A Hybrid Convolutional Neural Network for Plankton Classification

Jialun Dai, Zhibin Yu, Haiyong Zheng(✉), Bing Zheng, and Nan Wang

College of Information Science and Engineering,
Ocean University of China, Shandong 266100, China
zhenghaiyong@ouc.edu.cn

Abstract. Plankton are fundamental and essential to marine ecosystem, and its survey is significant for sustainable development and ecosystem balance of oceans. The large amount of plankton species and complex relationship among different classes bring difficulty for us to design an automatic plankton classification system. Thus, we develop our model based on convolutional neural network and aim to overcome these shortages. We consider two different ways to extract global and local features to describe shape and texture information of plankton. Furthermore, we design a pyramid fully connected structure to merge different inner products from each sub networks. The experimental results prove our model can take advantage of multiple features and performs better than original convolutional neural network.

1 Introduction

Plankton are a diverse group of organisms that live in the water column of large bodies of water and that cannot swim against a current, which provide a crucial source of food to many large aquatic organisms, such as fish and whales [1]. Diversity and abundance of plankton vary with the currents, geography of seas and ocean basins, and atmospheric conditions. Plankton composition also changes with the seasons, climate, and pollution [2]. So the plankton survey, including species composition, abundance distribution as well as their spatial and temporal changes, has a very important scientific and practical significance for our marine ecosystem, environmental monitoring and marine fishery.

Traditional plankton survey by net and water sampling is hard to meet the demand of *in situ* long-time continuous observation and large-scale fast real-time analysis, especially more and more *in situ* plankton imaging systems are developed and deployed currently [3], such as VPR (Video Plankton Record) [4] and SPC (Scripps Plankton Camera) [5]. Scientists are increasingly using imaging-based technologies to study these creatures in their natural habit, and images from such systems provide an unique opportunity to model and understand plankton ecosystems. Therefore, more and more image datasets for plankton classification are published recently [5,6].

Due to the increasingly plankton image datasets, the corresponding plankton analysis attracts more and more attention [7–9]. Based on the *in situ* images

C.-S. Chen et al. (Eds.): ACCV 2016 Workshops, Part III, LNCS 10118, pp. 102–114, 2017.
DOI: 10.1007/978-3-319-54526-4_8

captured by VPR [4] and SIPPER (Shadowed Image Particle Profiling Evaluation Recorder) [10], Xiaoou Tang *et. al.* combined the shape and texture features and studied the plankton classification (at most 7 classes) via neural networks (NN), principle component analysis (PCA), support vector machine (SVM), etc. [11,12]. By using ZooScan integrated system [13], Gaby Gorsky *et. al.* acquired the zooplankton images and extracted more than sixty properties as features to train six different classifiers (NN, SVM, and Random Forest) for predicting abundance of 20 categories [14]. Besides, Heidi Sosik and Robert Olson [15] studied the automated taxonomic classification of phytoplankton in 22 categories sampled with imaging-in-flow cytometry. Also, ADIAC (Automatic Diatom Identification And Classification) [16] and DiCANN (Dinoflagellate Categorisation by Artificial Neural Network) [17] focused the classification on Diatom and Dinoflagellate respectively, while also used the basic shape and texture feature descriptors and traditional classifiers such as NN, SVM, RF (Random Forest), and DT (Decision Tree), on classification of less than 6 categories.

As we see, although plankton image classification has been addressed for more than two decades, this issue is still very challenging because:

1. both the species and the morphologies are huge;
2. both the intra-class variance and the inter-class similarity are large;

which make this problem a very hard fine-grained visual recognition [18], so that the classical classifiers (NN, SVM, RF, DT, etc.) with traditional handcrafted features (SIFT, HOG, LBP, etc.) cannot be work well due to the overfitting [19].

Benefit from the big data and high performance computing, deep learning has been proved more and more workable in many fields such as image classification, speech recognition, and bioinformatics [20].

Deep learning methods, which bring a possible solution for big data analysis and image classification problems, can be used to overcome the shortage of the current plankton research. Convolutional neural networks (CNNs) belong to a typical deep neural network. With the deeper and larger of networks, the CNNs can be more capable to solve the problem of large-scale and difficult image recognition. And CNNs have achieved remarkable performance in image and video recognition [21–23].

The design of convolutional neural networks follows the discovery of visual mechanisms in living organisms. CNNs are originated from the neocognitron. The necognitron is introduced in 1980 by Fukushima [24]. It can be seen as a predecessor to convolutional networks. The neocognitron differs from convolutional networks because it does not force units located at several positions to have the same trainable weights. In 1989, LeCun integrated some constraints in backpropagation network to enhance the ability of learning networks to improve the final predictions [25]. And it is shown to outperform all other techniques in 1998 [26]. CNNs are further improved by Krizhevsky and his colleagues in 2012 [21]. They develop AlexNet and become the champion of ImageNet Large Scale Visual Recognition Challenge in 2012. With the powerful learning capacity, convolutional neural networks also can be applied to various tasks for resolving different problems.

For convolutional neural network, it can achieve remarkable performance in image classification under the conditions that it requires many images, hardware and good-designed network structure. Feature designed methods always can obtain good results in some special occasions, and can not apply to more classes or images further. However, based on data and learning capacity, convolutional neural network can explore more abstract and high-level information compared with feature designed methods, which means that it can find more useful patterns to obtain better results in image classification.

Unlike the common features used to classify faces, cars or other objects in daily life, plankton have their own special features such like setae and texture. In this paper, we aim to use different feature descriptors to extract those special plankton features. We not only use original images but also consider 2 traditional feature extraction methods to capture local and global plankton features for classification tasks based convolutional neural network. We believe that the global feature images can represent plankton appearance information and the local feature images describe the plankton texture. In order to take advantage of 3 different features, we designe a pyramid structure in fully connected layer to emerge the information from three channels.

As we mentioned before, inter-class similarity and intra-class variance make plankton classification difficult. Inter-class similarity means that plankton from different classes may be similar and intra-class variance means that plankton from the same class may be various. Humans always classify different objects from shape at first. However, this way does not work well in plankton classification. For different plankton class, the shape of different plankton may be similar. And under this condition, we can distinguish plankton texture, which is the inner-feature in plankton. Texture are various for different plankton class. For the same plankton class, texture are always similar though the shape of plankton from the same class may be different. So shape and texture are very important in plankton classification. And we can transform original images to global feature images and local feature images to represent shape and texture, and fuse these features in convolutional neural networks to improve performance in plankton classification.

Our main contribution is not only a hybrid convolutional neural network model, but a way to analyze plankton images. In biological point of view, both shape and texture are important to classify planktons. And we try to take advantage of this point to design the network model. Experimental results show that our model can achieve better result compared with a single CNN model.

The structure of our paper is organized as follows: some related methods about plankton classification are introduced in Sect. 2. The proposed approach is illustrated in Sect. 3, where we give a detail description about the network composition. In Sect. 4, some experimental results and analysis are discussed in this part. The final conclusion is given in Sect. 5.

2 Research Background

2.1 Convolutional Neural Network

A typical convolutional neural network (CNN) consists of 3 different neuron layers: convolutional layers, pooling layers and fully connected layers.

Convolutional Layers. As a typical deep neural network, each neuron of CNN in the convolutional layer is formed using the input from a local receptive field in the preceding layer and the learned kernels (weights). Neurons within the same feature map share the same kernels but are obtained using different input receptive fields. The kernels used from different feature maps in the same layer are different. Subsequently, an activation function is used. This can be represented as:

$$y_{ij}^{kl} = f((W^k * x)_{ij} + b_j^{kl}) \tag{1}$$

x is the input; i and j is the central position of convolutional computation between kernel and input; y_{ij}^{kl} is the output value of the k^{th} feature map from the l^{th} layer; W^k is the kernel weights; b_j^{kl} denotes the bias of this feature map and $f(x)$ is the activation function.

Pooling Layers. The pooling layer is a form of non-linear down-sampling. Its function is to progressively reduce the spatial size of the representation to reduce the amount of parameters and computation in the network, and hence to also control overfitting. Besides, it provides a form of translation invariance. The pooling layer only changes the size of the input maps while not altering the number of input maps. Averaging and the maximum are the most popular ways to implement the pooling operations. And maxing pooling is the most common implement which is also used in our experiment.

$$y_j^l = f(\beta_j^l \cdot down(y_j^{l-1}) + b_j^l) \tag{2}$$

beta is a constant; $down(\cdot)$ expresses a sub-sampling function, which works as max-pooling or average pooling.

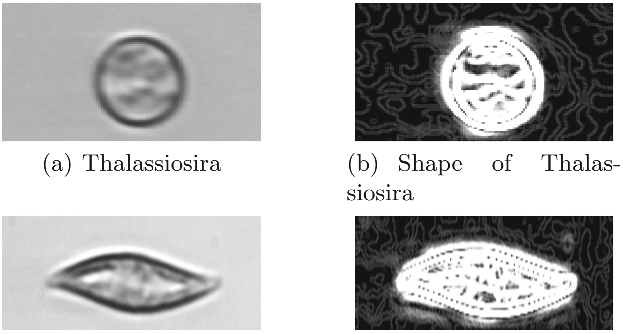
Fully Connected Layer. Neurons in a fully connected layer have full connections to all activations in the previous layer, as seen in regular neural networks. Their activations can hence be computed with a matrix multiplication followed by a bias offset.

$$y_j^l = f(x_j^l) = f(\sum_{i=1}^{N} y_i^{l-1} * w_{ij} + b_j^l) \tag{3}$$

y_j^l is the output value of the j^{th} neuron from l layer; x_j^l is the presynaptic value; N is the total number of input neurons of the previous layer; b_j^l denotes the bias in l layer and $f(x)$ is the activation function.

3 Proposed Method

In order to enhance the plankton image feature extraction problem, we considered 2 different ways to extract plankton features along with original images. Some plankton have significant differences on the shape of cell wall with similar texture of nucleus and cytoplasm, while some of them are similar in the cell wall shape with different internal texture. Our architecture uses multi-sources data that are composed of original images, global feature images and local feature images as the input at the beginning of our architecture for extracting more abstract and representative features. In the first step we extract global and local features from original plankton images for pre-processing. And then we build three sub networks based on AlexNet to load different plankton features respectively. Each sub network has its own input but share the output error (Fig. 3).



(a) Thalassiosira

(b) Shape of Thalassiosira

(c) Pleurosigma

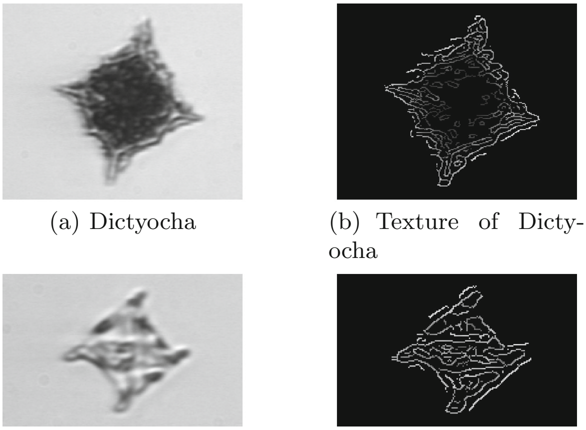
(d) Shape of Pleurosigma

Fig. 1. The two images on the left are similar in texture. It's difficult to distinguish them just depending on texture information. And the internal texture may not work in some plankton classes. And two images on the right are processed into global feature images, which represents the shape information of plankton.

3.1 Feature Extraction

Global Feature. Setae and shape, which can be considered as a kind of global feature, are always used for plankton classification [27]. We develop a method to extract the global features of plankton. We wish this kind of features can describe the appearance of plankton and omit the internal texture.

At first, bilateral filter [28] is used to remove some noise of plankton images and smooth the plankton shape. Then, scharr operator [29], which is optimized by Sobel operator, can mark plankton appearance information accurately. In the end, in order to distinguish global features (including shape and setae information), we increase the contrast value of image to make shape information outstanding and omit the information inside the plankton. And the image after contrast enhancement can be seen to represent the global feature.



(a) Dictyocha

(b) Texture of Dictyocha

(c) Eucampia

(d) Texture of Eucampia

Fig. 2. The two images on the left are similar in shape. It is difficult to classify these two images correctly just based on shape information. And the internal texture can provide valuable information to distinguish them. And two images on the right are detected by canny edge detector, which describe the important internal information of plankton. The extra information can be used to improve accuracy.

Local Feature. The texture of plankton is also a useful feature for classification. Some plankton, such as Fig. 2(a) and (c) which are described in Fig. 2, have similar shape but different transparency. In this case, the internal texture provides a possible solution to distinguish them. Canny edge detector is a multi-stage algorithm to detect a wide range of edges in images [30], which extracts useful texture information from different vision objects and dramatically reduces the amount of data to be processed. It can generate a series of edges by the intensity gradients of the image and tracking edge by hysteresis. Here, we use canny edge detector to extract the internal texture (including nucleus and cytoplasm) of plankton.

3.2 Network Structure

The architecture we propose is shown in Fig. 3. Our network structure is composed of three sub Alex networks: network A for training global feature images; network B for training original images and network C for training local feature images. Global and local images for network A and C are preprocessed. Although this model has three different inputs, these three sub networks share the same label on backpropagation. We used a pyramid fully connected structure after convolutional and pooling layers. Because the images with global and local features have the same size of original images, the size of convolutional and pooling layers in three sub networks are the same. Considering the large gap between global and local features, we combine the inner products of two sub

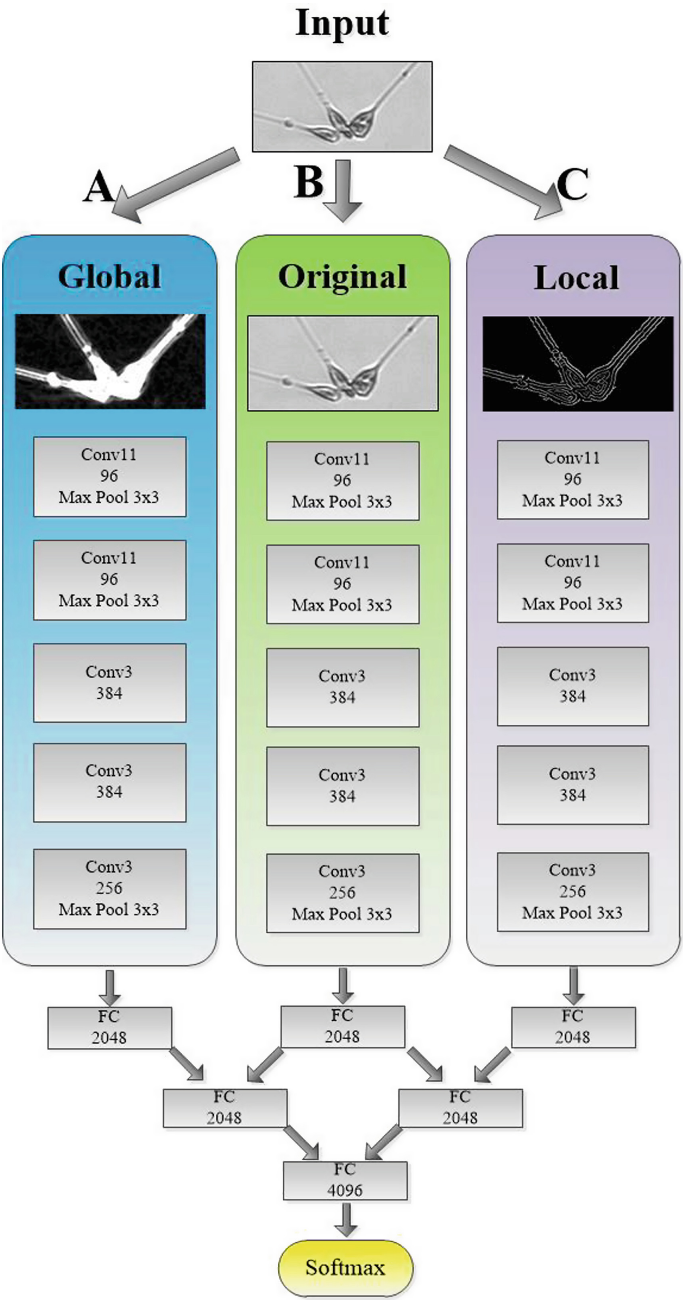


Fig. 3. Proposed architecture for plankton classification. The network A operates on global feature images. The network B is trained on original plankton images, which hold all information of plankton. And the network C operates on local feature images. The hybrid model can explore more detailed and dimensional features based on these features fusion to improve accuracy.

networks respectively (original and local; original and global) and then merge them together. We use the inner products from original images as a bridge to connect global and local features. Please note that there is no direct connection between network A and C before the 8^{th} layer. Unlike the forward equations mentioned in Sect. 2, the forward process from the 6^{th} to 7^{th} fully connected layers can be concluded as the following equations:

$$y_{ab7m} = f(\sum_{i}^{2048} w_{71im} y_{a6i} + \sum_{j}^{2048} w_{72jm} y_{b6j} + b_{ab7}) \tag{4}$$

$$y_{bc7m} = f(\sum_{j}^{2048} w_{73jn} y_{a6j} + \sum_{k}^{2048} w_{74kn} y_{b6k} + b_{ab7}) \tag{5}$$

while y_{ab7m} denotes the output of the m^{th} neuron on the left 7^{th} fully connected layer; y_{bc7n} denotes the output of the m^{th} neuron on the right 7^{th} fully connected layer; w_{71im} is the weight between the i^{th} neuron on the 6^{th} and the m^{th} neuron on the 7^{th} layer; y_{a6i} is the i^{th} neuron output of the 6^{th} layer in network A; y_{b6i} is the j^{th} neuron of the 6^{th} layer in network B; y_{b6k} is the k^{th} neuron of the 6^{th} layer in network C and b_{ab7} is the left 7^{th} layer bias. Similarly, we can deduce the remaining forward path to obtain y_{abc8} which is shown in Fig. 4.

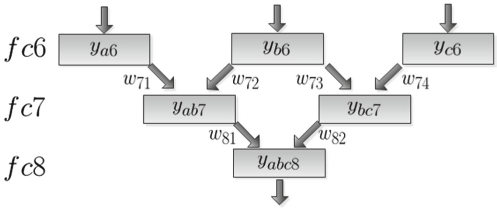


Fig. 4. Detail of pyramid structure in the fully connected layers

The back propagation rule should also follow the proposed pyramid structure. Specially, the error derivative of the 6^{th} layer in network B is calculated as:

$$\delta_{b6j} = f'(x_{b6j})(\delta_{ab7m} w_{72} + \delta_{bc7n} w_{73}) \tag{6}$$

where δ_{b6j} is the error of the j^{th} neuron in the 6^{th} layer of network B and $f'(x)$ is the derivative of the activation function. Please note that only network B will receive the full back propagated error from the last layer. Network A and C only receive a part of error from the 8^{th} layer.

4 Experimental Analysis

4.1 Dataset

The Imaging FlowCytobot (IFCB) at Woods Hole Oceanographic Institution is a system that has been continuously imaging plankton since 2006. They provide WHOI-Plankton: a large scale, fine-grained visual recognition dataset for plankton classification, which comprises over 3.4 million labeled images across 70 classes [6]. Actually, we can obtain over 3 million images across 103 classes from their website. But in such 103 classes dataset, the plankton number of each class is quite unbalanced that one class is over 2 million, some classes are also over 100 thousand and many classes are below 500. And the accuracy on such big 103 classes plankton dataset is unreasonable. Therefore, we collect 30 plankton classes randomly whose images are over 1000 and sample each class with 1000 images for fairly comparison. Here the dataset we use to train and test our architecture contains 30000 images totally in 30 classes. Plankton images are divided into two parts: training set and test set. And training set and test set are sampled randomly. The images for training and test are set as 4:1. And the images in the dataset are shown in Fig. 5. Each sub image in Fig. 5 denotes a kind of plankton. The detail of each class is described in Table 1. And the number of Table 1 is arranged as row-major order in Fig. 5.

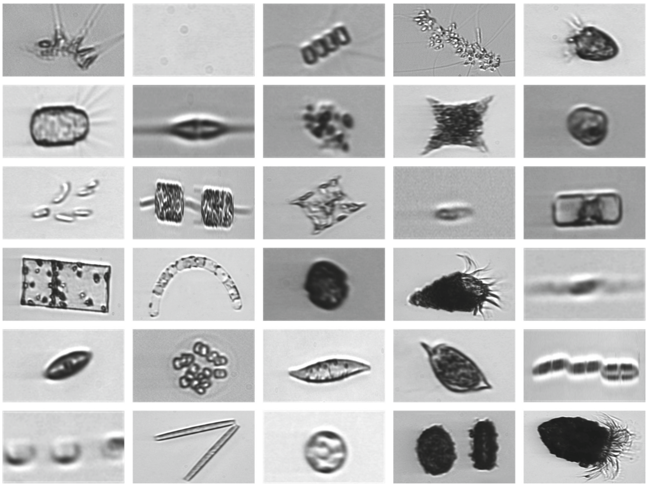


Fig. 5. This image shows that 30 images of 30 classes in dataset respectively. And the size of each images are different, we resize the image and arrange these images for convenience.

Table 1. The table demonstrate that the exact name of plankton classes in dataset. And the number order in table is follow by row-major order in above picture.

Number	Class name	Number	Class name
1	Asterionellopsis	16	Guinardia_flaccida
2	Bad	17	Guinardia_striata
3	Chaetoceros	18	Heterocapsa_triquetra
4	Chaetoceros_flagellate	19	Laboea_strobila
5	Ciliate_mix	20	Leptocylindrus
6	Corethron	21	Pennate
7	Cylindrotheca	22	Phaeocystis
8	Detritus	23	Pleurosigma
9	Dictyocha	24	Prorocentrum
10	Dino30	25	Pseudonitzschia
11	Dinobryon	26	Skeletonema
12	Ditylum	27	Thalassionema
13	Eucampia	28	Thalassiosira
14	Flagellate_sp3	29	Thalassiosira_dirty
15	Guinardia_delicatula	30	Tintinnid

4.2 Experimental Result

To start with, we train AlexNet [21] on the 30 classes dataset, the accuracy of which is set as the baseline. the following experimental results in this part are all based on the baseline.

All these results from single network to multi networks are presented in Table 2. Except training AlexNet on original images, we also have measured the performance of AlexNet on global feature images and local feature images. We can see that the two single network underperform in accuracy comparing with the result of network trained on original images. It is possible that a network only with the global feature images or local feature images may performs poorer than the network with original images.

We also compare the results of three possible architecture with only two sub networks. The two sub networks are merged in the 8^{th} layer. We find that the accuracy is improved a little when we consider both original images and local feature images. We get the best result when we combine all 3 features using the architecture described in Fig. 3. Compared with the baseline, the hybrid architecture has improved more than 1% accuracy.

To prove that our architecture can be extensive to other network structures, we also evaluate it with GoogLeNet [31] based on 30 classes dataset. And the experimental results are shown in Table 3. We have measured the performance of GoogLeNet on original images, global feature images and local feature images

Table 2. The experimental results of AlexNet in 30 plankton classes classification.

Models	Accuracy
AlexNet trained on original images (Baseline)	94.75%
AlexNet trained on global features	94.06%
AlexNet trained on local features	93.09%
Two AlexNets trained on original images and global features	95.32%
Two AlexNets trained on original images and local features	94.50%
Two AlexNets trained on global features and local features	93.33%
Three networks trained on original images, global features and local features	95.83%

Table 3. The experimental results of GoogLeNet in 30 plankton classes classification.

Models	Accuracy
GoogLeNet trained on original images (Baseline)	95.2%
GoogLeNet trained on global features	93.4%
GoogLeNet trained on local features	93.2%
Three GoogLeNets trained on original images, global features and local features	96.3%

respectively. And compared to single GoogLeNet, our hybrid architecture of GoogLeNet also outperforms about 1%.

Our experiments is based on the publicly available deep learning toolbox: Caffe [32]. Four NVIDIA GTX 980Ti 6GB GPUs are used here to implement all above experiments. And we trained our models using stochastic gradient descent with momentum of 0.9 and weight decay of 0.0005. The initialization of the weights in our architecture is set as AlexNet in ImageNet classification [21]. We initialize and train the neuron biases in the 2^{th}, 4^{th} and 5^{th} convolutional layers. And the neuron biases in the remaining layers are set with constant 0. When training AlexNet on original images, we do some image processing by subtracting the mean activity over the training set form each pixel, random mirroring of images and cropping the images into 227×227 randomly. We greedy search the parameters of each architecture to get the best performance.

5 Conclusion

In this paper, we propose a hybrid convolutional neural network to classify 30 classes plankton image set automatically and effectively. We define two different feature extraction operator to calculate the global and local features for plankton classification. We design a pyramid structure to combine original feature, global feature and local feature. Experimental results show that our architecture can extract and classify plankton images more effectively.

The distribution of plankton is highly inhomogeneous. How to classify the plankton effectively on an unbalanced data set is still a challenge. We leave this part in our future work.

Acknowledgement. This work was supported by the National Natural Science Foundation of China under Grant Nos. 61271406, 61301240, and the Fundamental Research Funds for the Central Universities under Grant No. 201562023.

References

1. Lalli, C., Parsons, T.R.: Biological Oceanography: An Introduction. Butterworth-Heinemann, New York (1997)
2. Sardet, C.: Plankton: Wonders of the Drifting World. University of Chicago Press, Chicago (2015)
3. Reynaud, E.G. (ed.): Imaging Marine Life: Macrophotography and Microscopy Approaches for Marine Biology. Wiley, Hoboken (2013)
4. Davis, C.S., Thwaites, F.T., Gallager, S.M., Hu, Q.: A three-axis fast-tow digital video plankton recorder for rapid surveys of plankton taxa and hydrography. Limnolo.Ocean. Methods **3**, 59–74 (2005)
5. Jaffe, J.S., Roberts, P.L.D., Ratelle, D., Laxton, B., Orenstein, E., Carter, M., Hilbern, M.: Scripps plankton camera system (2015)
6. Orenstein, E.C., Beijbom, O., Peacock, E.E., Sosik, H.M.: Whoi-plankton-a large scale fine grained visual recognition benchmark dataset for plankton classification. arXiv preprint arXiv:1510.00745 (2015)
7. Benfield, M.C., Grosjean, P., Culverhouse, P.F., Irigoien, X., Sieracki, M.E., Lopez-Urrutia, A., Dam, H.G., Hu, Q., Davis, C.S., Hansen, A., Pilskaln, C.H., Riseman, E.M., Schultz, H., Utgoff, P.E., Gorsky, G.: RAPID: research on automated plankton identification. Oceanography **20**, 172–187 (2007)
8. MacLeod, N., Benfield, M., Culverhouse, P.: Time to automate identification. Nature **467**, 154–155 (2010)
9. Erickson, J.S., Hashemi, N., Sullivan, J.M., Weidemann, A.D., Ligler, F.S.: In situ phytoplankton analysis: theres plenty of room at the bottom. Anal. Chem. **84**, 839–850 (2011)
10. Samson, S., Hopkins, T., Remsen, A., Langebrake, L., Sutton, T., Patten, J.: A system for high-resolution zooplankton imaging. IEEE J. Ocean. Eng. **26**, 671–676 (2001)
11. Tang, X., Stewart, W.K., Vincent, L., Huang, H., Marra, M., Gallager, S.M., Davis, C.S.: Automatic plankton image recognition. Artif. Intell. Biol. Agric. **12**, 177–199 (1998)
12. Tang, X., Lin, F., Samson, S., Remsen, A.: Binary plankton image classification. IEEE J. Ocean.Eng. **31**, 728–735 (2006)
13. Grosjean, P., Picheral, M., Warembourg, C., Gorsky, G.: Enumeration, measurement, and identification of net zooplankton samples using the zooscan digital imaging system. ICES J. Mar. Sci. J. Conseil **61**, 518–525 (2004)
14. Gorsky, G., Ohman, M.D., Picheral, M., Gasparini, S., Stemmann, L., Romagnan, J.B., Cawood, A., Pesant, S., García-Comas, C., Prejger, F.: Digital zooplankton image analysis using the zooscan integrated system. J. Plankton Res. **32**, 285–303 (2010)

15. Sosik, H.M., Olson, R.J.: Automated taxonomic classification of phytoplankton sampled with imaging-in-flow cytometry. Limnol. Oceanogr. Methods **5**, 204–216 (2007)
16. Buf, H., Bayer, M.M.: Automatic Diatom Identification. World Scientific, Singapore (2002)
17. Culverhouse, P.F., Williams, R., Reguera, B., Herry, V., González-Gil, S.: Do experts make mistakes? A comparison of human and machine identification of dinoflagellates. Mar. Ecol. Prog. Ser. **247**, 17–25 (2003)
18. Khosla, A., Jayadevaprakash, N., Yao, B., Li, F.F.: Novel dataset for fine-grained image categorization: stanford dogs. In: Proceedings of CVPR Workshop on Fine-Grained Visual Categorization (FGVC) (2011)
19. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. **15**, 1929–1958 (2014)
20. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature **521**, 436–444 (2015)
21. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp. 1097–1105 (2012)
22. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8689, pp. 818–833. Springer, Heidelberg (2014). doi:10.1007/978-3-319-10590-1_53
23. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
24. Fukushima, K.: Neocognitron: a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biol. Cybern. **36**, 193–202 (1980)
25. LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L.D.: Backpropagation applied to handwritten zip code recognition. Neural Comput. **1**, 541–551 (1989)
26. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proc. IEEE **86**, 2278–2324 (1998)
27. Zheng, H., Zhao, H., Sun, X., Gao, H., Ji, G.: Automatic setae segmentation from chaetoceros microscopic images. Microsc. Res. Tech. **77**, 684–690 (2014)
28. Tomasi, C., Manduchi, R.: Bilateral filtering for gray and color images. In: 1998 Sixth International Conference on Computer Vision, pp. 839–846. IEEE (1998)
29. Scharr, H.: Optimal operators in digital image processing. Ph.D. thesis (2000)
30. Canny, J.: A computational approach to edge detection. IEEE Trans. Pattern Anal. Mach. Intell. **8**, 679–698 (1986)
31. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–9 (2015)
32. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: Convolutional architecture for fast feature embedding. arXiv preprint arXiv:1408.5093 (2014)