Intelligent Plankton Image Classification with Deep Learning

Abhishek Verma

Department of Computer Science, New Jersey City University
Jersey City, NJ 07305, USA
averma@njcu.edu

Hussein Al-Barazanchi

Department of Computer Science, California State University
Fullerton, CA 92831, USA
hussein_albarazanchi@csu.fullerton.edu

Shawn X. Wang

Department of Computer Science, California State University
Fullerton, CA 92831, USA
xwang@fullerton.edu

Abstract: Plankton are extremely diverse groups of organisms that exist in large water columns. They are sources of food for fishes and many other marine life animals. The plankton distribution is essential for the survival of many ocean lives and plays a critical role in marine ecosystem. In recent years, intelligent image classification systems were developed to study plankton distribution through classification of the plankton images taken by underwater imaging devices. Due to the significant differences in both shapes and sizes of the plankton population, accurate classification poses a daunting challenge. The mixed quality of the collected images adds more difficulty to the task. In this paper, we present an intelligent machine learning system built on convolutional neural networks (CNN) for plankton image classification. Unlike most of the existing image classification algorithms, CNN based systems do not depend on features engineering and they can be efficiently extended to encompass new classes. The experimental results on SIPPER image datasets show that the proposed system achieves higher accuracy compared with the state-of-the-art approaches. The new system is also capable of learning a much larger number of plankton classes.

Keywords: SIPPER Plankton Image; Convolutional Neural Network; Deep Learning; Image Classification.

Biographical notes: Abhishek Verma received his Ph.D. in Computer Science from New Jersey Institute of Technology, NJ, USA. He is presently Associate Professor of Computer Science at New Jersey City University, NJ, USA. His research interests are within the broad area of data science, big data, and machine learning. Deep learning on big datasets such as deep convolutional nets, model ensembling, image/video/speech recognition, natural language processing, financial market analysis, sentiment analysis. Computer vision on big datasets in video and image. Fusing multiple modalities from video/images/text/speech. Data mining, artificial intelligence, and biometrics for surveillance and security.

Hussein Al-Barazanchi received his MS degree in Computer Science from California State University, Fullerton. His research interests are in deep learning, computer vision, model ensembling, and image recognition.

Shawn X. Wang is a Professor of Computer Science at California State University, Fullerton. He holds a BS in Mathematics from Xiamen University, a MS in Computer Science from Fudan University, and a Ph.D. in Computer and Information Science from New Jersey Institute of Technology. His research interests include data mining, big data, cloud computing, bioinformatics, and cybersecurity.

1 Introduction

Plankton exist in large water bodies and form the main source of food for fishes and other marine animals. They are very different in both sizes and shapes due to their huge population and diversity. There are two types of plankton, phytoplankton and zooplankton. While the phytoplankton is the plant type of plankton, the zooplankton is the animal type. In addition to being a source of food, Phytoplankton is also a key player in carbon fixation cycle where they contribute to about half of our planets' fixed carbon. The distribution of the two types of plankton is highly sensitive to the changes in their environment. For instance, their population and distribution can change quickly as a result of changes of the pollution level in their surrounding environment. Therefore, marine scientists study the alterations of plankton population as an early indicator for environmental issues. The investigation of plankton population in terms of their types and numbers can advance our understanding of the marine ecosystem. It may also shed light on other environmental issues.

In the early days, research on plankton distribution relied on collection of plankton samples. The available techniques for plankton sample collection were limited to such tools as towed nets, pumps, or Niskin bottles, etc. An expert is then assigned to analyze and classify the collected samples. These approaches required significant amount of resources and manpower. The invention and advances of underwater imaging systems lead to a variety of alternative ways to study plankton population and distribution. The underwater imaging systems were utilized to capture plankton images. These images were then stored in computers and they were analyzed to study the population and distribution of plankton. Several different underwater imaging systems have been built to collect images of plankton samples. The three most popular systems are Video Plankton Recorder (VPR) Davis et al. (1992), under water holographic camera system (HOLOMAR) Watson et al. (1998), and Shadowed Image Particle Profiling and Evaluation Recorder (SIPPER) Samson et al. (2001). These systems make it possible to collect immense number of images of plankton samples continuously. Even with the advances in sample image collection, the process of sample classification remained manual for several years. Most recently, the fast development in computer vision algorithms and powerful GPUs make it possible to build intelligent image classification systems and automate the process of plankton image classification.

One of the early attempts to automate the classification of plankton images can be traced back to Tang *et al*. Tang et al. (1996). In their experiments, they used samples that were collected by VPR imaging system. Their approach utilized Fourier descriptors with invariant moment features and gray-scale morphological granulomtries. In 2005, Luo *et al*. introduced an approach that achieved an accuracy of 90% on five classes of plankton

images captured by SIPPER system Luo et al. (2004). The system they developed took advantage of active learning with support vector machines. In the following year, Tang *et al*. invented a new algorithm for binary plankton classification based on normalized multilevel dominant eigenvector estimation Tang et al. (2006). The algorithm used shape descriptors and accomplished 91% accuracy. Zhao *et al*. Zhao et al. (2009), in 2009, suggested a system that used random sampling with multiple classifiers. The classification accuracy with this technique was boosted to 93% with seven classes of plankton. In 2014, a new framework called PNDA was developed by Li *et al.* Zhifeng et al. (2014). Their approach reached an accuracy of 95% also with seven classes.

A drawback with all the aforementioned algorithms is that they depended on features engineering. The accuracy relies on the quality of the used features. This is problematic for two reasons. First of all, looking for good features is time consuming. These features are very much dependent on the existing dataset. They are chosen to be optimal representation of the data. Secondly, when new data and classes are added a different set of features must be decided to integrate the new classes into the system.

Most recently, Lee *et al.* proposed an approach using convolutional neural network Lee et al. (2016). Their approach worked on a large number of samples. It was learned that the imbalance of the numbers of samples among different classes caused decline in classification accuracy. In order to deal with this obstacle their algorithm employed some specific tactics that are not easy to be generalized. Their experiments achieved an accuracy of 94.7% for five classes of plankton. All these works only classified five to seven classes of plankton.

In this paper, we propose an intelligent classification system that is built based on convolutional neural networks as the building block. Our experimental results using the SIPPER datasets demonstrate improvement in classification accuracy compared with the state of the art approaches.With the same seven types of plankton our design improved the classification accuracy to 98.2%. In addition, the proposed system was extended to a lot more classes of plankton. To test the robustness of our design, we also conducted two phases of experiments with different level of imbalance among the datasets. The experimental results indicated that our design is not very sensitive to such imbalance.

The rest of this paper is organized as follows. In section II, we provide a description of the SIPPER image datasets used in our experiments. In section III, we discuss the details of the proposed system. We present the experimental setup and results in section IV. Section V concludes the paper and suggests some future improvements.

2 Plankton Image Datasets

The plankton images that we used in this study are provided by the University of South Florida (Tampa, FL, USA) Kramer (2010). Those images are obtained using version 3 of SIPPER system which capture 3-bit image resolution so the images are of low quality. All the samples were gathered during the time period between 2010 and 2014 from the Gulf of Mexico. A quick analysis of the entire dataset revealed two main issues. The first issue is that due to the low resolution used by the SIPPER system the quality of the images is low and at the same time the noise level is high. The second issue is the high level of variations within the same plankton class accompanied with similar appearance between different plankton classes. Additional challenges include deformation and occlusion of plankton objects. The entire dataset consists of more than 750 thousands samples. These samples are identified

into 81 different classes by the marine scientists at USF. To compare our approach with the state of the art approaches from most recent studies Lee et al. (2016); Luo et al. (2004); Tang et al. (2006); Zhao et al. (2009); Zhifeng et al. (2014), we first selected only seven types of plankton for Phase one of our experiments. Those seven classes were the same classes used in previous research with same distribution per plankton class. The details of these classes with their population are provided in Table I. Figure 1 shows randomly selected samples from this set.

Table 1 Plankton Types and Their Distribution

.	**#Sample**	**Average Width**	**Average Height**
Acantharia	131	70	46
Calanoid	172	346	319
Chaetognath	450	160	273
Doliolid	485	68	45
Larvacean	529	61	43
Radiolaria	563	110	69
Trichodesmium	789	107	97

Since there are many classes of plankton in the entire dataset and the numbers of samples in each class are very different, in the next two phases of experiments we selected classes with different sample sizes to study the performance of our approach in terms of classification accuracy. In phase two, we selected the classes that have more than one thousand samples per class and we limited the max number of images per class to be two thousands. The reason is to prevent the network from overfitting on classes with more images and not learning enough on the classes with smaller numbers of images. The result of this selection is a dataset with 52 classes. In phase three, we selected the classes with a minimum number of 10 images per class and the max number of images is set to two thousand images. This selection resulted in a dataset of 77 classes. Experiments on these two datasets will provide more accurate assessment of the feasibility of such system to a real world application. They will also reveal how imbalance among the class data sizes affect the classification accuracy.

3 Convolutional Neural Networks

A typical algorithm based on computer vision starts with extracting distinguishable features in the dataset. Ideally, those features are best representation of the characteristics of the data. In the meantime, they are invariant within the same type of data. Most of the existing approaches for feature extraction rely on engineering hand designed features for the targeting objects. This process is difficult, time consuming, and laborious. Since the selected features very much depend on the data, it is difficult to scale them up for new classes. In other words, a different set of features need to be selected when new classes are added to the dataset. The other line of methods is to let the algorithm learn the features automatically. These methods are more efficient. For image classification, CNN is proved to be the most effective and promising method within this catagory of learning methods.

CNN was discovered in the work of Hubel *et al*. Hubel and Wiesel (1968) on catsâŁ™ visual cortex. Fukushima was the first to build a simulation of this mechanism in his system Neocognitron Fukushima (1980). LeCun et al. LeCun et al. (1998) developed a

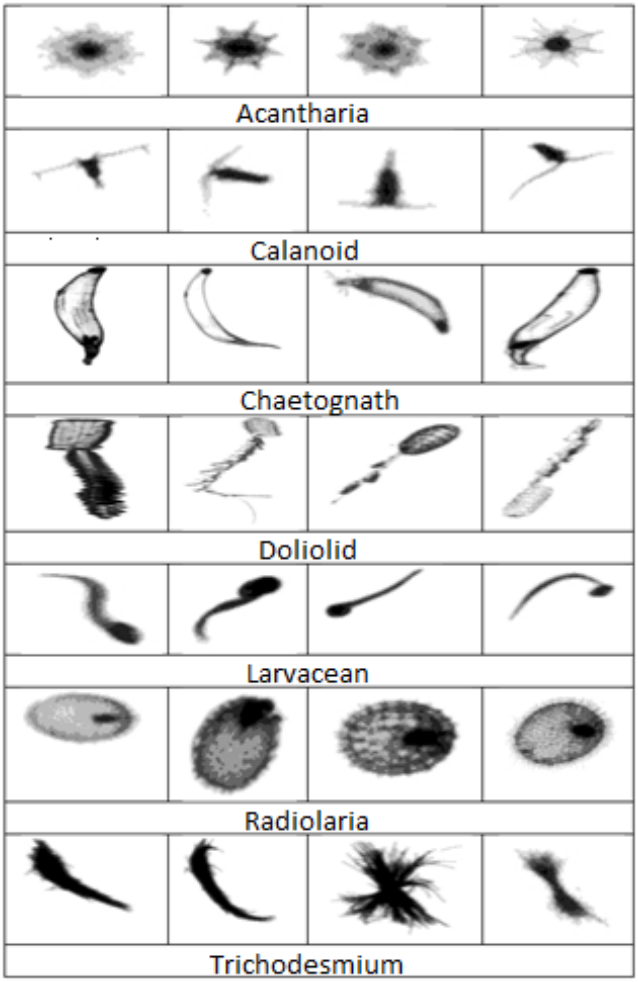

Figure 1 Random Samples from the Seven Classes

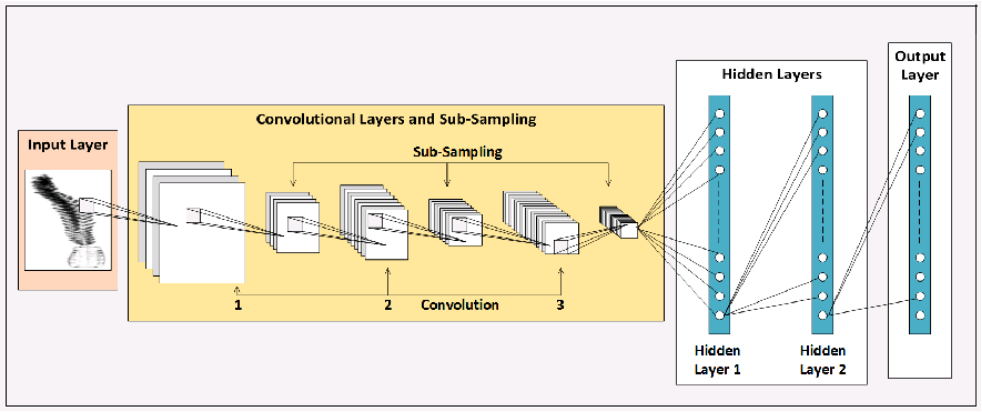

Figure 2 The CNN Architecture

CNN architecture and used it for a handwritten recognition system. The popularity of CNN remained limited due to its high computational complexity until the introduction of GPUs. The advances in GPUs and the successful application of CNN in Imagenet competition Krizhevsky et al. (2012) boosted its popularity.

Convolutional neural networks based system consists of two main parts. The typical design of the CNN is shown in Figure 2. The first part is the convolutional layers. These layers work as distinguishable feature extractor. Typically, a convolutional layer is further divided into three components, namely filter layer, non-linearity, and sub-sampling Krizhevsky et al. (2012); LeCun et al. (1998, 2010). The filter layer is a specific form of fully connected neural network Boureau et al. (2010); Van (2014). Neurons in the filter layer are sparsely connected to neurons in the next layer and they follow a topographical layout. Those connections are based on the related areas in the visual context. Images are fed to the filter layer in the format described in the equation (1). The symbols h, w and c refer to the height, width and number of channels respectively. Equation (2) shows how the filter layer compute the output LeCun et al. (2010).

$$h \times w \times c \tag{1}$$

$$y_j = b_j + \sum_i K_{ij} \times x_i \tag{2}$$

The non-linearity layer consists of activation function. The common activation functions are the tanh() and sigmoid() functions Krizhevsky et al. (2012); LeCun et al. (2010). A problem with these activation functions is their slow convergence rate when used with gradient descent. Non-Saturating activation are faster Krizhevsky et al. (2012); LeCun et al. (2010). Equations (3) and (4) show the sigmoid and tanh functions. For faster convergence we used Rectified Linear Units (ReLUs) which is shown in equation (5).

$$sigmoid(x) = \frac{1}{(1 + e^{-x})} \tag{3}$$

$$tanh(x) = \frac{(e^{2x} - 1)}{e^{2x} + 1)} \tag{4}$$

$$f(x) = max(0, x) \tag{5}$$

The sub-sampling layer is composed of pooling. Pooling is a technique for dimensionality reduction Boureau et al. (2010) by removing unrelated information Simonyan et al. (2014). The output of this layer is the reduced version of the input Van (2014). A problem that arises from filter layer's features is that they are not translation invariant. The pooling is used to help the system to be immune to translation in the input. Pooling helps to reduce the sensitivity of activations in neural network induced by the pixels' locations and the neural network structure Srivastava et al. (2014). The common functions used in pooling are maximum and average functions and they are usually named max-pooling and average-pooling. There are two different ways to feed the input to those functions, namely the separate or overlapping mode Van (2014). The purpose of using multiple layers of convolution is to enable the system to learn the feature hierarchies. In the first layer, the system learns the low level features which are pixel intensities. The middle

layer learns the objects' edges which are mid-level features. The system learns the high level features for example the objects themselves in the final layer of convolution.

The second major part of CNN based systems is the classifier. Fully connected neural networks (hidden layers) are used in this part as shown in equation (6). To regularize the hidden layers, a dropout layer is added after each hidden layer. Dropout is a recent technique developed by Srivastava *et al*. Nair et al. (2010). The purpose of the dropout layer is to reduce the problem of overfitting and enhance generalization on the test data. This method works by removing random neurons with their connections during the process of learning.

The last layer in the system is the output layer. The output layer assigns the sample to a specific class according to the outputs of the previous layers and some computation. To compute the classification outputs in the system, the softmax function is used as shown in equation (7). The two components convolutional layers and fully connected layers works jointly to form the leanring system. The number of convolutional layers and dropout layers used in the system depends on the designer of the system, but more layers mean more computations and likely higher accuracy.

$$Y(x) = f((\sum_i W_i \times x_i) + b) \tag{6}$$

$$P_j = \frac{exp(x_j)}{(\sum_k exp(x_k))} \tag{7}$$

We used backpropagation algorithm for learning and stochastic gradient descent for optimization. We set the batch size to 128 images for training and 64 images for validation and testing. Initially the momentum and learning rate are set to 0.9 and 0.01 respectively. We set the number of epochs to 150 epoch. Learning rate is updated each 25% of epochs to be half of the old value.

Table 2 Division of Data Sets

Dataset	Training	Validation	Testing
Sipper-7	2182	466	471
Sipper-52	70352	15074	15077
Sipper-77	74680	16009	16002

4 Experimental Setup and Results

To compare the performance of our design with the previous research and also evaluate how the imbalance of class sizes affect the accuracy, we conducted three phases of experiments using different sets of data selected from the dataset obtained using SIPPER system from the University of South Florida. In the first phase, we took only seven classes for the purpose of comparing the results with previous studies and we called this set sipper-7. The second phase composed of classes with more than 1,000 samples per class and up to 2,000 samples which resulted in 52 classes and we named it sipper-52. In other words, the second phase

selected classes with samples between 1,000 and 2,000. The last phase is composite of classes with more than 10 images per class which resulted in 77 classes and we called it sipper-77. In each phase, we divided the dataset to 70% training, 15% validation, and 15% testing. Table II shows the details of this division.

The input to the convolutional neural networks should be of fixed dimension. However, the dimensions of images in the sipper dataset are very different. As can be seen from Table I for sipper-7, the average height of all images in sipper-7 is 138 and the average width is 142 pixels. Actually, in sipper-7, the minimum height is 43 pixels while the maximum height is 319 pixels. Meanwhile, the minimum width is 61 pixels and the maximum width is 346 pixels, respectively. Similarly, in sipper-52 and sipper-77 the dimensions of the images are very different. To ensure a fixed dimension of input to CNN, we have to resize the images in the dataset. There are two ways of resizing images. The first way is to resize the images in regard to aspect ratio while the second technique is without regard to the aspect ratio. We chose to do the resizing without regard to the aspect ratio. The dimensions were set to 256*256 for height and width.

Table 3 CNN Architecture

Layer	Size	Stride
Input Image	256 * 256	
Cropping	222 * 222	
Convolution 1	64 Kernel (3 * 3)	1
Pool 1	3 * 3	2
Convolution 2	64 Kernel (3 * 3)	1
Pool 2	3 * 3	2
Convolution 3	128 Kernel (3 * 3)	1
Pool 3	3 * 3	2
Convolution 4	128 Kernel (3 * 3)	1
Pool 4	3 * 3	2
Convolution 5	256 Kernel (3 * 3)	1
Pool 5	3 * 3	2
Hidden Layer 1	256	
Dropout	20%	
Hidden Layer 2	256	
Dropout	20%	
Output Layer	7, 52, or 77	

We utilized the common configuration of convolutional neural networks that is filter layer trailed by non-linearity followed by max pooling layer. This CNN architecture follows the general design guides of VGG Net Simonyan et al. (2014). Table III shows the details of the architecture. In front of the CNN is the input layer which receives images as an input. The input layer is followed by 5 convolutional layers. All reception fields for convolutional layers through the network are of size 3*3. We set the number of the fully connected layers to two hidden layers. The number of neurons in each hidden layer is set to 256 neurons. To alleviate the problem of overfitting, a dropout layer Srivastava et al. (2014) is attached to each fully connected layer with 20% dropout ratio. ReLU activation function Nair et al.

(2010) is used for all the neurons in the convolutional and dense layers. The input to the output layer is the output from the second hidden layer. Output layer size was set to 7, 52, or 77 depending on which sipper dataset were used. The details of initializing the weights and biases of the network are shown in Table IV.

Table 4 CNN Architecture

Layer	Weight		Bias	
	Initialization	STD	Initialization	STD
Convolution 1 - 5	xavier	0.1	constant	0.2
Hidden Layer 1 - 2	gaussian	0.01	constant	1
Output Layer	gaussian	0.01	constant	1

We used the architecture in Table III and Table IV for all 3 sipper datasets. It gave very good results for sipper-7 while the results for sipper-52 and sipper-77 were not as good as sipper-7. The reason is for the low quality of the images and also similarity between different classes and variations within the same class. In all experiments, we did not use data augmentation. Table V shows the results of the experiments. As can be seen in the table, the accuracy for sipper-52 is slightly better than that for sipper-77. This is expected because of the imbalance in the numbers of images for the classes. Nevertheless, it is worth noticing that the difference in accuracy is not significant, indicating certain level of robustness of our design.

Table 5 Experimental Results

Dataset	Epoch	Validation	Testing
Sipper-7	136	98.43 %	98.20 %
Sipper-52	150	81.95 %	81.79 %
Sipper-77	148	80.72 %	80.54 %

5 Conclusion and Future Work

Efficient analysis and classification of huge amounts of plankton data requires robust algorithms. In this paper, we presented a learning system based on convolutional neural networks. We conducted comprehensive experiments and in depth comparison of the performance analysis of CNN across 3 plankton image datasets. Results of our experiments using the SIPPER dataset show improvement in classification accuracy in comparison to the previous approaches from other research groups. One major advantage of the CNN is scalability for classification of new classes without the need for features engineering. In the future we plan to extend our work by including Hybrid Convolutional Neural Networks and compare accuracy with CNN we used in this paper. Another area of potential research is the multi-column CNN feature fusion methodology. Additionally, we plan to perform comparative assessment on multiple large scale color image datasets.

References

Al-Barazanchi, H. and Verma, A. and Wang, S. (2015) 'Performance Evaluation of Hybrid CNN for SIPPER Plankton Image Classification', *The Proc. of the IEEE Third International Conference on Image Information Processing*, pp. 551–556, December 2015, Himachal Pradesh, India

Boureau, Y.-L. and Ponce, J. and LeCun, Y. (2010) 'A theoretical analysis of feature pooling in visual recognition', *In Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pp. 111–118, 2010.

Davis, S. and Gallager, S. M. and Solow, A. R. (1992) 'Microaggregations of oceanic plankton observed by towed video microscopy', *Science*, vol. 257, pp. 230–232, Jul. 1992.

Fukushima, K. (1980) 'Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position', *Biological Cybernetics*, 36, pp. 193–202.

Hubel, D. and Wiesel, T. (1968) 'Receptive fields and functional architecture of monkey striate cortex', *Journal of Physiology (London)*, 195, pp. 215–243.

Jia, Y. and Shelhamer, E. and Donahue, J. (2014) 'Caffe: Convolutional architecture for fast feature embedding', *arXiv preprint, arXiv:1408.5093*, 2014.

Kramer, K. (2010) 'System for Identifying Plankton from the SIPPER Instrument Platform', *Ph.D. dissertation, University of South Florida*, 2010.

Krizhevsky, A. and Sutskever, I. and Hinton, G. (2012) 'ImageNet Classification with Deep Convolutional Neural Networks', *Proc. Neural Information and Processing Systems*, 2012.

LeCun, Y. and Bottou, L. and Bengio, Y. and Haffner, P. (1998) 'Gradient-based learning applied to document recognition', *Proceedings of the IEEE*, 86(11), pp. 2278–2324.

LeCun, Y. and Kavukcuoglu, K. and Farabet, C. (2010) 'Convolutional networks and applications in vision', *In Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium*, pp. 253–256.

Lee, H. and Park, M. and Kim, J. (2016) 'Plankton classification on imbalanced large scale database via convolutional neural networks with transfer learning', *The Proc of 2016 IEEE International Conference on Image Processing*, pp. 3713–3717, September 2016, Phoenix, Arizona, USA.

Luo, T. and Kramer, K. and Samson, S. and Remsen, A. and Goldgof, D. B. and Hall, L. O. and Hopkins, T. (2004) 'Active learning to recognize multiple types of plankton', *In Proceedings of the 17th International Conference on Pattern Recognition*, 2004, vol. 3, pp. 478–481.

Nair, V. and Hinton, G. E. (2010) 'Rectified linear units improve restricted boltzmann machines', *In Proc. 27th International Conference on Machine Learning*, 2010.

Oquab, M. and Bottou, L. and Laptev, I. and Sivic, J. (2014) 'Learning and transferring mid-level image representations using convolutional neural networks', *in Computer Vision and Pattern Recognition (CVPR)*, 2014 IEEE Conference on. IEEE, 2014, pp. 1717–1724.

Orenstein, E. C. and Beijbom, O. and Peacock, E. E. and Sosik, H. M. (2015) 'Whoi-plankton- A large scale fine grained visual recognition benchmark dataset for plankton classification', *CoRR, vol. abs/1510.00745*, 2015.

Samson, S. and Hopkins, T. and Remsen, A. and Langebrake, L. and Sutton, T. and Patten, J. (2001) 'A system for high-resolution zooplankton imaging', *IEEE J. Ocean. Eng.*, vol. 26, no. 4, pp. 671–676, Oct. 2001.

Simonyan, K. and Zisserman, A. (2014) 'Very deep convolutional networks for large-scale image recognition', *arXiv:1409.1556*, 2014.

Srivastava, N. and Hinton, G. and Krizhevsky, A. (2014) 'Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research', pp. 1929–1958*, 2014.

Tang, X. and Lin, F. and Samson, S. and Remsen, A. (2006) 'Binary plankton image classification', *IEEE J. Ocean. Eng.*, vol. 31, no. 3, pp. 728–735, Jul. 2006.

Tang, X. and Stewart, W. K. (1996) 'Plankton image classification using novel parallel-training learning vector quantization network', *OCEANS'96. MTS/IEEE. Prospects for the 21st Century. Conference Proceedings*. Vol. 3, 1996.

J. Van, (2014) 'Analysis of Deep Convolutional Neural Network Architectures', 2014. *Retrieved from http://referaat.cs.utwente.nl/conference/21/paper/7438/analysis-of-deep-convolutional-neural-network-architectures.pdf.*

Watson, J. and Craig, G. and Chalvidan, V. (1998) 'High resolution in situ holographic recording and analysis of marine organisms and particles (Holomar)', *In Proc. IEEE Int. Conf. OCEANS*, 1998, pp. 1599–1604.

Zhao, F. and Lin, F. and Seah, H. (2009) 'Bagging based plankton image classification', *in Proc. IEEE Int. Conf. on Image Process.*, 2009, pp. 2517–2520.

Zhao, F. and Lin, F. and Seah, H. S. (2010) 'Binary SIPPER plankton image classification using random subspace', *Neurocomputing*, vol. 73, no. 102, pp. 1853–1860, 2010, Subspace Learning / Selected papers from the European Symposium on Time Series Prediction.

Zhifeng, L. and Feng, Z. and Jianzhuang, and Yu, L. Q. (2014) 'Pairwise Nonparametric Discriminant Analysis for Binary Plankton Image Recognition', *IEEE J. Ocean. Eng.*, vol. 39, no. 4, pp. 695–701, 2014.