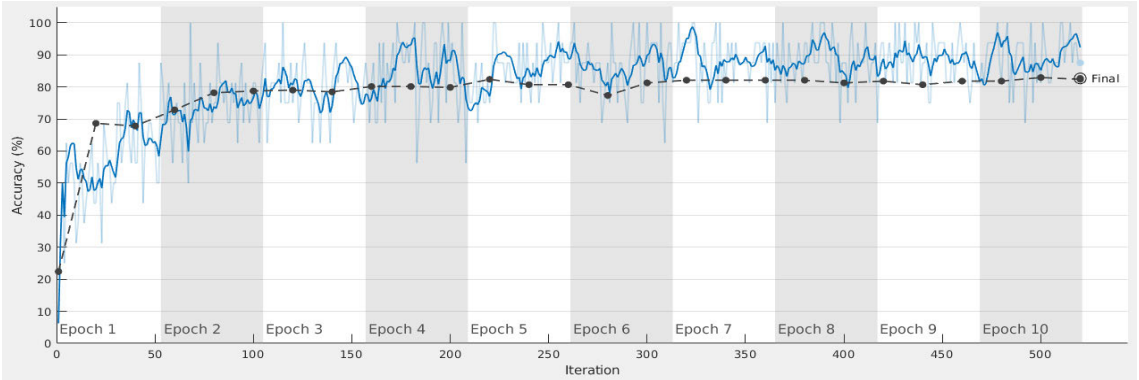

FIGURE 7. Training a shallow CNN from scratch using the real training dataset. The dark blue line represents the accuracy change of each training batch (batch size is 16) with iteration, while the black line represents the accuracy of the whole validation dataset.

from scratch, but it cannot outperform the two methods using transfer learning.

5) using the semisynthetic sonar images for training, the OA results of the five methods can all be further improved, which indicates the effectiveness of the proposed semisynthetic sonar image generation method.

The reason why fine-tuning a transferred VGG19 deep learning model can perform well is that by pretraining VGG19 on ImageNet which include abundant objects, the trained VGG19 have learned enough features for identifying different kinds of objects, and by fine-tuning the transferred VGG19 model, the weights are quickly adjusted to be more suitable for sidescan sonar images. The reason why training a shallow CNN from scratch will not perform very well here is that it cannot be used to distinguish complex shapes, which usually need a much deeper model, and that it still has a lot of different weights to learn. For comparison, a training progress of 10 epochs of the shallow CNN and that of the pre-trained VGG19 are shown in Fig. 7 and Fig. 8, respectively. From Fig. 7 and Fig. 8, we can see that while training a shallow CNN from scratch struggles for convergence in 10 epochs, fine-tuning the transferred VGG19 model can easily achieve fast convergence.

Although pre-training a deep neural network model (VGG16, VGG19, etc.) on ImageNet and then fine-tuning the pre-trained model on the real training dataset can achieve more accurate results than the other three methods, imbalance of real training data may still cause more misclassification, which can be improved by using semisynthetic sonar images for training. The training progress of fine-tuning the pre-trained VGG16 (without transferring the last two fully connected layers) and the pre-trained VGG19 using both real training dataset and semisynthetic data are given in Fig.9 and Fig.10, respectively, from which we can see that the convergence is faster and the accuracy of the training batch and that of the validation dataset are closer, when compared with Fig. 8. When comparing Fig. 9 with Fig. 10, it can be seen that fine-tuning the pretrained VGG19 is more stable and accurate than fine-tuning the pre-trained VGG16 (without transferring the last two fully connected layers). One reason may be that the last fully connected layer of a CNN can offer more transfer adaptation [62]. To further demonstrate the role of the semisynthetic data, the confusion matrixes on the test dataset using the VGG19 fine-tuned by only real training dataset and by both real training dataset and the semisynthetic data are given in Table 2 and Table 3, respectively.

TABLE 2. Confusion matrix on the test dataset using the VGG19 fine-tuned by only real training dataset.

True Class	Predicted Class				
	Drowning victim	Mine	Airplane	Wreck	Seafloor
Drowning victim	5	3		1	2
Mine	1	36			2
Airplane			7	12	
Wreck			1	114	
Seafloor					173

TABLE 3. Confusion matrix on the test dataset using the VGG19 fine-tuned by both real training dataset and the semisynthetic data.

True Class	Predicted Class				
	Drowning victim	Mine	Airplane	Wreck	Seafloor
Drowning victim	9	1			1
Mine		38			1
Airplane			15	4	
Wreck			1	114	
Seafloor					173

In Table 2 and Table 3, the number of each wrongly predicted class is marked in red, and that of the correctly predicted is marked in blue. From Table 2 and Table 3, we can see that:

1) among the five object classes, drowning victims and airplanes are more likely to be misclassified into other categories, due to less real training samples;
2) due to more samples and less feature complexity, seafloor images can be represented more effectively and therefore are all correctly classified in both cases;

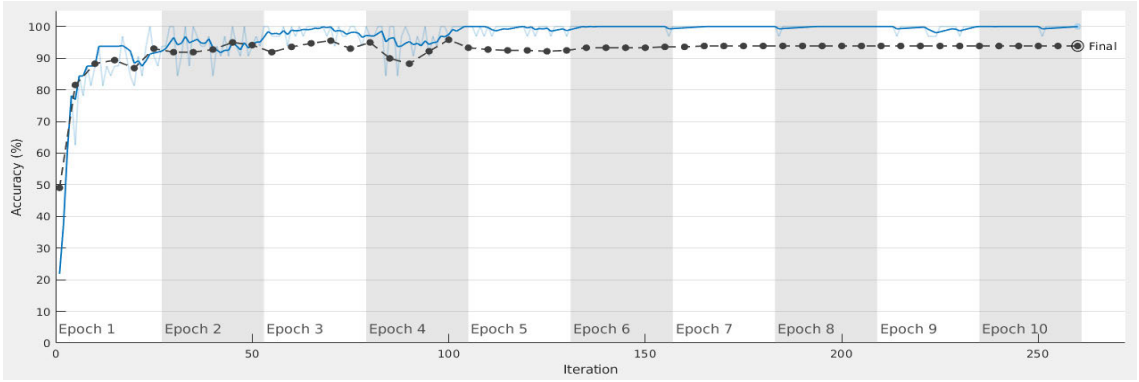

FIGURE 8. Fine-tuning the pre-trained VGG19 using the real training dataset. The dark blue line represents the accuracy change of each training batch (batch size is 32) with iteration, while the black line represents the accuracy of the whole validation dataset.

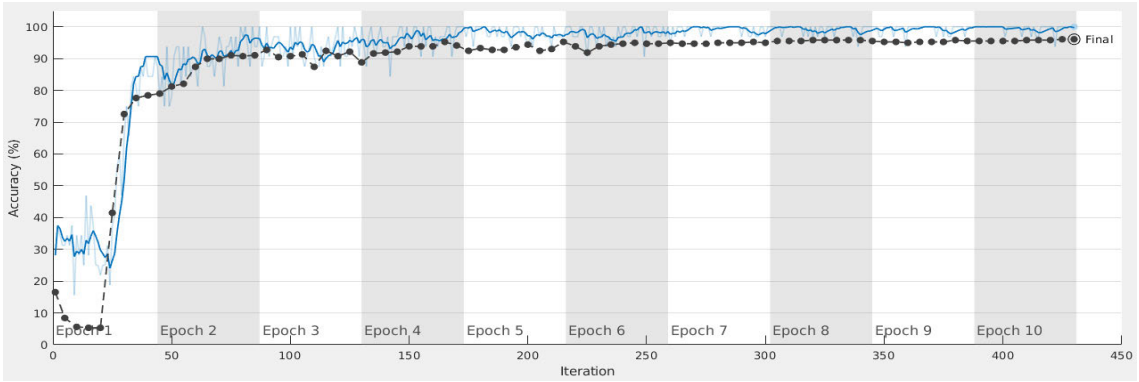

FIGURE 9. Fine-tuning the pre-trained VGG16 (without transferring the last two fully connected layers) using both the real training dataset and semisynthetic data. The dark blue line represents the accuracy change of each training batch (batch size is 32) with iteration, while the black line represents the accuracy of the whole validation dataset.

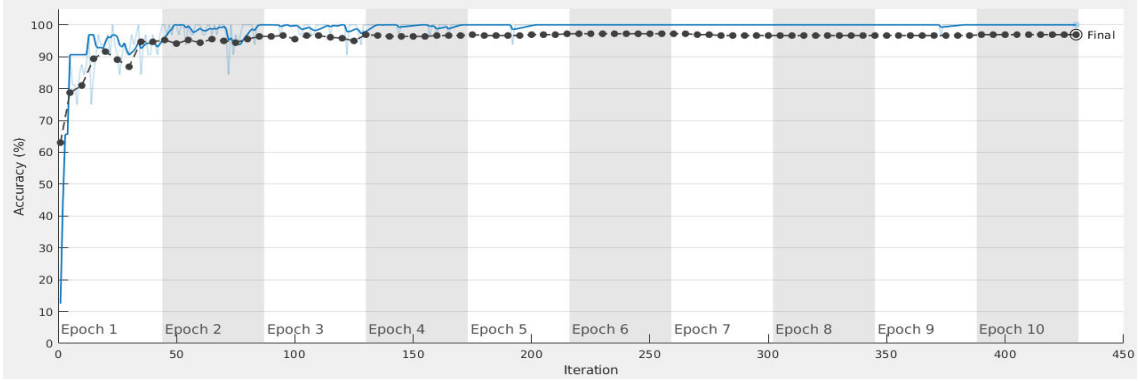

FIGURE 10. Fine-tuning the pre-trained VGG19 using both the real training dataset and semisynthetic data. The dark blue line represents the accuracy change of each training batch (batch size is 32) with iteration, while the black line represents the accuracy of the whole validation dataset.

3) by adding the semisynthetic data including drowning victim, airplane and mine images into the training dataset, misclassification of drowning victims, airplanes and mines can be effectively reduced.

Although the proposed semisynthetic data generation method is not strict, it can still be effectively used for training to improve the final classification accuracy, mainly because the semisynthetic sonar images preserve the major shape of an object and comply with the distribution of real sidescan sonar images.

We have demonstrated that the proposed semisynthetic data generation method can produce useful data for training a more accurate deep neural network. Furthermore, it would be interesting and meaningful to evaluate the quality of

TABLE 4. Scoring standard of the semisynthetic images and the related semantic terms.

Score	Semantic Terms
5	Excellent, indistinguishable from real images
4	Good, subtly distinguishable from real images
3	Fair, distinguishable from real images
2	Poor, significantly different from real images
1	Bad, too fake to be used

the generated semisynthetic data. Image quality assessment can be categorized as subjective versus objective. Subjective methods are based on psycho-physical experiments involving human observers. A total of 50 observers, including 42 graduate students who have took the course of digital image processing, and 8 teachers whose major is image processing, were invited to assign quality scores to each semisynthetic sidescan sonar image from an integer scale after training using the real sidescan sonar images. These scores characterize the similarity between the semisynthetic images and real images and are related to semantic terms, which are defined in Table 4.

The average score given by 50 observers for each semisynthetic image is regarded as the quality score of each image. All the scores belonging to the same category are further averaged to give an overall evaluation of the quality of this kind of semisynthetic images, which are given in Table 5. Average of scores of all semisynthetic images, which shows the overall quality of the synthesized images, is also given in Table 5.

In addition to the subjective assessment, an objective assessment of the semisynthetic samples is also given, which is based on the Fréchet Inception Distance (FID) [63]. FID uses the statistics of real samples and compare it to the statistics of synthetic samples, and have been widely used to evaluate the quality of synthetic samples of different generative adversarial networks since it was first adopted by Heusel [64]. Let $p(.)$ be the distribution of the synthetic samples and $p_w(.)$ the distribution of the real samples. The Fréchet Inception Distance $d(.,.)$, which is the distance between the Gaussian with mean and covariance (m, C) obtained from $p(.)$ and the Gaussian (m_w, C_w) obtained from $p_w(.)$, can be given by:

$$d^2\left((m, C), (m_w, C_w)\right) = \|m - m_w\|_2^2 + \mathrm{Tr}\left(C + C_w - 2\left(CC_w\right)^{1/2}\right) \quad (3)$$

where m and m_w refer to the feature-wise mean of the real and generated images; C and C_w are the covariance matrix for the real and generated feature vectors; and Tr refers to the trace linear algebra operation.

After fine-tuning an Inception model using all the real sonar images of airplane, drowning victim and mines, the output of the Average Pooling layer for the real images used for training can be assumed to follow a multidimensional Gaussian distribution. Meanwhile, if we take the semisynthetic images as input, the output of the Average Pooling layer will follow a different Gaussian distribution. Then the FID value can be calculated according to (3) to evaluate the similarity between the semisynthetic images and the real images, and the result is given in Table 5.

TABLE 5. Subjective average quality scores given by observers and objective assessment of FID.

AVG (Mine)	AVG (Airplane)	AVG (Drowning victim)	AVG (All)	FID
4.72	4.59	4.43	4.61	10.087

From Table 5, we can see the average score of all the semisynthetic images is 4.61, which shows that the overall quality is very good and close to excellent. It should be noted that average scores of three kinds of semisynthetic objects are different: average score of mine images is the highest, while that of drowning victim images is the lowest. Observers feel that the shape of mine in semisynthetic images can be well imagined and is the most real, but they are not very sure about the shape of drowning victims. The FID value is 10.087, which demonstrates that the semisynthetic samples are very similar to the real samples according to the experiments in [64]. Subjective evaluation is consistent with objective evaluation, both indicating that the semisynthetic images are very similar to the real images.

IV. CONCLUSION

To urgently promote underwater objects classification in sidescan sonar images, especially civilian objects classification, we built the real underwater object dataset of *SeabedObjects-KLSG*. We have demonstrated that pre-training a deep CNN, e.g., VGG19, and fine-tuning the transferred CNN can achieve more satisfactory results. Furthermore, we also proposed a semisynthetic sidescan sonar image generation method, and by adding the semisynthetic sonar images into the training dataset, we show that the classification accuracy can be improved. The quality of the semisynthetic data is also evaluated by subjective scores assigned by human observers and by objective value of FID, which both indicate that the semisynthetic images are very similar to the real images. The proposed method gave an effective and practical way for underwater object classification. Our work indicates that even for a small and imbalanced dataset, transfer learning is still preferred, and if we can simulate more data and offer a more balanced dataset for training, misclassification will be further reduced.

Because data play a vital role in deep learning methods, the *SeabedObjects-KLSG* dataset will be open on Github for research, and it is important for researchers to continually expand the dataset. Although the proposed semisynthetic data generation method can help to create more images and help to improve the final classification accuracy, it will be better to use a more precise simulation model, which will be considered in our future work. Although VGG19 has been preferred in transfer learning, other CNN models, such as

GoogLeNet and ResNet, can also be used and achieve similar performance. Besides model-based transfer learning, other transfer learning methods can also be tried to see if they can bring benefit. Our work can also be further combined with deep learning-based object detection methods such as R-CNN [27], YOLO [28], Faster R-CNN [65] and SSD [66] to locate and label all the objects in a sonar image consisting of multi objects.

ACKNOWLEDGMENT

The authors would like to thank L-3 Klein Associates, EdgeTech, Lcocean, Hydro-tech Marine, and Tritech for their great support for providing the valuable real sidescan sonar images during the past years, which contributes a lot to the establishment of our dataset.

REFERENCES

[1] T. Celik and T. Tjahjadi, "A novel method for sidescan sonar image segmentation," *IEEE J. Ocean. Eng.*, vol. 36, no. 2, pp. 186–194, Apr. 2011.

[2] S. Reed, Y. Petillot, and J. Bell, "An automatic approach to the detection and extraction of mine features in sidescan sonar," *IEEE J. Ocean. Eng.*, vol. 28, no. 1, pp. 90–105, Jan. 2003.

[3] C. Barngrover, R. Kastner, and S. Belongie, "Semisynthetic versus real-world sonar training data for the classification of mine-like objects," *IEEE J. Ocean. Eng.*, vol. 40, no. 1, pp. 48–56, Jan. 2015.

[4] X.-F. Ye, Z.-H. Zhang, P. X. Liu, and H.-L. Guan, "Sonar image segmentation based on GMRF and level-set models," *Ocean Eng.*, vol. 37, no. 10, pp. 891–901, Jul. 2010.

[5] M. Mignotte, C. Collet, P. Pérez, and P. Bouthemy, "Three-class Markovian segmentation of high-resolution sonar images," *Comput. Vis. Image Understand.*, vol. 76, no. 3, pp. 191–204, Dec. 1999.

[6] M. Mignotte, C. Collet, P. Perez, and P. Bouthemy, "Sonar image segmentation using an unsupervised hierarchical MRF model," *IEEE Trans. Image Process.*, vol. 9, no. 7, pp. 1216–1231, Jul. 2000.

[7] G. G. Acosta and S. A. Villar, "Accumulated CA–CFAR process in 2-D for online object detection from sidescan sonar data," *IEEE J. Ocean. Eng.*, vol. 40, no. 3, pp. 558–569, Jul. 2015.

[8] M. Lianantonakis and Y. R. Petillot, "Sidescan sonar segmentation using texture descriptors and active contours," *IEEE J. Ocean. Eng.*, vol. 32, no. 3, pp. 744–752, Jul. 2007.

[9] S. Daniel, S. Guillaudeux, and E. Maillard, "Adaptation of a partial shape recognition approach," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. Comput. Cybern. Simulation*, Oct. 1997, pp. 2157–2162.

[10] G. Huo, S. X. Yang, Q. Li, and Y. Zhou, "A robust and fast method for sidescan sonar image segmentation using nonlocal despeckling and active contour model," *IEEE Trans. Cybern.*, vol. 47, no. 4, pp. 855–872, Apr. 2017.

[11] A. Abu and R. Diamant, "A statistically-based method for the detection of underwater objects in sonar imagery," *IEEE Sensors J.*, vol. 19, no. 16, pp. 6858–6871, Aug. 2019.

[12] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, U.K.: MIT Press, 2016.

[13] S. Daniel, F. Le Leannec, C. Roux, B. Soliman, and E. P. Maillard, "Side-scan sonar image matching," *IEEE J. Ocean. Eng.*, vol. 23, no. 3, pp. 245–259, Jul. 1998.

[14] G. J. Dobeck, J. C. Hyland, and L. Smedley, "Automated detection and classification of sea mines in sonar imagery," *Proc. SPIE*, vol. 3079, pp. 90–110, Jul. 1997.

[15] H. Midelfart, J. Groen, and Ø. Midtgaard, "Template matching methods for object classification in synthetic aperture sonar images," in *Proc. Underwater Acoust. Meas. Conf.*, 2009, pp. 1–6.

[16] V. Myers and J. Fawcett, "A template matching procedure for automatic target recognition in synthetic aperture sonar imagery," *IEEE Signal Process. Lett.*, vol. 17, no. 7, pp. 683–686, Jul. 2010.

[17] H. Cho, J. Gu, and S.-C. Yu, "Robust sonar-based underwater object recognition against angle-of-view variation," *IEEE Sensors J.*, vol. 16, no. 4, pp. 1013–1025, Feb. 2016.

[18] J. Sawas, Y. Petillot, and Y. Pailhas, "Cascade of boosted classifiers for rapid detection of underwater objects," in *Proc. Eur. Conf. Underwater Acoust.*, vol. 10, 2010, pp. 1–8.

[19] C. Barngrover, A. Althoff, P. DeGuzman, and R. Kastner, "A brain-computer interface (BCI) for the detection of mine-like objects in sidescan sonar imagery," *IEEE J. Ocean. Eng.*, vol. 41, no. 1, pp. 123–138, Jan. 2016.

[20] P. Hollesen, W. A. Connors, and T. Trappenberg, "Comparison of learned versus engineered features for classification of mine like objects from raw sonar images," in *Advances in Artificial Intelligence* (Lecture Notes in Computer Science), vol. 6657. Berlin, Germany: Springer-Verlag, 2011, pp. 174–185.

[21] D. P. Williams, V. Myers, and M. Schatten Silvious, "Mine classification with imbalanced data," *IEEE Geosci. Remote Sens. Lett.*, vol. 6, no. 3, pp. 528–532, Jul. 2009.

[22] X. X. Zhu, D. Tuia, L. Mou, G.-S. Xia, L. Zhang, F. Xu, and F. Fraundorfer, "Deep learning in remote sensing: A comprehensive review and list of resources," *IEEE Geosci. Remote Sens. Mag.*, vol. 5, no. 4, pp. 8–36, Dec. 2017.

[23] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, Jul. 2006.

[24] A. Krizhevsky, I. Sutskever, and G. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2012, pp. 1097–1115.

[25] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: http://arxiv.org/abs/1409.1556

[26] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[27] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Region-based convolutional networks for accurate object detection and semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 1, pp. 142–158, Jan. 2016.

[28] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[29] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.

[30] J. Fu, J. Liu, Y. Wang, and H. Lu, "Densely connected deconvolutional network for semantic segmentation," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2017, pp. 1520–1528.

[31] D. Einsidler, M. Dhanak, and P.-P. Beaujean, "A deep learning approach to target recognition in side-scan sonar imagery," in *Proc. OCEANS MTS/IEEE Charleston*, Charleston, SC, USA, Oct. 2018, pp. 1–4.

[32] X. Cao, R. Togneri, X. Zhang, and Y. Yu, "Convolutional neural network with second-order pooling for underwater target classification," *IEEE Sensors J.*, vol. 19, no. 8, pp. 3058–3066, Apr. 2019.

[33] X. Cao, X. Zhang, Y. Yu, and L. Niu, "Deep learning-based recognition of underwater target," in *Proc. IEEE Int. Conf. Digit. Signal Process. (DSP)*, Beijing, China, Oct. 2016, pp. 89–93.

[34] P. Zhu, J. Isaacs, B. Fu, and S. Ferrari, "Deep learning feature extraction for target recognition and classification in underwater sonar images," in *Proc. IEEE 56th Annu. Conf. Decis. Control (CDC)*, Melbourne, VIC, Australia, Dec. 2017, pp. 2724–2731.

[35] K. Denos, M. Ravaut, A. Fagette, and H.-S. Lim, "Deep learning applied to underwater mine warfare," in *Proc. OCEANS*, Aberdeen, Scotland, Jun. 2017, pp. 1–7.

[36] K. Zhu, J. Tian, and H. Huang, "Underwater object images classification based on convolutional neural network," in *Proc. IEEE 3rd Int. Conf. Signal Image Process. (ICSIP)*, Shenzhen, China, Jul. 2018, pp. 301–305.

[37] D. P. Williams, "Underwater target classification in synthetic aperture sonar imagery using deep convolutional neural networks," in *Proc. 23rd Int. Conf. Pattern Recognit. (ICPR)*, Cancún, Mexico, Dec. 2016, pp. 2497–2502.

[38] T. Rimavicius and A. Gelzinis, "A comparison of the deep learning methods for solving seafloor image classification task," *Inform. Softw. Technol.*, vol. 756, pp. 442–453, 2017.

[39] A. Diegues, J. Pinto, P. Ribeiro, R. Frias, and D. C. Alegre, "Automatic habitat mapping using convolutional neural networks," in *Proc. IEEE/OES Auto. Underwater Vehicle Workshop (AUV)*, Nov. 2018, pp. 1–6.

[40] X. Luo, X. Qin, Z. Wu, F. Yang, M. Wang, and J. Shang, "Sediment classification of small-size seabed acoustic images using convolutional neural networks," *IEEE Access*, vol. 7, pp. 98331–98339, 2019.
[41] Y. Wang, C. Wang, H. Zhang, Y. Dong, and S. Wei, "A SAR dataset of ship detection for deep learning under complex backgrounds," *Remote Sens.*, vol. 11, no. 7, p. 765, 2019.
[42] Y. Wang, C. Wang, H. Zhang, Y. Dong, and S. Wei, "Automatic ship detection based on RetinaNet using multi-resolution Gaofen-3 imagery," *Remote Sens.*, vol. 11, no. 5, p. 531, 2019.
[43] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
[44] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.
[45] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.
[46] H. Chang, J. Han, C. Zhong, A. M. Snijders, and J.-H. Mao, "Unsupervised transfer learning via multi-scale convolutional sparse coding for biomedical applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 5, pp. 1182–1194, May 2018.
[47] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Unsupervised domain adaptation with residual transfer networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2016, pp. 136–144.
[48] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Learning and transferring mid-level image representations using convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Columbus, OH, USA, Jun. 2014, pp. 1717–1724.
[49] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2014, pp. 3320–3328.
[50] J. Bell, "A model for the simulation of sidescan sonar," Ph.D. dissertation, Dept. Comput. Elect. Eng., Heriot-Watt Univ., Edinburgh, Scotland, Aug. 1995.
[51] O. George and R. Bahl, "Simulation of backscattering of high frequency sound from complex objects and sand sea-bottom," *IEEE J. Ocean. Eng.*, vol. 20, no. 2, pp. 119–130, Apr. 1995.
[52] J. M. Bell, "Application of optical ray tracing techniques to the simulation of sonar images," *Opt. Eng.*, vol. 36, no. 6, pp. 1806–1813, Jun. 1997.
[53] T. Capéran, G. Hayward, and R. Chapman, "A 3D simulator for the design and evaluation of sonar system instrumentation," *Meas. Sci. Technol.*, vol. 10, no. 12, pp. 1116–1126, Dec. 1999.
[54] J. A. Fawcett, "Modeling of high-frequency scattering from objects using a hybrid Kirchhoff/diffraction approach," *J. Acoust. Soc. Amer.*, vol. 109, no. 4, pp. 1312–1319, Apr. 2001.
[55] J. A. Fawcett and R. Lim, "Evaluation of the integrals of target/seabed scattering using the method of complex images," *J. Acoust. Soc. Amer.*, vol. 114, no. 3, pp. 1406–1415, Sep. 2003.
[56] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *Proc. Workshop Stat. Learn. Comput. Vis. (ECCV)*, 2004, pp. 1–22.
[57] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004.
[58] Z.-H. Zhou and J. Feng, "Deep forest: Towards an alternative to deep neural networks," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, Macao, China, Aug. 2017, pp. 3353–3359.
[59] Y. Wang, C. Wang, and H. Zhang, "Ship classification in high-resolution SAR images using deep learning of small datasets," *Sensors*, vol. 18, no. 9, p. 2929, 2018.
[60] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 10, pp. 1615–1630, Oct. 2005.
[61] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 1–27, Apr. 2011.
[62] E. Tzeng, J. Hoffman, N. Zhang, and K. Saenko, "Simultaneous deep transfer across domains and tasks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2015, pp. 4068–4076.
[63] D. C. Dowson and B. V. Landau, "The Fréchet distance between multivariate normal distributions," *J. Multivariate Anal.*, vol. 12, no. 3, pp. 450–455, Sep. 1982.
[64] M. Heusel, H. Ramsauer, T. Unterthiner, and B. Nessler, "GANs trained by a two time-scale update rule converge to a local Nash equilibrium," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2017, pp. 6626–6637.
[65] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.
[66] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot MultiBox detector," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 21–37.

GUANYING HUO (Member, IEEE) received the B.Sc. degree in telecommunications engineering and the M.Sc. degree in information and communications engineering from Xidian University, Xi'an, China, in 2001 and 2004, respectively, and the Ph.D. degree in computer application technology from Hohai University, Nanjing, China, in 2012. He is currently a Postdoctoral with the Second Institute of Oceanography, Ministry of Natural Resources, and an Associate Professor with Hohai University. His current research interests include sonar image processing and deep learning.

ZIYIN WU received the Ph.D. degree in marine geology from Zhejiang University, Zhejiang, China, in 2008. He is currently a Professor with the Second Institute of Oceanography, Ministry of Natural Resources and the School of Oceanography, Shanghai Jiao Tong University. His primary research interests include seabed survey and information systems.

JIABIAO LI received the Ph.D. degree in marine geology from the Institute of Oceanology Chinese Academy of Sciences, Qingdao, China, in 2001. He is currently the Director of the Second Institute of Oceanography, Ministry of Natural Resources, and an Academician of the Chinese Academy of Engineering. His primary research interests include submarine geological science and submarine exploration engineering.