Fish recognition using deep convolutional neural network and data augmentation

Ziqiang Zheng, Chao Wang, Zhibin Yu*, Haiyong Zheng, Weiwei Wang
College of Information Science and Engineering
Ocean University of China
Qingdao 266100, China
Email: zhengziqiang1@gmail.com
*Corresponding author: yuzhibin@ouc.edu.cn

***Abstract*—Nowadays, as a sub topic of computer vision and fishery industry, fish recognition is still a challenging work not only because of various kinds of fish, but also because of the complex background of images. In this paper, we aim to classify different fish images obtained from cameras of fishing vessels. Fish detection is different with the best known and the most well-investigated object detection - face detection. Fish has more different shapes than human faces. And fish always take only a small part of the whole image. For these two reasons, it's a challenge for us to detect and classify the fish. Our work is done with Kaggle dataset. The Kaggle dataset aims to detect and classify the species of fish. The competition provides a train dataset which contains six species of tuna fish, fish but not tuna and no fish. Eight target categories are available in the dataset. The dataset is critically imbalanced, the Albacore tuna has thousands of images while the Opah has only sixty images. In order to overcome these problems we introduced two methods to improve our accuracy for the fish classification. Our approach can avoid over-fitting caused by the imbalance of training dataset. We employ the AlexNet, GoogLeNet, Caffenet and VGGNet neural network for the classification. As a result of the small dataset, the VGGNet architecture performs worse than AlexNet. We propose a local region based the fish area modeling approach so that the local feature can be modeled. In order to obtain more image information and realize imbalanced datasets classification, we have done some data augmentation. In order to get better performance, we also have done some image preprocessing, and it really works.**

I. Introduction

Nearly half the population of the world depends on seafood for their main source of protein. And fish play an important role in our marine ecosystem. So it's valuable for us to do the fish detection and classification. We research to achieve the fish detection and classification automatically[1], which has drawn increasing attention all over the world. And with the fast development of marine resources, the ocean observation is required constantly. The fish recognition is one of the important parts in ocean observation[2].

In the western and central pacific, where 60% of the world's tuna is caught, illegal, unreported, and unregulated fishing practices are threatening marine ecosystems, global seafood supplies and local livelihoods. The Nature Conservancy is inviting the Kaggle community to develop the algorithms to automatically detect and classify species of tunas. The Kaggle dataset includes Albacore tuna, Bigeye tuna, Yellowfin tuna, Mahi, Opah, Shark, Other(meaning that there are fish present but not in the above six categories), and No Fish(meaning that no fish in the picture). We divided the Kaggle dataset into training dataset and validation dataset. And we take four fifths of the images provided by Kaggle as the training dataset and take the rest as the validation dataset. Each image has only one fish category. And the Table I shows the imbalance of the Kaggle dataset. From this table we can see that the top three categories of fish images almost account for four fifths of the dataset.

Most prior recognition researches are based on the object on the ground such as face detection, car detection and pedes-

TABLE I
Kaggle database

classes	Total	Training	Valiation
ALB	45.5%	43.1%	47.5%
YFT	19.4%	20.6%	21.0%
NoF	12.3%	13.8%	13.3%
OTHER	7.9%	7.6%	8.1%
BET	5.2%	5.4%	4.2%
Sum	90.3%	90.5%	94.0%

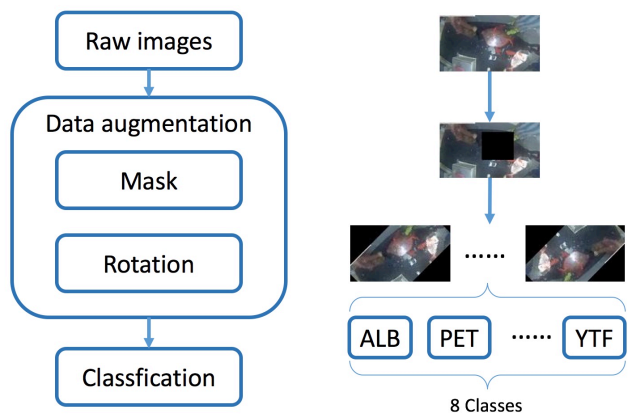
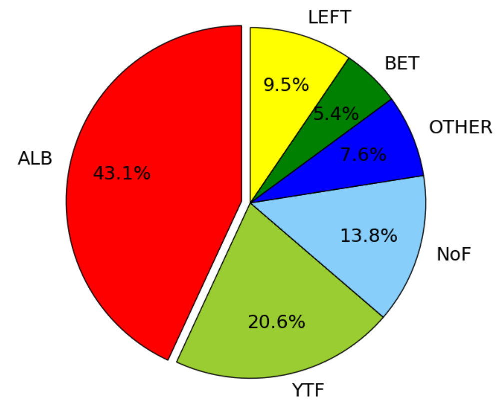


Fig. 1. First we make a mask in the fish region, then we rotate the processed images at different angles and get some new different images. After the data augmentation, at last we use the CNN to fish the fish classification.

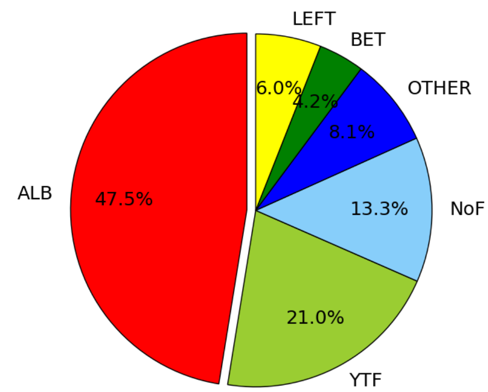
trian detection[3]. Fish detection is different from recognition researches on the ground[4]. For the reason that the fish images are usually uncleared and the fish only cover small areas in an image, the fish detection is more difficult than the object detection on the ground. And the shape of the fish is more complex than face and cars. So the fish recognition needs more sophisticated detection and requires the methods to locate where the fish are in the fish images[5]. It's difficult for our machines to distinguish the fish and the background. Even some machine leaning methods can learn some high-level features from the fish images, they can't make sure that the extracted features are that really make contribution to the recognition. Another challenge for fish recognition is that the images from the cameras of fishing vessels are uncleared. It's difficult for our model to learn the characteristics of the object by the unclear pictures[6]. Some image processing methods could be employed to solve the problem.[7] We have used some image processing methods in our experiments and it really works. Recently, several papers have brought some methods for fish detection and classification[8]. We have employed the deep convolutional neural network for the research because of its fantastic performance in image detection and classification. It is an efficient classification method for images and widely used in object recognition tasks[9].

In this paper, we will introduce two methods to improve the classification accuracy. Our methods can be described with the Fig. 1. We have done some data preprocessing before training our model. First we make a mask in the fish region and make the fish region black, which we regard as the NoF fish images. Then we use some methods to achieve data augmentation. At last we use the convolutional neural network(CNN) as the classifier. The first method is based on data augmentation. In consideration of the imbalance of the dataset, we got some new images by rotating the selected images. Through this method, we can increase the number of some species of fish images, which can help avoid that our model is over-fitted with some categories of fish images, and rotating the fish images can improve the robustness of detection and achieve sophisticated detection. We increased the size of our datasets in this way and the validation accuracy increased two percent.

The second improvement is based on an image preprocessing method. In order to force the neural networks to focus on fish not the vessel, we have made a mask of the area where the fish are located, and we have used both the processed images and the raw images to train our model. Fig. 3 shows the differences between the two kinds of images. This method can force our neural network to concentrate on the area where the fish present. And the features are learned from comparing the raw images and the images with mask. In order to prove our assumption, we test some fish images and get the image data from the last pooling layer. The higher value area in the pooling layer means higher contribution to the final decision. Through comparing the pooling gray image with the raw image, we found that the area of fish is brighter than other areas, which proves what we expected that the fish area make



(a) Training database



(b) Valing database

Fig. 2. Data distribution of Kaggle

greater contribution in the classification. Fig. 4 illustrates the image information of the region of fish and other regions without fish. We can see that the region of fish is brighter than other regions and the pooling image data can confirm that the learned features are from fish. In addition, we use the processed images to fine-tune our pre-trained model and it performs better when testing.

The architecture of our model used in the experiment is the convolutional neural network, which has achieved remarkable performance in image classification and object detection recently[10][11]. And CNN is usually composed of convolutional layer, pooling layer and fully connected layer. Compared with feature-designed extraction, it can explore more abstract and high-level information by deep neural network[12]. Convolution neural networks have been applied with great success to the image recognition of objects on the ground. Anologously we can apply it to the fish recognition. We have employed different CNN architectures for the fish classification. But there still exists the problem that our model can not distinguish between foreground and background. Our methods can prevent the CNN architecture model from concentrating on the boat region rather than the fish region. In addition, we use the processed images to fine-tune our pre-trained model and it performs better when testing. This method can help avoid over-fitting the of our model.

(a) the raw image

(b) image with mask

Fig. 3. We make a mask of the area where the fish are located, because the mask can force our model to concentrate on the fish area by comparing the raw images and the processed images. So it can help our convolutional neural network detect the fish region. we have trained our model using both the two images at the same time, which can improve the robustness of the fish recognition.

II. Experiments

The Table I indicates the details of the Kaggle Database. The first column is the name of fish and the numbers in the rest columns mean that the proportion of this class in all fish. And the Fig. 2 is also the another form of fish proportion. We can see the distribution of the dataset is imbalanced. The ALB and YFT fish images cover nearly two thirds of the image dataset while the LAG fish category only account for a few percent.

TABLE II
SINGAL MODEL RESULT ON FULL DATA

database	model	accuracy
full dataset	alexnet	0.94
full dataset	googlenet	0.942
full dataset	vgg16	0.44

TABLE III
THE TEST LOSS IN THE KAGGLE COMPETITION

method	model	test loss
None	Caffenet	1.92
None	GoogleNet	2.56
Rotating	Caffenet	1.77
Rotating	GoogleNet	1.93
Mask	Caffenet	1.87
Mask	GoogleNet	2.25
Rotating + Mask	Caffenet	1.71
Rotating + Mask	GoogleNet	1.85

The imbalance of the dataset can lead to the over-fitting of model, and our model can't extract enough features from those fish categories that account for only a small part of the dataset. So it's a challenge for our model to realize the feature learning on the imbalanced dataset. In order to solve this problem, we rotate some selected fish images to get new fish images. We got eight times images by rotating the fish images at different angles. We have made the data augmentation by this method, which can help to solve the imbalance of the dataset.

The Table II illustrates the benchmarks. The Table III shows the test loss of our models uploaded in the Kaggle competition. We have trained the dataset with four different deep convolutional neural network structure, Alexnet, GoogleNet, Caffenet and VGGNet[13]. Above the accuracy table, we can see that the AlexNet and the GoogLeNet achieved greater performance than the VGGNet. Due to the imbalance and the insufficience of the dataset, the VGGNet could be over-fitted with our training set so that it performs badly when testing. And the VGGNet wastes more time to converge on small dataset than GoogleNet and Alexnet. Comparing the accuracy between Alexnet and GoogleNet, we come to the conclusion that the GoogleNet performs better than Alexnet because of more layers and parameters. The Table IV shows the accuracy that we have rotated some images to get new images. From the table we can see this method actually worked out and the accuracy has increased about two percent. And by comparing the loss we can see that the loss has decreased by using the rotating method. According to this, the data augmentation method can help our model extract high-level features from the fish images. And our model can be more robust in this way. Basing this, we have made a mask of the region of fish for some fish images. In this experiment, we have picked up about one fourth of the seven fish category images except for the NoF images, we made the mask int the images and replace the region where the fish are with the

TABLE IV
KAGGLE DATABASE

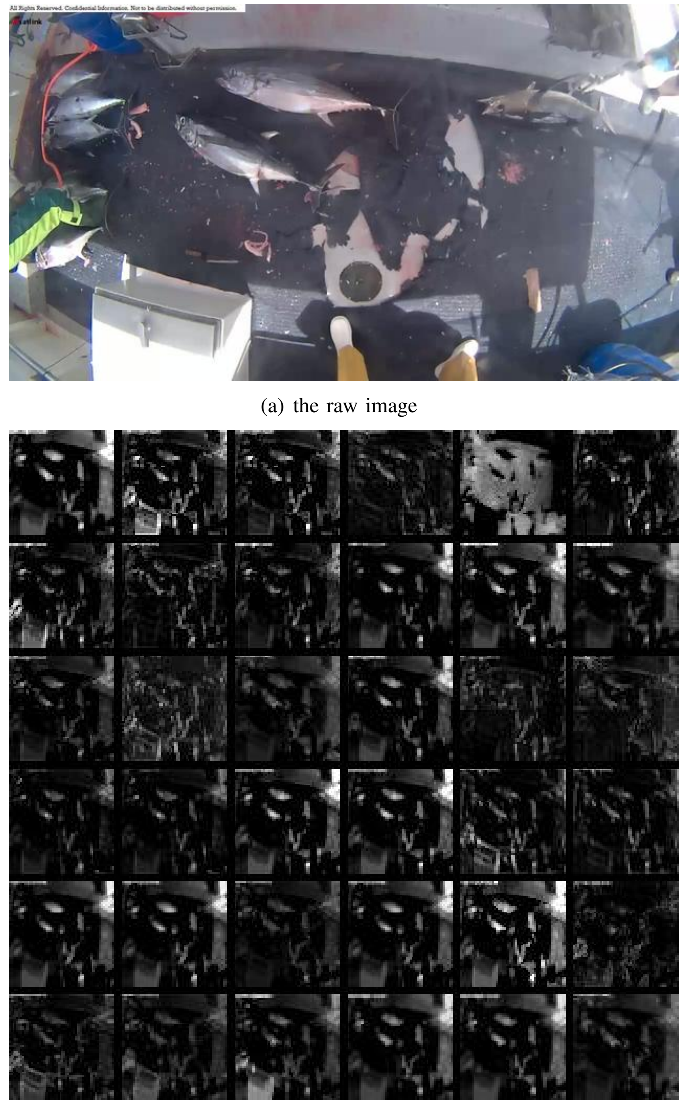
database	model	accuracy
full+rotating	alexnet	0.9673
full+rotating	googlenet	0.964
full+rotating	vgg16	0.45

TABLE V
KAGGLE DATABASE

database	model	accuracy
full+rotating+mask	alexnet	0.9703
full+rotating+mask	googlenet	0.954

black areas. Then we put these processed images into the NoF category. The Fig. 3 shows the differences between the raw image and the image with mask. Then We trained our model with the raw images and the processed images using different architecture of networks. By comparing these different images, our model can concentrate on the region of the fish and learn more features of fish rather than the feature of the background. The accuracy has also increased in the Table V, which proves that our methods are effective. By comparing the test loss int the Table III, we can see that both the rotating method and the mask method can decrease the test loss. The test loss when using both the two methods at the same time has decreased more than only using one method. And because we make a mask in the fish region, our model can concentrate on the fish area quickly. So the mask method can accelerates the convergence rate.

In order to prove that the fish region has really made contribution in the fish recognition, we modified Caffe[14] to get the pooling images. And the Fig. 4 shows the comparison between the raw images and the pooling images. In consideration of the big number of the pooling images, we have picked up 36 pooling images from 64 pooling images and resized each of all to merge them into one image. From the two sets of images, we can see that the fish region is brighter than other regions in most of the pooling images, and this proves that fish make more contribution than the background for the classification.



(a) the raw image

(b) the pooling image

Fig. 4. There are 36 pooling images in the final pooling image. Considering that the size of the pooling images and the number of them, we make them smaller by resizing the images and merge them into one image. By comparing the two images, The region of fish are brighter than other regions in the most of the pooling images, which can prove that the fish has more influence on the classification than other regions.

III. CONCLUSION

In this paper, we used two methods to improve the deep neural networks to detect the fish more efficiently. The experiment results show that our methods can classify fish from images captured on a camera of fishing vessels well. And the classification accuracy has increased more than two percent while the test loss decreased. The data augmentation method can solve the imbalance of the dataset well and it can also improve the accuracy of our model. The image processing methods such as rotating and making mask can also work to get the better performance. These methods could be used for sophisticated detection and recognition, which can help to get more high-level and abstract features from images. Our next research is to merge the different networks and make good use of them to get a better performance. Our research can benefit the develop of the marine resources, as well as commercial applications such as fisheries and aquaculture.

REFERENCES

[1] C. Spampinato, D. Giordano, R. Di Salvo, Y.-H. J. Chen-Burger, R. B. Fisher, and G. Nadarajan, "Automatic fish classification for underwater species behavior understanding," in *Proceedings of the first ACM international workshop on Analysis and retrieval of tracked events and motion in imagery streams*. ACM, 2010, pp. 45–50.

[2] Á. Rodríguez, M. Bermúdez, J. R. Rabuñal, and J. Puertas, "Fish tracking in vertical slot fishways using computer vision techniques," *Journal of Hydroinformatics*, vol. 17, no. 2, pp. 275–292, 2015.

[3] K. Anantharajah, Z. Ge, C. McCool, S. Denman, C. Fookes, P. Corke, D. Tjondronegoro, and S. Sridharan, "Local inter-session variability modelling for object classification," in *IEEE Winter Conference on Applications of Computer Vision*. IEEE, 2014, pp. 309–316.

[4] R. Socher, B. Huval, B. P. Bath, C. D. Manning, and A. Y. Ng, "Convolutional-recursive deep learning for 3d object classification." in *NIPS*, vol. 3, no. 7, 2012, p. 8.

[5] P. X. Huang, B. J. Boom, and R. B. Fisher, "Underwater live fish recognition using a balance-guaranteed optimized tree," in *Asian Conference on Computer Vision*. Springer, 2012, pp. 422–433.

[6] M. T. Rodrigues, F. L. Padua, R. M. Gomes, and G. E. Soares, "Automatic fish species classification based on robust feature extraction techniques and artificial immune systems," in *Bio-Inspired Computing: Theories and Applications (BIC-TA), 2010 IEEE Fifth International Conference on*. IEEE, 2010, pp. 1518–1525.

[7] H. Lu, Y. Li, and S. Serikawa, "Underwater image enhancement using guided trigonometric bilateral filter and fast automatic color correction," in *2013 IEEE International Conference on Image Processing*. IEEE, 2013, pp. 3412–3416.

[8] M.-C. Chuang, J.-N. Hwang, and K. Williams, "A feature learning and object recognition framework for underwater fish images," *IEEE Transactions on Image Processing*, vol. 25, no. 4, pp. 1862–1872, 2016.

[9] H. Qin, X. Li, J. Liang, Y. Peng, and C. Zhang, "Deepfish: accurate underwater live fish recognition with a deep architecture," *Neurocomputing*, 2015.

[10] A. Sharif Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "Cnn features off-the-shelf: an astounding baseline for recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2014, pp. 806–813.

[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[12] H. Lee, R. Grosse, R. Ranganath, and A. Y. Ng, "Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations," in *Proceedings of the 26th annual international conference on machine learning*. ACM, 2009, pp. 609–616.

[13] Z. Ge, C. McCool, C. Sanderson, and P. Corke, "Modelling local deep convolutional neural network features to improve fine-grained image classification," in *Image Processing (ICIP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 4112–4116.

[14] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," *arXiv preprint arXiv:1408.5093*, 2014.