Predicting Plankton Classification with Convolutional Neural Networks

Alexa Giftopoulos | Iliana Maifeld-Carucci | Bryan Egan

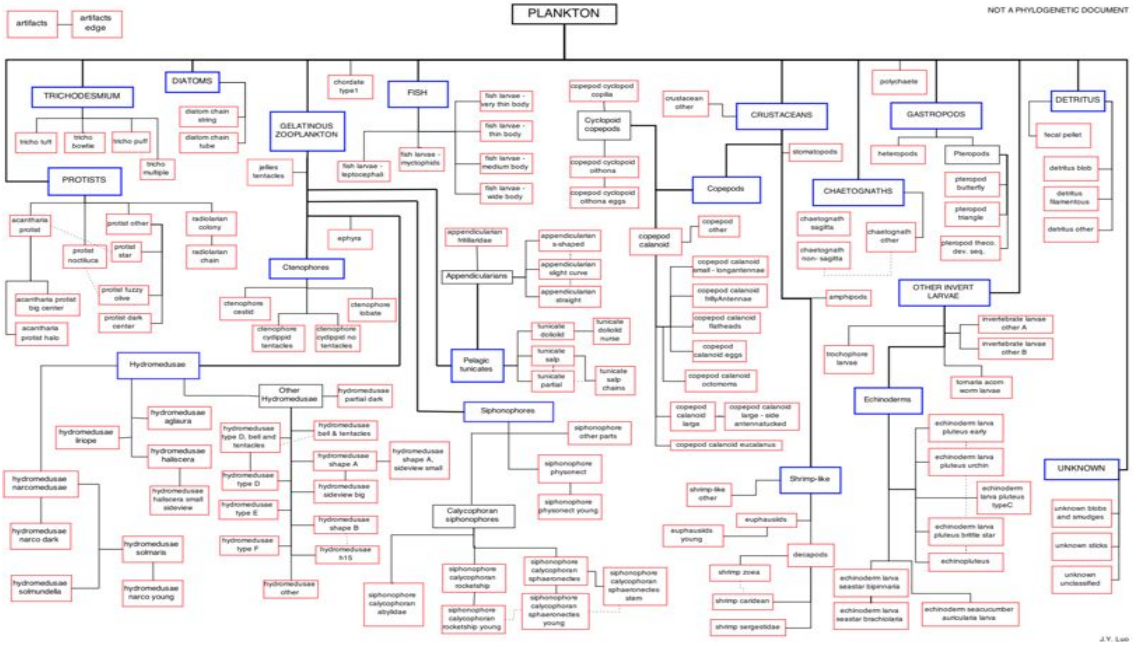
DATS 6203

Introduction

- Plankton are small microscopic organisms that float or drift in the ocean.
  - Critical food source for larger organisms.
- Because of their place on the food chain, a drop in population could impact the ecosystem.
- It is crucial to measure and monitor the plankton populations moving forward.
- There are a lot of difficulties and pitfalls.
  - Efficiently classifying plankton is time consuming and inefficient.
  - Traditional methods cannot scale to the granularity or scope necessary for large-scale studies.
- Progress is being made.
  - New approach uses an underwater camera system to capture microscopic, high-resolution images.
  - Manual analysis is infeasible.
  - Automation could make it more efficient and accurate.

The Dataset

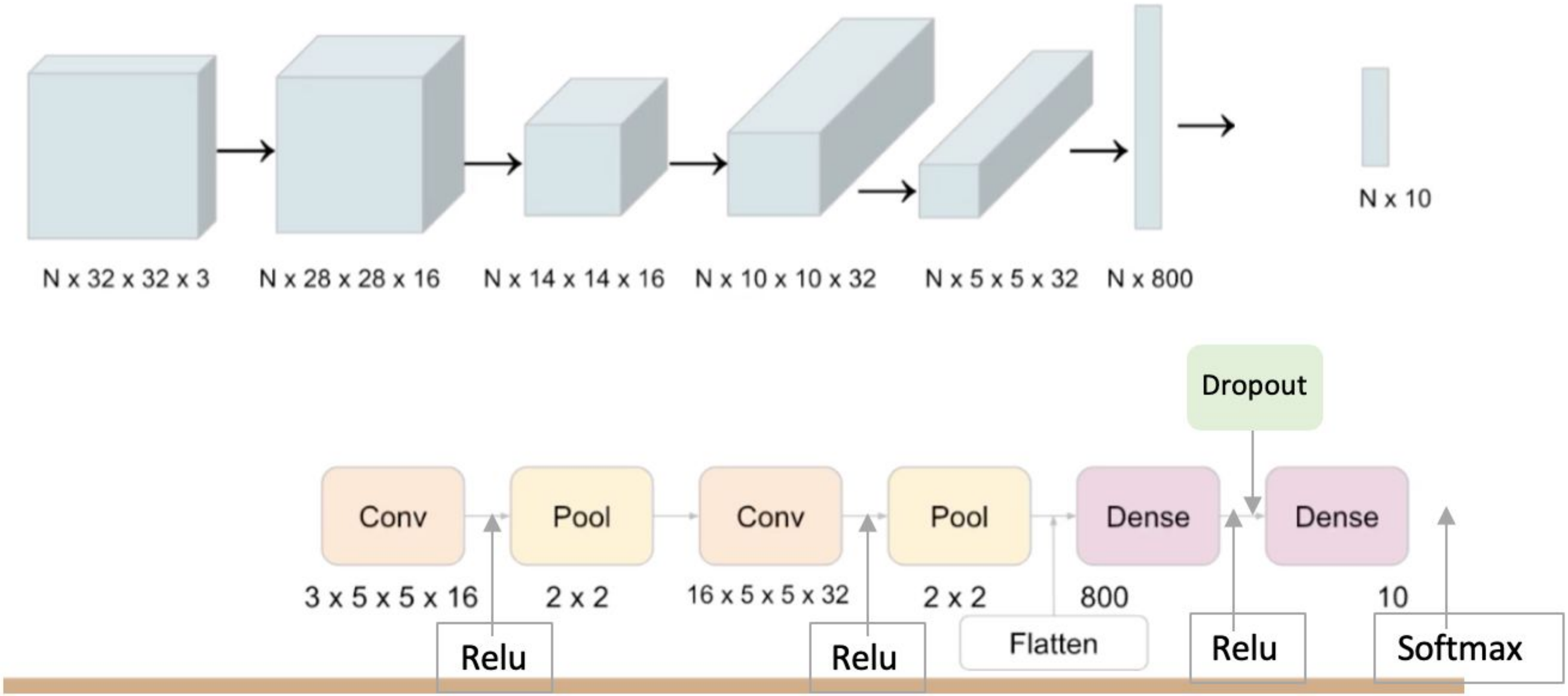
- Dataset obtained from Kaggle
- Data provided by Oregon State University’s Hatfield Marine Science Center
- Contains about 30,000 images.
- A hierarchy of classes with 11 overarching species (dark blue)
    - 7 subspecies classes (light blue)
    - Other subclasses (red)



Data Preprocessing

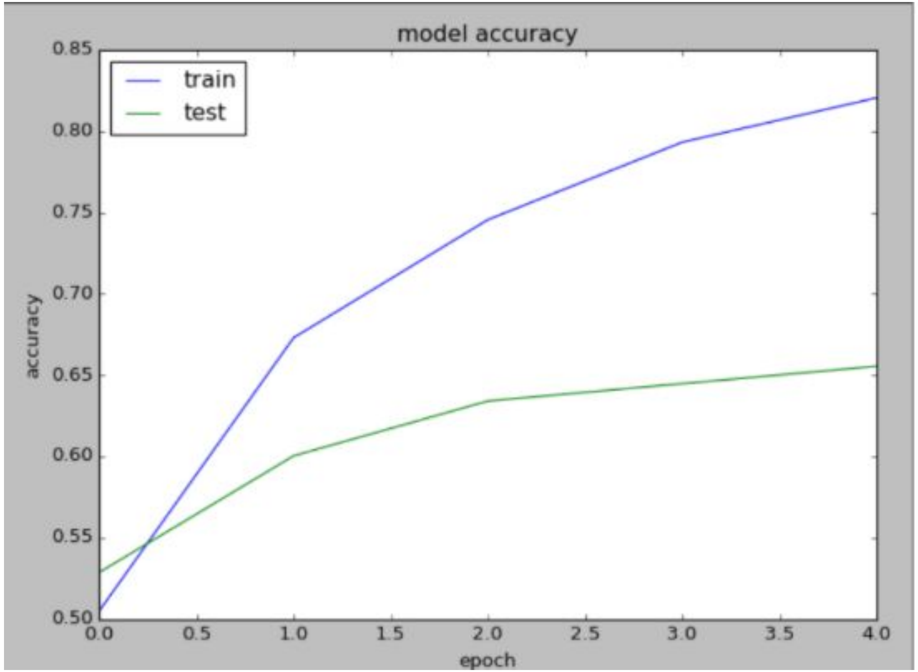
- 8 classes used:
  - Removed three classes that did not provide sufficient number of images to train and test on.
- 70/20/10 - Train/Validation/Test
- Resized and rescaled images.
  - Resized to 64x64 for consistency.
  - Rescaled color range of pixels to be normally distributed on [0,1] interval (1/255)
  - 3 color channels (RGB)
- Feature wise mean normalization
- Standardization
- ZCA Whitening

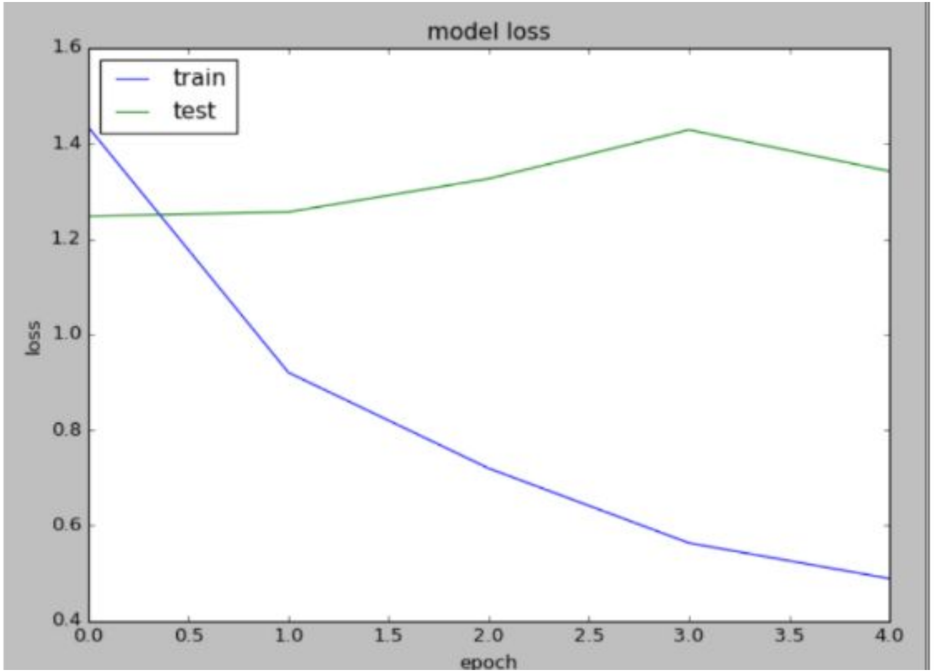
Base Model



https://www.udemy.com/deep-learning-convolutional-neural-networks-theano-tensorflow/learn/v4/overview

Base Model





Train Accuracy: 83%
Test Accuracy: 65%

Train Loss: 0.5
Test Loss: 1.3

Parameter Tuning

Layers

Layers	Accuracy
2	65%
3	62%
4	61%

Dropout Rate

Rate	Accuracy
10%	65%
20%	65%
30%	64%
40%	64%

Activation Functions

Type	Accuracy
Softmax	68%
Sigmoid	68%
Relu	63%

Resizing Images

Size	Accuracy
32 x 32	62%
64 x 64	70%
96 x 96	67%

Kernel Size

Size	Accuracy
3 x 3	62%
5 x 5	65%
8 x 8	61%

Optimizers

Type	Accuracy
SGD	65%
Adam	65%
Adagrad	64%
RMSProp	64%

Loss Functions

Type	Accuracy
MSE	75%
Poisson	73%
Categorical Crossentropy	75%

Experimental Parameters

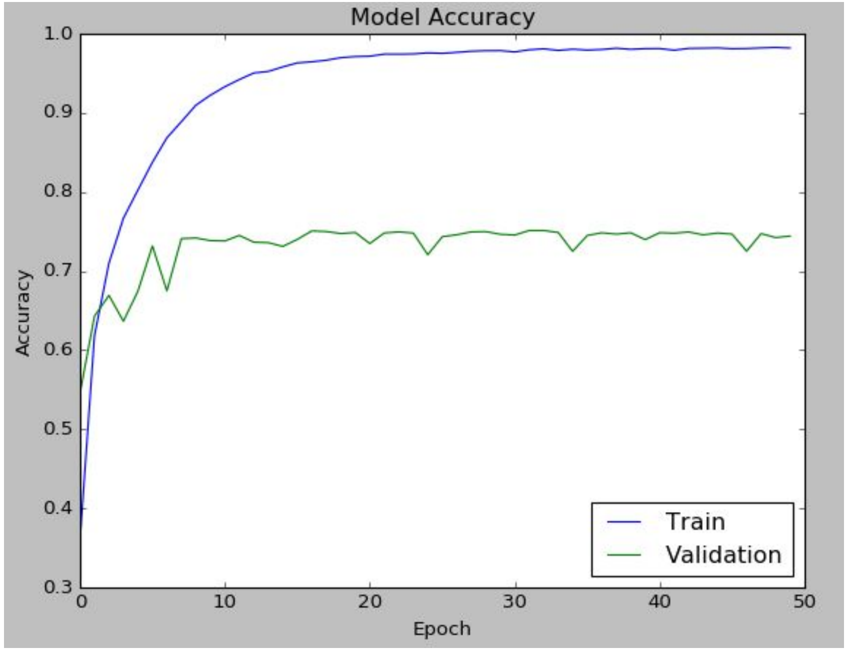
Mini-Batch

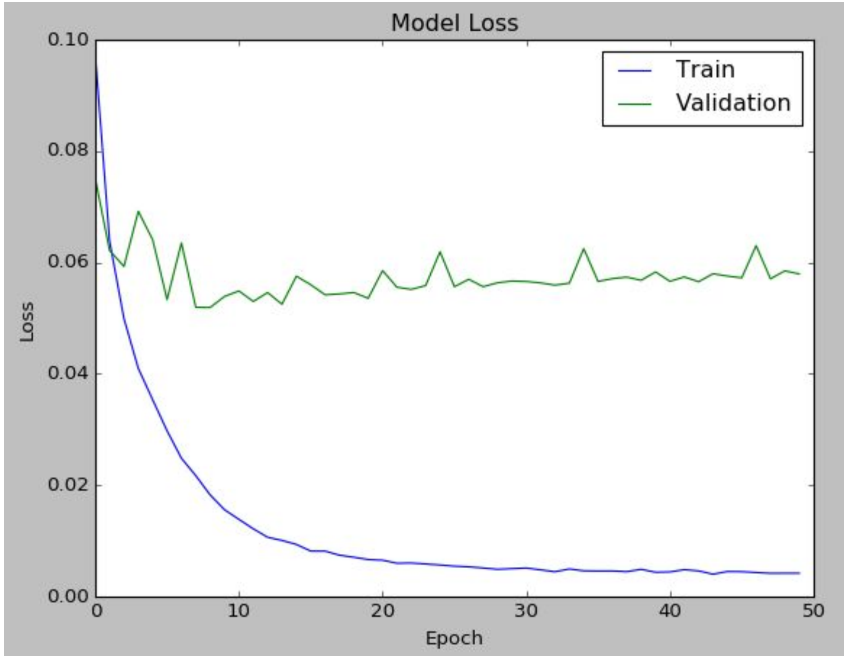
Size	Accuracy
20	72%
60	73%
100	74%
150	70%
200	73%
500	68%

Epochs

Size	Accuracy
5	70%
20	74%
50	75%
100	74%
500	73%

Final Model Results





Train Accuracy:98%
Test Accuracy: 74%

Train Loss: 0.0042
Test Loss: 0.0579

Conclusion

The final model is a much better model than the original.

But problems still persist.

The discrepancy in accuracy from the training to the test set means there is a high level of overfitting.

The loss function remaining static is also a cause for concern.