Plankton Classification

Bolun Yan, Shijia Shu, Zhichen Zhang, Jordan Prazak
University of California, San Diego
Electrical and Computer Engineering
{boy010, s3shu, zhz087, jprazak}@eng.ucsd.edu

Abstract

This project uses ResNet and DenseNet to classify plankton species. This module would be beneficial to determine the health level of an ocean area. The final test accuracies for ResNet and DenseNet are 76.2% and 86.21%. Plans are in place to improve our classification model by adding more high-quality and recognizable images to each class, applying weighted loss to overcome imbalances in the dataset, and tuning the hyper-parameters of DenseNet and ResNet.

1. Introduction

Classifying plankton species based on image data with an algorithm is a very useful tool for estimating plankton populations in an ecosystem. Plankton are incredibly small and populous in Ocean ecosystems. As a result, estimation of plankton populations by hand counting or hand analysis is impractical and ineffective. Using an algorithm, plankton populations can be instead estimated by sampling a large volume of image data from undersea cameras. Plankton populations are an important indicator of Ocean health due to their importance in the food chain, carbon cycle, CO2/O2 balance of Ocean ecosystems. Data about the presence and density of plankton species can be key factors to determine the health level of an ocean area. It could even be used as a feature set to classify and label healthy and unhealthy ocean areas using a separate algorithm.

2. Objective

The algorithms presented in this paper take as input gray scale images of a variety of plankton species as well as a few other undersea objects. The algorithms output a class label that to which species of plankton or type of undersea object was observed. In all there are 121 classes that an input image can be classified into.

3. Related Work

An early approach by Hu and Davis uses a dual classifier approach in [11]. A neural network classifier is used on a shape based feature set and a support vector machine classifier is used on a texture based feature set. A classification is only made if the two algorithms agree. Since this is an earlier paper, the neural network is simpler than what is available now. The dual classification method helps improve on the limited accuracy of the two algorithms independently. The approach of this paper and other modern projects have more computing resources and a better understanding of neural networks at our disposal. As a result these more modern papers rely solely on the success of just a neural network based classifier. A similar neural network approach to this paper is used by Al-Barazanchi et. al in [12]. This approach uses a convolutional neural network and achieves a large amount of accuracy, but only does os on 7 different plankton classes. This success is therfore limited compared to the approach presented by this paper. A team of researchers from Ghent university participated in the National Data Science Bowl contest that produced our dataset and porduced a very successful solution. This group improved their solution over seveeral months. They used complex data augmentation methods as well as a technique they refer to as 'cyclic pooling' within the network itself. [10]

4. Dataset

The National Data Science Bowl dataset was provided by the Oregon State Universitys Hatfield Marine Science Center [2], which contains approximate 33k labeled grayscale images of 121 different plankton classes. There are many different species, ranging from the smallest single-celled protists to copepods, larval fish, and larger jellies. Each raw image was run through an automatic process to extract regions of interest, resulting in smaller images that contain a single organism/entity. As a result, the size of images have a range of 50 x 30 pixels to 150 x 150 pixels.

An observation of the dataset shows that hand-drawing has two main features. Firstly, the image quality of each class varies. Some classes have clear and recognizable plankton images while others are noisy and ambiguous. Secondly, some classes have very little variation.

The main reason for the variation in the image quality is the method by which the images were captured. The raw images were captured by a towed, underwater imaging system using shadow-graph imagery with a line-scan camera. A high-resolution continuous image is parsed into 2048 x 2048 pixel frames, which were corrected using flat-fielding. The frames were thresholded and segmented into regions of interest (ROIs). These ROI segments are the images of the dataset. As a result, the images spanning the gamut from blurry to clear, and tiny to big. Figure 1 shows this feature.

Figure 1. Image Size Comparison

In addition, the relationships between the classes are not identical. Some of the relationships are taxonomic, some are behavioral / ecological, and some are based on shape.The plankton relationships document shows the exact relationships of different categories in Figure 3. Red boxes indicate the main categories, blue boxes indicate major groups, solid lines indicate direct relationships, and dashed lines indicate minor relationships or shape similarities. For the categories with minor relationships, the images appear similar. The Figure 2 shows the images of two similar classes with direct relationships.

Figure 2. bipinnaria vs. brachiolaria

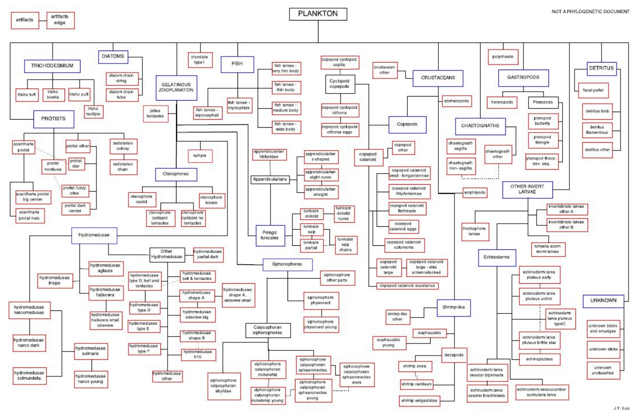


Figure 3. Image Size Comparison

4.1. Data Prepossess

Due to the variety of the image sizes, all images were resized to 128 x 128. This specific size keep the features of images and ensure the performance of our deep learning models.

For classes that have fewer images ($\leq$ 100 images), we augmented the number of images by flip, rotate and Gaussian filter. The amount of images could be enlarged by 10 times.

To avoid over-fitting and better evaluating the model performance. The subset was splitted as 80 percent for training, 10 percent for validation and 10 percent for testing.

5. Method

In this project, our group explored several neural networks to perform the classification task. In this section we will introduce the methodology of ResNet and DenseNet Network and why DenseNet outperform the ResNet in the Plankton Classification task.

5.1. ResNet

ResNet [4] is an artificial neural network with 152 layers in total. Figure 4 shows a short version of ResNet. The first layer of ResNet is a 7*7 convolutional layer and all other layers are 3*3 convolutional layers. There are two pooling layers in total in the architecture.

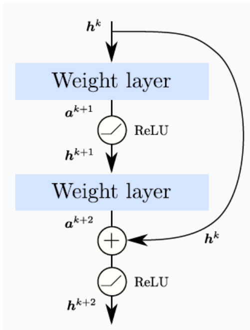


Figure 4. ResNet Architecture (The Residual Block)

The residual layers are equipped with shortcut connections, which can learn how to change the input instead of learning what the output should be. Since the identity mappings are stacked, the deep model should not produce a training error greater than the shallower counterpart. In other words, skipping layers should not degrade the performance. The above image demonstrates the shortcut connection mechanism. In our use case, another advantage is that the skipping connection will help to reduce the effect of gradient vanishing. This is an issue that arises in neural networks with many layers like the one being used in this paper.

5.2. DenseNet

Traditional convolutional feed-forward neural networks connect the output of the l^{th} layer as input to the $(l+1)^{\text{th}}$ layer, which gives rise to the following layer transition: $x_l = H_l(x_{l\text{-}1})$ [5]. ResNets add a skip-connection that bypasses the non-linear transformations with an identity function, which makes the gradient flow directly through the identity function from later layers to the earlier layers. However, the identity function and the output of H_l are combined by summation, which may impede the information flow in the network. In order to improve the information flow between the layers, the DenseNet is constructed by direct connections from any layer to all subsequent layers. This kind of connection is refered to as dense connectivity. The Figure 5 below shows the schematics of a regularly connected network, a network with a single residual layer, and densely connected network.

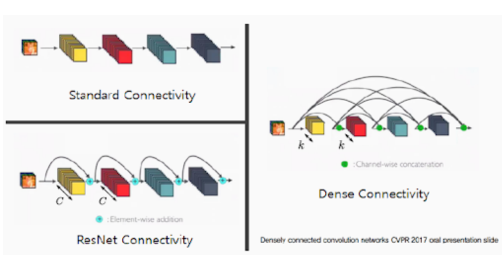


Figure 5. Standard Connectivity vs. ResNet Connectivity vs. Dense Connectivity

As shown in Figure 5 the l^{th} layer receives the feature-maps of all preceding layers, $x_0, ..., x_{l\text{-}1}$, as input: $x_l = H_l([x_0, x_1, ..., x_{l\text{-}1}])$, where $[x_0, x_1, ..., x_{l\text{-}1}]$ refers to the concatenation of the feature-maps produced in layers 0, . . . , $(l-1)$. Because of its dense connectivity, we refer to this network architecture as Densely Connected Convolutional Network (DenseNet) [6]. The DenseNet has totally 201 layers but only 20 million parameters. The DenseNet achieved deeper network with slim layers. The computational time was much faster than ResNet.

5.2.1 Loss Function

Since the ResNet and DenseNet are directly performing the classification task for each input, the loss functions of these two networks are both set to be cross entropy, which is defined in the Equation 1.

$$\begin{aligned}\mathcal{L}(x, y) &= l_n \\ &= -w_n \left[y_n \cdot \log x_n + (1 - y_n) \cdot \log(1 - x_n)\right]\end{aligned} \tag{1}$$

6. Implementation details

Several hyper parameters are selected to use in the models.

In ResNet and DenseNet, we used ResNet34 [9] and DenseNet169 [8] as our base network in each network and changed the last fully connected layer to a 121-class classifier to suit our output. The batch size for each model is 64 to make the GPU fully functional during training. The Adam optimizer was chosen to be used as optimizer for both the generator and the discriminator due to the ability of the Adam optimizer to adjust the learning rate in an efficient manner. The initial learning rate of both models is 0.0001 and the weight decay is 0.001 in order to avoid over-fitting. The weights in the Cross-Entropy loss are 1 because of the similar data distribution of each class.

7. Experiments

7.1. Evaluation Criteria

To evaluate the performance of the networks, the trained networks ran a forward pass on the test data and the accuracy was computed. Both the overall accuracy and the accuracy of each class were considered. The definition of accuracy is Equation 2

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}} \tag{2}$$

In this project, a sample is treated as correctly classified if the predicting result matches the true class label. Therefore, the overall accuracy is the percentage of images that are correctly classified among all samples. The accuracy of each class is the percentage of images that are correctly classified that belong to a specific class.

The model was evaluated on the test set, which was 10 percent of the total supplied training data. The prediction accuracy for each class is shown in the confusion matrix. The vertical axis is the true label of the classes, and the horizontal axis is the predicting result. The color of the square shows the proportion of images being classified correctly. The brighter the color, the better prediction results for the corresponding classes.

7.2. Testing Performance on the Full dataset

The test accuracy on all 121 classes is shown in Figure 6. The prediction accuracy varies relatively widely among the classes. For example, The Artifacts, hydromedusae and ctenophore achieved almost one-hundred percent accuracy. However, some classes like fish_larva_very and fish_larva_deep obtain zero percent accuracy. From the inspection on the data fed into the model, it was observed that classes with fewer training samples are frequently the ones with the lowest classification accuracy . This led to the decision to use data set augmentation.

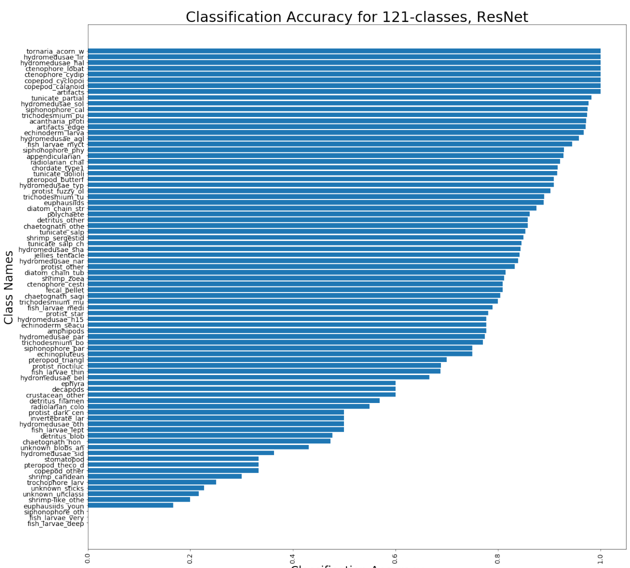


Figure 6. Classification Accuracy on 121-classes

7.3. Improvement with Data Augmentation

The figure on the bottom left shows the top five classes with lowest classification accuracy and to which classes they are confused with. All the images with poor classification accuracy suffer from having high visual similarity to confused classes and a low number of training images. Data augmentation was applied on all classes with less than 100 images in the dataset to improve the result in classes with low accuracy.

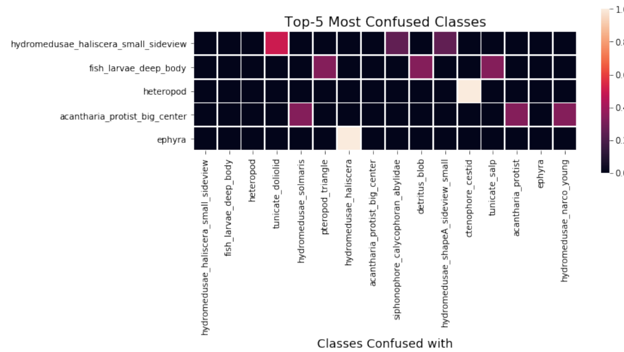


Figure 7. Confusion matrix for classes with lowest test accuracy

With the help of the augmented dataset, the results for the above top five classes improved markedly, from almost 0% accuracy to around 70% accuracy. By data augmentation, the accuracy of each class was successfully balanced to around the same level while keeping the overall accuracy unchanged.

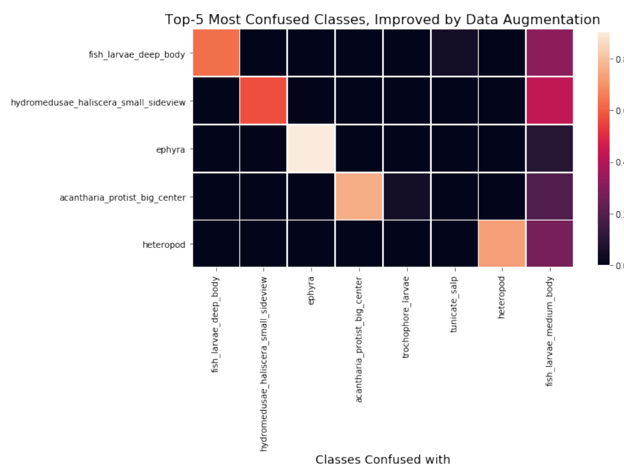
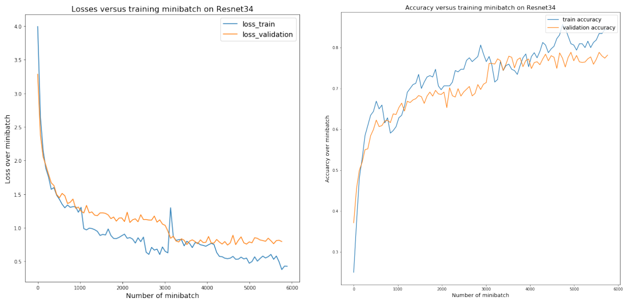


Figure 8. Confusion matrix for improved class classification accuracy, Augmented dataset

7.4. ResNet

The model performed well and the loss dropped rapidly and converged, as is shown in Figure 9. The model was trained with the augmented dataset. The training accuracy is 82.4 percent and training loss is 0.574. The validation accuracy is 76.5 percent and validation loss is 0.893. The overall accuracy on the test set was 76.2 percent.



(a) Loss (b) Accuracy

Figure 9. Training and validation loss and accuracy curve of ResNet

The prediction performance on the test set is shown in the confusion matrix. It can be seen from Figure 10 that some classes achieved around hundred percent classification accuracy.

1	artifacts
2	chordate_type1
3	echinoderm_larva_seastar_brachiolaria
4	copepod_cyclopoid_oithona_eggs
5	protist_fuzzy_olive
6	tornaria_acorn_worm_larvae
7	hydromedusae_typeD_bell_and_tentacles
8	acantharia_protist
9	tunicate_partial
10	hydromedusae_haliscera
11	All others

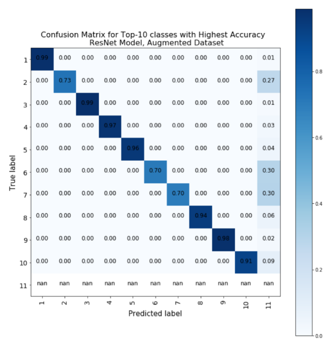
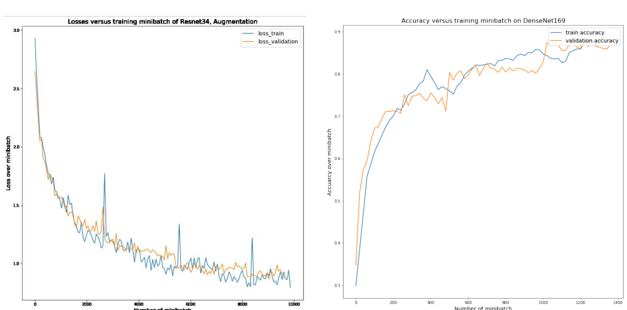


Figure 10. Confusion matrix of ResNet, Top-10 classes

7.5. DenseNet

DenseNet was trained all 121 classes from the training set and the results are shown in Figure 11. The final training accuracy was 87.24 percent and final training loss was 0.308. The final validation accuracy was 86.57 percent and final validation loss was 0.3706. The overall accuracy on test set was 86.21 percent.



(a) Loss (b) Accuracy

Figure 11. Training and validation loss and accuracy curve of DenseNet

8. Discussion

Several interesting observations were found during analysis of the performance. From the preliminary test results, the overall accuracy for ResNet and DenseNet were both adequately high, with accuracy around 73% and 83%. The DenseNet network apparently outperformed the ResNet for this specific classification task. From the above confusion matrix for both network, the top-10 most accurately classified class are consistent across the two networks and these classes all contain much more images in comparison with the classes with poor performance. The data augmentation successfully helped the networks to improve the overall performance on classification, especially for the ones with fewer training samples in the original dataset. With the augmented dataset, the final classification accuracy achieved was around 76% and 86% accuracy for the ResNet and DenseNet model. However, after visualizing the classification accuracy for all classes, a few classes which originally performed well before data augmentation had worse accuracy after the data augmentation. This indicates that the classification accuracy for some classes suffered despite the overall improvement in classification accuracy.

9. Future Work

We will do four steps to improve our classification model in the future. First, we will add higher quality and clearer images to each class, especially for the classes that had few images. This is because the current dataset is too noisy and imbalanced. Second, we will apply weighted loss to overcome any remaining imbalances in the dataset. Third, we will take more time to tune the hyper parameters of DenseNet and ResNet, like Learning Rate; image size; and mini-batch size. Tuning these hyper parameters will result in incremental improvements in classification accuracy. Finally, we will utilize different varieties of DenseNet that have more convolutional layers. Applying all of these engancements should provide modest improvements in classification accuracy.

10. Contributions

Bolun and Shijia primarily worked on building the ResNet and DenseNet models using PyTorch, performing the training on the network and evaluating the performance of the classification results. Zhichen and Jordan contributed the time on generating the dataset as well as data augmentation for improving the classification results.

11. Acknowledgements

Thank you to Peter Gerstoft and his teaching staff for advice and mentorship on this project. An additional thanks to Peter for suggesting the topic of plankton classification

as an area of research.

References

[1] D. Ciresan, U. Meier, and J. Schmidhuber, Multi-column deep neural networks for image classification, 2012 IEEE Conference on Computer Vision and Pattern Recognition, 2012.

[2] National Data Science Bowl, kaggle, [Online]. Available: https://www.kaggle.com/c/datasciencebowl/data. [Accessed: 13-Jun-2019].

[3] Google Cloud Platform. [Online]. [Accessed: 17-Mar-2019].

[4] K. He, X. Zhang, S. Ren, and J. Sun, Deep Residual Learning for Image Recognition, 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In NIPS, 2012.

[6] G. Huang, Z. Liu, L. V. D. Maaten, and K. Q. Weinberger, Densely Connected Convolutional Networks, 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[7] PyTorch, ResNet-34, Kaggle, 13-Dec-2017. [Online]. Available: https://www.kaggle.com/pytorch/ResNet34. [Accessed: 17-Mar-2019].

[8] Densely Connected Convolutional Networks - arXiv. [Online]. Available: https://arxiv.org/pdf/1608.06993.pdf. [Accessed: 17-Mar-2019].

[9] dalgu90, dalgu90/ResNet-18-tensorflow, GitHub, 07-May-2018. [Online]. Available: https://github.com/dalgu90/ResNet-18-tensorflow. [Accessed: 17-Mar-2019].

[10] Dieleman, Sander. Classifying Plankton with Deep Neural Networks. Sander Dieleman, 17 Mar. 2015, https://benanne.github.io/2015/03/17/plankton.html.

[11] Hu, Q, and C Davis. Accurate Automatic Quantification of Taxa-Specific Plankton Abundance Using Dual Classification with Correction. Marine Ecology Progress Series, vol. 306, 2006, pp. 5161., doi:10.3354/meps306051.

[12] Al-Barazanchi, Hussein A, et al. Plankton Image Classification Using Convolutional Neural Networks., California State University, Fullerton, 2015,