Fall 2023

Poriferal Vision: Using MobileNet to Classify Sponge Spicules through Transfer Learning

Brian Tran

Poriferal Vision: Using MobileNet to Classify Sponge Spicules through Transfer Learning

Presented to

The Department of Computer Science

San Jose State University

In Partial Fulfillment

Of the Requirements for the Class

Fall 2023: CS 298

By

Brian Tran

October 2023

Advisor: Philip Heller

Committee Member: Saptarshi Sengupta

Committee Member: Amanda Kahn (Moss Landing Marine Laboratories)

Abstract

Global warming is an ongoing issue where the Earth is rapidly warming up. It negatively affects the growth of coral through ocean warming and ocean acidification. Many coral communities, home to a large variety of marine life, are expected to be severely impacted by these effects. Past evidence suggests that sponges will take over as the primary reef builders since many species of sponges have skeletons made of silica or glass which is not affected by ocean acidification. More research is needed to determine which kinds of sponge will most likely be able to thrive in today's climate.

This can be done by sampling the seabed for the target era and identifying the spicules that are present in the sample. However, classifying the spicules by hand accurately and within a reasonable amount of time is not tractable with large amounts of spicules. Transfer learning with a pre-existing convolutional neural network (CNN) can be utilized to train a model with a small spicule dataset to classify spicules.

In this project, I use transfer learning with MobileNet, a pre-existing CNN, to classify seven categories of spicules. I then use image transformations, zero padding, and background padding on the data before training the model to try to improve its performance on the data. Background padding had the best performance although none of the different iterations of the model could classify all categories well at once.

Index terms—**Convolutional Neural Networks (CNN), Transfer Learning, Transformations, Zero Padding, Background Padding.**

Acknowledgements

I would like to thank my project Advisor, Dr. Philip Heller for introducing Poriferal Vision to me and allowing me to be a part of it during my time as a graduate student.

I would like to thank my committee member, Dr. Amanda Kahn from Moss Landing Marine Labs for providing the data used to train the model. This project could never have taken off without her contributions.

I would like to thank my second committee member, Dr. Saptarshi Sengupta for his suggestions on how to improve the performance of the model. I would not have been able to make as much progress in this project without his expertise in Machine Learning.

Table of Contents

List of Figures

List of Tables

1. INTRODUCTION

Corals are structures created by multiple organisms called polyps. There are at least 835 known species of corals that serve as the foundation for coral reefs are the homes for millions of marine species [1]. The coral reefs however, are at risk of disappearing due to global warming.

Global warming is an ongoing process in which human activity is accelerating the rate at which the surface of the Earth is heating up which affects the environment in many ways. It is the result of the greenhouse effect where a buildup of gases in the atmosphere such as carbon dioxide absorb solar energy reflected by the Earth's surface which traps heat. The oceans absorb some of this heat which raises their temperatures. Warmer oceans cause corals to expel the algae that give them their color and provide food for them, a phenomenon known as coral bleaching[2]. This results in the corals turning white and becoming vulnerable to death by starvation. Ocean acidification affects corals through a chemical process where higher levels of atmospheric carbon dioxide causes the ocean to absorb more carbon dioxide which increases the amount carbonic acid in the ocean resulting in the removal of dissolved carbonate that corals need to create calcium carbonate to maintain and grow their skeletons[3]. Both effects have dire consequences for coral reefs and the marine life that lives in them. According to Bell et al. [4], most coral species will not survive until the end of the century. However, there are studies that indicate that sponges could possibly survive and flourish in oceans affected by ocean acidification.

Sponges are one of many benthic organisms or organisms that generally live at the bottom of a body of water. Sponges can be classified into four classes of which three of them (Demospongiae, Hexactinellida, Homoscleromorpha) are made of silica or glass while the last one (Calcarea) is made of carbonate [5]. They are filter feeders that are capable of processing 10 m^3 of water per osculum or an opening in the sponge that water that enters a sponge is expelled [6]. Sponges are a part of a process known as the sponge loop where they convert dissolved organic matter (DOM) such as dissolved carbon and nitrogen into particulate organic matter (POM) in the form of detritus caused by the rapid production and shedding of sponge cells [7]. The detritus can be utilized by organisms at a higher trophic level[8]. This is important for communities located in oligotrophic seas, or seas with a low level of nutrients, where there are no other organisms that can provide this service[9]. They also accumulate pollutants from the water they filter which is useful for monitoring the state of the environment they live in[10].

In the past there have been several instances where sponges were the dominant reef builders. Some of those instances were caused by events such as the mass extinction that took place between the Triassic and Jurassic time periods which devastated organisms that built their shells and exoskeletons out of calcium carbonate such as corals due to ocean acidification[4]. A study of the fossils from that period shows that glass sponges thrived after the loss of those organisms. Bell, et al. [4] also mention that there are some reefs in the Caribbean, Atlantic, Pacific, and Indo-Pacific today that are now dominated by sponges after the coral populations declined. This indicates that sponges could potentially become the future for reefs. However, more research is needed to determine how well sponges can cope with global warming in the long run since there are many different factors that are affecting the environment today compared to the past such as human activity.

One way to further research how well sponges can thrive is by analyzing their spicules to see which types of sponge were able to thrive in the past. Spicules are the building blocks of a sponge’s skeleton. Sponges are generally made up of multiple spicules with each species having its own unique distribution of spicules which can be used to differentiate them. Most species of sponges have spicules that are made of silica while the remaining species have spicules that are made of calcium carbonate which means that they are susceptible to the effects of ocean acidification. Silica can resist breaking down for millions of years since its composition is similar to glass, so silica spicules are preserved in the geological record. Therefore, samples of silica spicules can be extracted from sediment in the seabed pertaining to a specific time period and identified to help determine which species of silica sponges are more likely to ignore the effects of ocean acidification and thrive. However, there are some challenges that come with classifying spicules with their respective species.

The size of sponge spicules can range from a few micrometers to a few millimeters depending on the species [11]. A sample of sediment from the seabed that belongs to a certain time period can contain thousands if not more of these spicules. There is technology to isolate and photograph individual spicules in a sample, but each image needs to be examined by an expert to classify which species a spicule belongs to. The problem with this is that a single person examining these spicules will not be able to go through all the images in a reasonable amount of time and multiple samples must be taken from an area to acquire enough data to make a conclusion about what species were present and thriving. Using a team of people to classify could reduce the amount of time it takes to classify these images, but each person in the team could have slightly different perceptions that could lead to conflicting conclusions about the species a spicule may belong to. A solution to avoid these issues is to use deep learning to help classify the spicules.

Deep learning is a way to simulate the human brain by trying to find patterns in data. This is done through algorithms known as Artificial Neural Networks (ANN) [12]. ANNs can be used for a variety of purposes, but for the purpose of image recognition, a specific kind of ANN known as Convolutional Neural Networks (CNN) need to be used. A CNN like an ANN has three components to it, which are an input layer, an intermediary set of hidden layers, and an output layer. The input layer takes in an input image as a set of pixel values. These values are passed to the hidden layers which are made up of a combination of convolutional, pooling, and fully-connected layers (Figure 1)[12], [13]. Convolutional layers use a kernel or two dimensional matrix filter to both provide smaller portions of an image to the neurons in a subsequent layer and recognize patterns for feature detection[13]. This allows the model to work more efficiently since every neuron in the model does not need to examine all the pixels from the original input image. The pooling layer helps to reduce the complexity of the model by applying a kernel to different parts of a set of values and returning a single value from each application of the kernel to produce a smaller area for subsequent layers to examine(Figure 2).Unlike a convolutional layer, a pooling layer does not directly contribute to the learning process. It decreases the chances of the model overfitting on the training data by reducing the number of parameters the model needs to learn [14]. The fully-connected layer is a set of multiple layers of neurons where every node in a layer is connected to all of those in the layers before and after it. This layer is at the end of the model and connects directly to the output layer which generates a value that represents the predicted label. A CNN needs to be trained with as much training data as possible to ensure the best accuracy when it classifies new data. This is done mainly through supervised learning where the training data is labeled which allows the CNN to adjust itself while training so that it can make better predictions on unseen data. Increasing the number of layers in the hidden layer allows a model to learn more complex features since more layers allows the model to examine the images in finer detail.

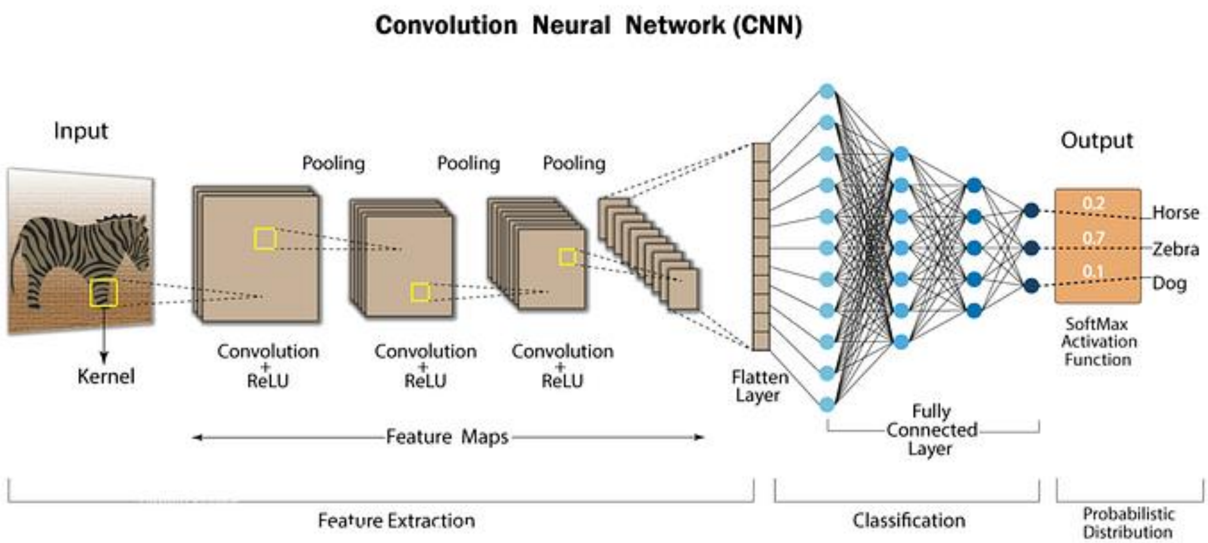

Figure 1: CNN Architecture

[https://nafizshahriar.medium.com/what-is-convolutional-neural-network-cnn-deep-learning-b3921bdd82d5]

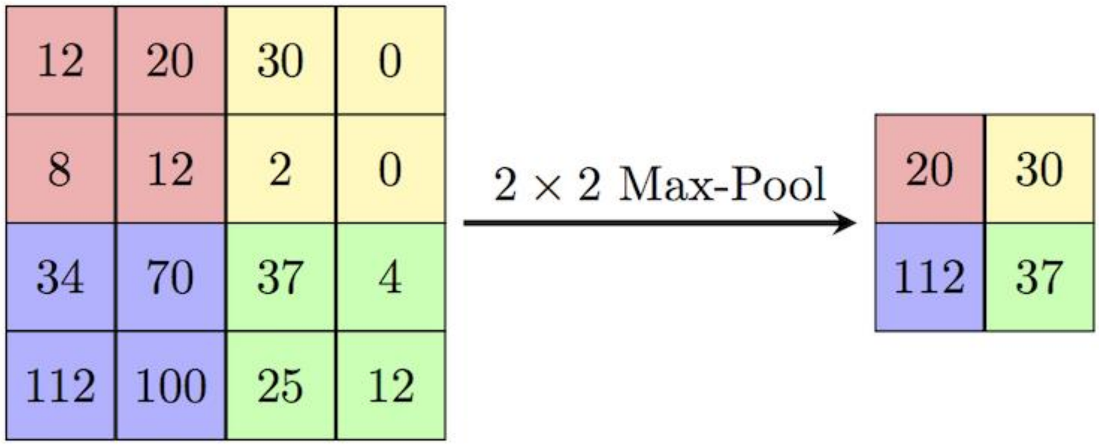

Figure 2: Pooling Example

[https://paperswithcode.com/method/max-pooling]

There have been many applications where CNNs can be used for image recognition. For example, Arora and Sharma [15] used CNNs to determine if there are early signs of brain cancer in MRI scans. The result of their study found that CNNs outperformed other machine learning algorithms by a significant amount. Similarly, Treebupachatsakul and Poomrittigul [16] use a CNN called LeNet to classify between two groups of bacteria which yielded good results. Shaily and Kala trained a CNN called ResNet on a bacteria dataset with 20 categories which achieved better results than other pre-existing classifiers [17]. [18] compared the performances of a CNN with a Support Vector Machine (SVM) when classifying between spicules and non-spicules with the CNN performing better than the SVM across all evaluation metrics. Based on the success of current applications of CNNs, it should be possible to use them to classify sponge spicules.

The purpose of this study is to use transfer learning to implement a program that will train an existing CNN called MobileNet to classify seven kinds of spicules. The performance of the model on unseen data during its training process is assessed using confusion matrices. Based off the performance of the initial model, I used transformations, zero padding, and background padding to preprocess the data before retraining the model to see if the model's performance improves and if so, which of the methods did the best.

2. METHODS

2.1 Data

The data analyzed in the current work is the basis of a large NSF grant proposal submitted in 2023 by Heller, Kahn, and Aiello of San José State University and Moss Landing Marine Laboratories. The particles of the samples are passed through a FlowCam which isolates each particle and photographs it. The FlowCam runs were paid for by a Level-Up grant from San José State University. The data contains 26 kinds of spicules, however I only used 7 of them because those categories each had at least 100 images which I decided should be the minimum number of images for the model to learn a category well. Figure 3 shows an example of each of the seven categories of spicules used to train the model.

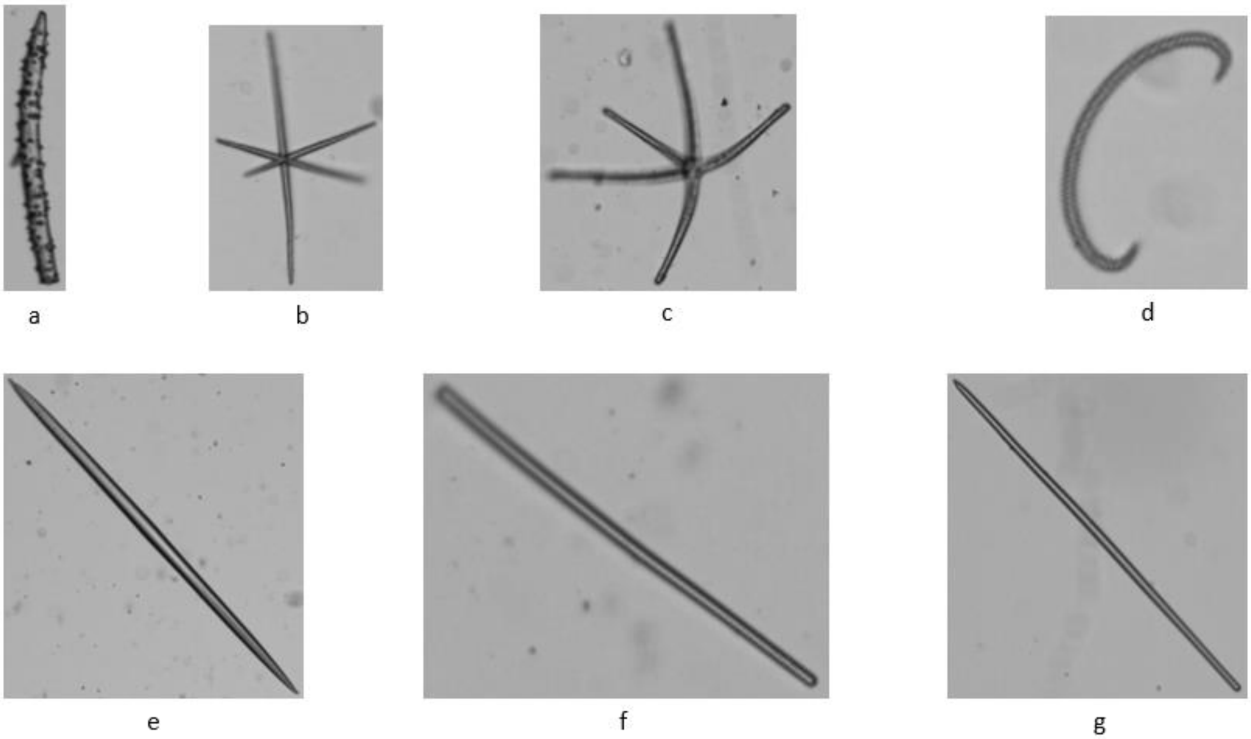

Figure 3: Examples of each Class of Spicule - a. Acanthosytle, b. Hexactin, c. Pentactin, d. Sigma, e. Oxea, f. Strongyle, g. Style

The data was divided into a training set, a validation set, and a test set using a 70-15-15 split. The training set is used by the model to learn from the data so that it can adjust its parameters accordingly. The validation set is used by the model to gauge its performance after every epoch or when the model goes through the entire training set once. The model will adjust its learning process when going through the training data again based on its performance on the validation set. The test set is used as a final evaluation of the model on unseen data. Table 1 shows how each spicule category is divided into the training, validation, and test sets.

Table 1. Spicule Dataset Distribution

Spicule	Training	Validation	Test	Total
Acanthostyle	279	61	61	401
Hexactin	90	20	20	130

Oxea	315	68	68	451
Pentactin	113	25	25	163
Sigma	154	34	34	222
Strongyle	275	60	60	395
Style	942	203	203	1348

2.2 Preprocessing Data

2.2.1 Transformations

The data is imbalanced across the different spicule categories. Style contains far more data than the other categories (Table 1). This puts the model at a possible risk of overfitting since it could learn Style better than the other categories due to the amount of training data from Style it will be exposed to compared to the other categories. One way to prevent overfitting from happening is to augment the data for underrepresented classes using transformations. Transformations in machine learning are a series of operations that can be applied to an image to create variations of that image which serves as both a way to generate more data samples from a limited set and introduce more variation into the data. In this project, six transformations from the Albumentations library were applied to every image in the training sets of two of the spicule categories. The two categories that the transformations were applied to are oxea and strongyle because they are very similar in appearance to style (Figure 3e, 3f, 3g). The six transformations that I applied to the data are contrast, gaussian noise, horizontal flip, vertical flip, transpose, and a combination of horizontal and vertical flipping. Figure 4 shows examples of each of the transformations applied to the data.

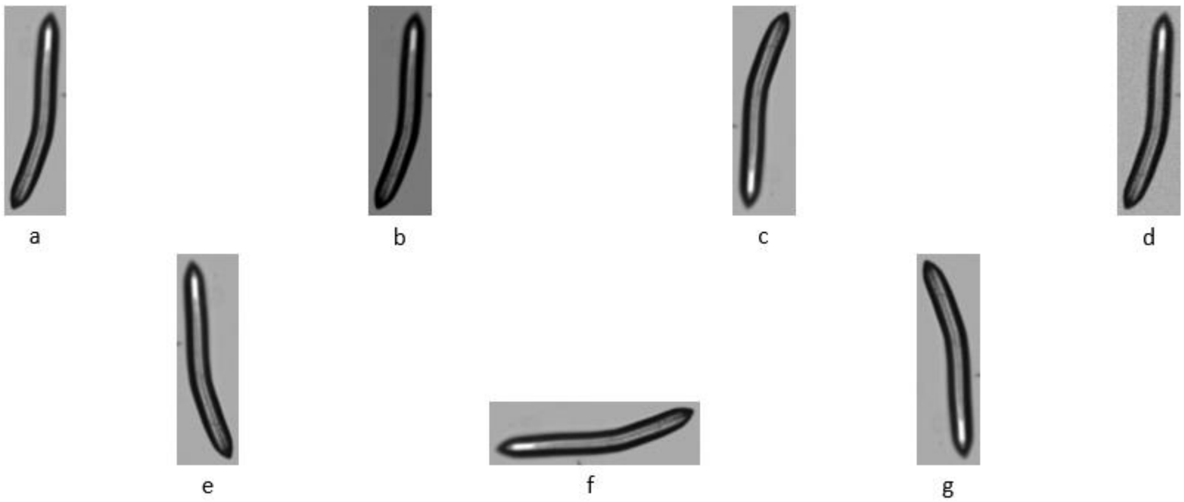

Figure 4: Examples of Each of the Transformations Used: a. Original Image, b. Contrast, c. Double Flip, d. Gaussian, e. Horizontal Flip, f. Transpose, g. Vertical Flip

2.2.2 Padding

The spicule images come in a variety of sizes and their dimensions can be very disproportionate depending on the orientation of the spicule at the time its picture was taken by FlowCam. Figure 5 shows the dimensions of two images in the spicule dataset.

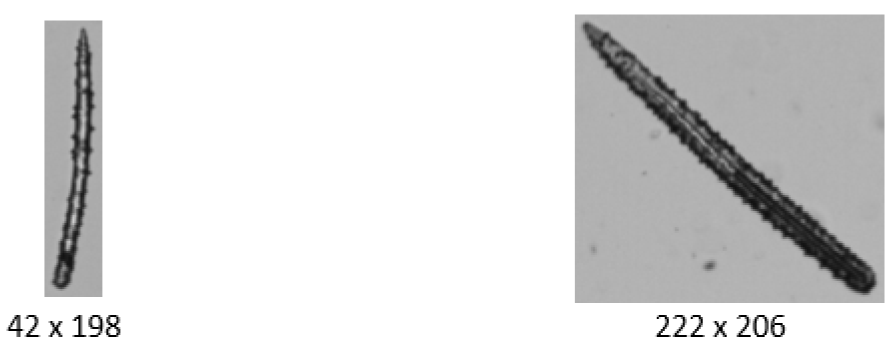

Figure 5: Dimensions of Some of the Spicule Images.

MobileNet by default expects an image with dimensions 224 x 224, so the spicule images will be forced to conform to that size. This results in the images becoming distorted since they are being stretched or compressed to fit into the 224 x 224 size for MobileNet. The amount of distortion will vary depending on how close the dimensions are to each other. Figure 6 shows how the spicules from Figure 5 would look to the model they are forced to become 224 x 224.

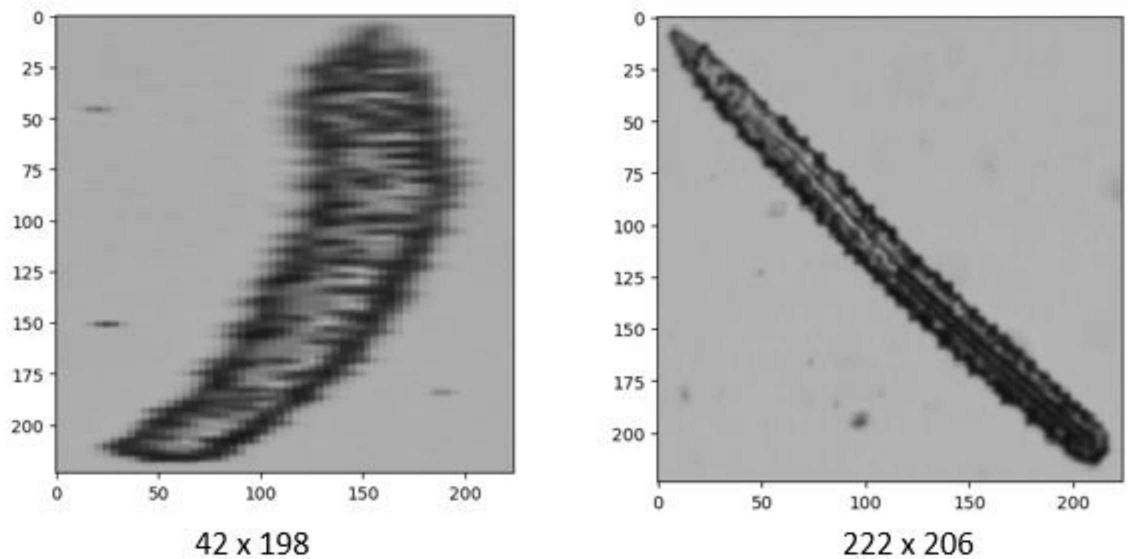

Figure 6: How the Images From Figure 5 Look to the Model

The 42 x 198 image is unrecognizable compared to the original version because the image was stretched far more in one dimension than the other. On the other hand, the 222 x 206 image looks very close to the original because its dimensions are very close to each other, so they are stretched by a very similar amount. This variation in the possible appearances of spicules to the model will affect how well it can learn the features of each kind of spicule. One way to help the model see the images similarly to their original appearance is to adjust the dimensions of the images by padding them with extra pixels so that their dimensions are equal. In this experiment, zero padding and background padding were used to adjust the dimensions of the spicules.

Zero padding is a method of adjusting the size of an image by adding black pixels which have a value of zero around an image. When a layer of a CNN encounters zero padding while training, it will not learn any significant features from the zero padded border since it does not contribute any meaningful information to the learning process [19]. Zero padding was applied two ways depending on the dimensions of the image. If the larger dimension of the image is less than 224, then the image is padded on all sides until it becomes 224 x 224. Otherwise, if the image's larger dimension is greater than 224, then the smaller of the two dimensions is background padded until the dimensions are equal. Figure 7 shows an example of what a zero padded image looks like to the model.

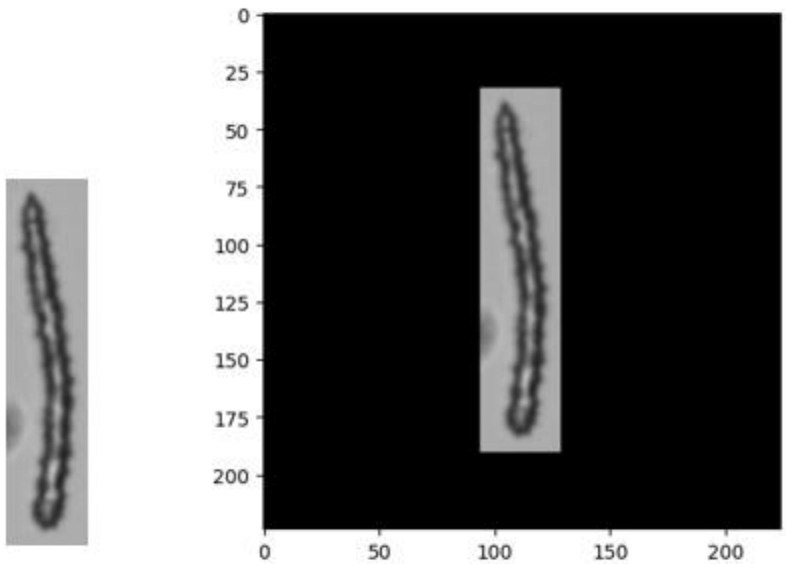

Figure 7: Comparison of a Spicule vs. the Zero Padded Version the Model Sees

Background padding is very similar to zero padding except that instead of padding with black pixels, it pads using the color of the background of the image. This is a method that is applicable to images that have relatively uniform backgrounds since adding more background should ideally not introduce significant features for the model to learn. The amount of padding added to the spicule images is the same as zero padding. The color for the background padding was picked using the color of the top rightmost pixel of the image. There is a possibility that the top rightmost pixel is a different color from the rest of the background, but it is not expected to be the case for most of the spicule images. Figure 8 shows an example of what a background padded image looks like to the model.

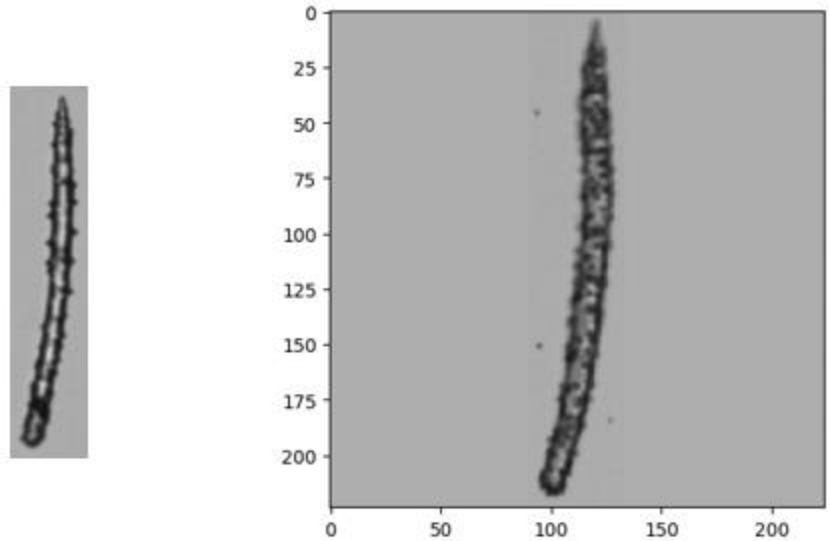

Figure 8: Comparison of a Spicule vs. the Background Padded Version the Model Sees

2.3 Transfer Learning

An extensive amount of training data and time is required to train a model from scratch, which would be impossible to do with limited sized data sets. Transfer learning is a solution to this problem by taking an existing pre-trained model and fine-tuning it so that it can reliably extract and learn features from a small dataset[20].

This project is a continuation of another project to determine a suitable model for classifying sponge spicules using transfer learning. In that project, Domala [21] did a comparison of eight different models (VGG16, VGG19, ResNet50, ResNet152V2, Inception-v3, InceptionResNetV2, MobileNet, MobileNetV2) to compare their performances on two datasets which are spicules vs. non-spicules and two kinds of spicules. The model that performed the best overall was MobileNet.

MobileNet is a model that is trained on the ImageNet Dataset which consists of 1,000 classes which altogether contains 1,281,167 training images, 50,000 validation images, and 100,000 test images. MobileNet is less complex than other models which allows it to take up less space, but it sacrifices some accuracy in return[22]. To train MobileNet with the sponge spicule data, I removed the last four layers of the original MobileNet model to remove some of the layers that learned some of the more complex features of the dataset. An output layer with the same number of outputs as categories in the spicule data set was added to the end of the model. The output layer has a softmax function that will change the output of the layer so that it produces a probability distribution of how confident the model is that an image belongs to each of the categories it. The last four layers of the new model were set to be trainable, so that they could learn the more complex features of the spicule dataset.

3. Results

3.1 Initial Model

Figure 9 and Table 2 show the performance of the model on the spicule data without any modifications. The model had an 84.9% accuracy overall. When considering accuracy per spicule category, acanthostyle, hexactine, pentactin, sigma, and style were able to achieve accuracies close to and above 90%. On the other hand, oxea and strongyle did not perform as well with accuracies of 55% and 80.9%. Most of the incorrectly classified images for each of these categories were predicted to be style. It is worth noting that for style, all the incorrect classifications were a combination of both oxea and strongyle.

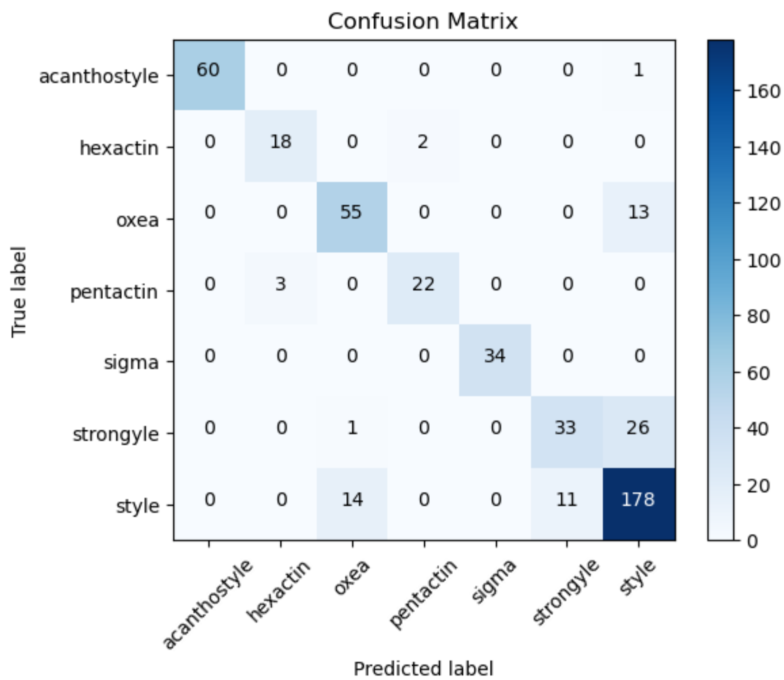

Figure 9: Result of Initial MobileNet Model

Table 2: Initial Model Accuracy Per Spicule

Spicule	**Correctly Classified**	**Incorrectly Classified**	**Accuracy**
Acanthostyle	60	1	0.984
Hexactin	18	2	0.900
Oxea	55	13	0.809
Pentactin	22	3	0.880
Sigma	34	0	1.000
Strongyle	33	27	0.550
Style	178	25	0.877
Total	400	71	0.849

Figure 10 and Table 3 show the performance of the model when it was trained on data that had transformations applied to it. The transformations were applied only to the training data for Oxea and Strongyle to try to improve their performances compared to the initial model. The accuracy for oxea and strongyle both decreased to 75% and 51.7% while the accuracy for style increased to 91.6%. The remaining four categories remained unchanged.

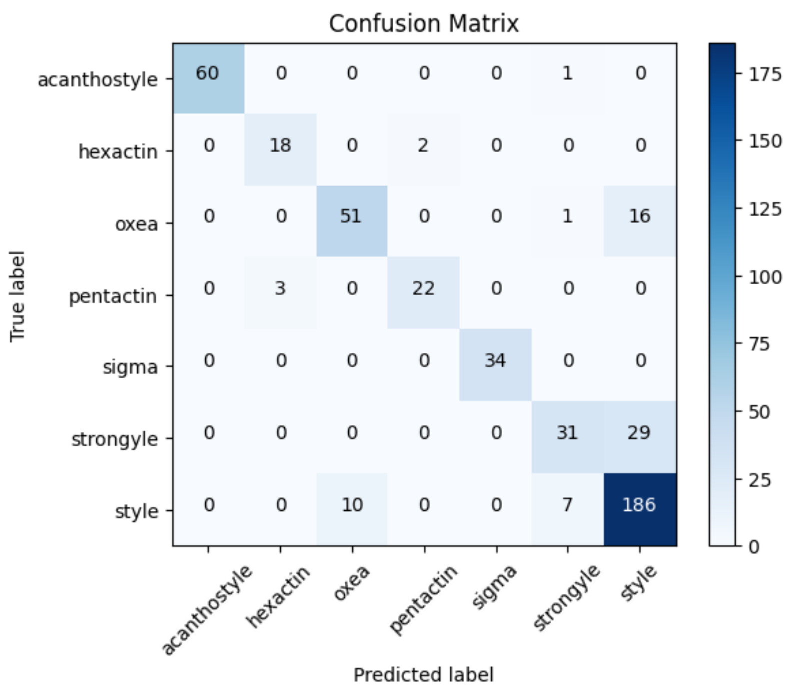

Figure 10: Result of using Transformations

Table 3: Accuracy Per Spicule for Model using Transformations on Data

Spicule	Correctly Classified	Incorrectly Classified	Accuracy
Acanthostyle	60	1	0.984
Hexactin	18	2	0.900
Oxea	51	17	0.750
Pentactin	22	3	0.880

Sigma	34	0	1.000
Strongyle	31	29	0.517
Style	186	17	0.916
Total	402	69	0.854

Figure 11 and Table 4 show the performance of the model when it was trained on zero padded spicule images. Compared to the initial model, Strongyle had a large decrease in accuracy to 35% while the accuracy for style increased to 95.1%.

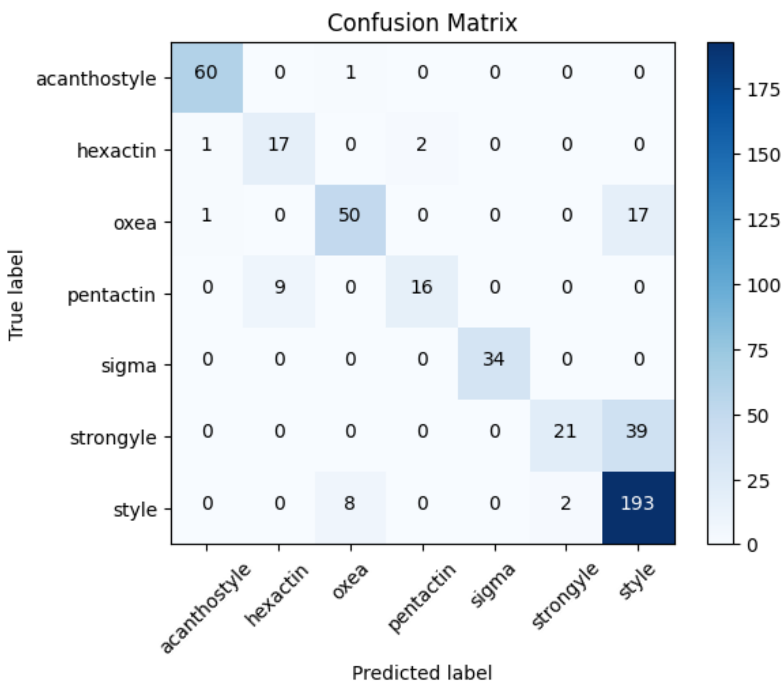

Figure 11: Result of Using Zero Padding

Table 4: Accuracy Per Spicule for Zero Padding

Spicule	**Correctly Classified**	**Incorrectly Classified**	**Accuracy**
Acanthostyle	60	1	0.984
Hexactin	17	3	0.850
Oxea	50	18	0.735
Pentactin	16	9	0.640
Sigma	34	0	1.000

Strongyle	21	39	0.350
Style	193	10	0.951
Total	391	80	0.830

Figure 12 and Table 5 show the performance of the model when it was trained on data that had background padding applied to all the images in the dataset. The overall performance of the model improved to about 87.3% and the accuracy of Style increased to 95.6%. Oxea and Hexactin had a slight decrease in accuracy.

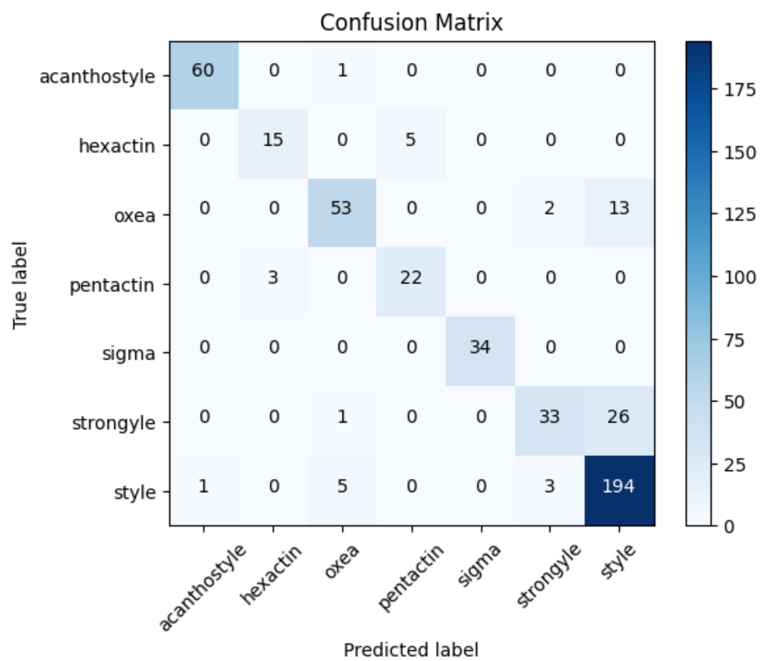

Figure 12: Result of Using Background Padding with All Data

Table 5: Accuracy per Spicule for Background Padding with All Data

Spicule	**Correctly Classified**	**Incorrectly Classified**	**Accuracy**
Acanthostyle	60	1	0.984
Hexactin	15	5	0.750
Oxea	53	15	0.779
Pentactin	22	3	0.880
Sigma	34	0	1.000

Strongyle	33	27	0.550
Style	194	9	0.956
Total	411	60	0.873

Figure 13 and Table 6 show the performance of the model when the amount of training data for style is limited to 300 images to give the model an equal exposure to Oxea, Strongyle, and Style. The model had a significant drop in overall accuracy to about 79% which was caused by the accuracy of Style dropping to about 59%. However, the accuracies of Oxea and Style had significant improvements to their accuracies to 94.1% and 95%.

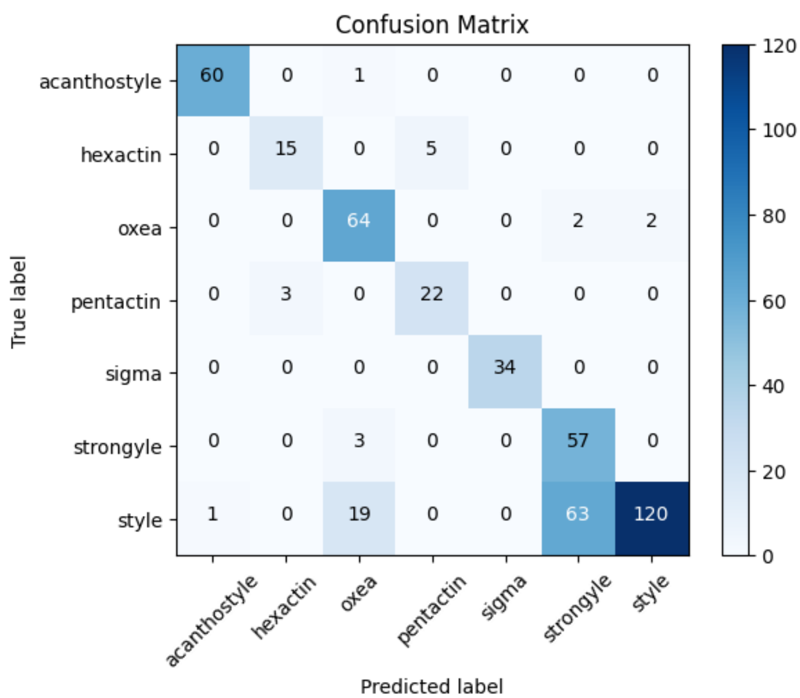

Figure 13: Result of Using Background Padding with Even Training Sets

Table 6: Accuracy Per Spicule for Background Padding with Even Data

Spicule	**Correctly Classified**	**Incorrectly Classified**	**Accuracy**
Acanthostyle	60	1	0.984
Hexactin	15	5	0.750
Oxea	64	4	0.941

Pentactin	22	3	0.880
Sigma	34	0	1.000
Strongyle	57	3	0.950
Style	120	83	0.591
Total	372	99	0.790

4. Discussion

The initial model performed within expectations. There were some issues differentiating between Hexactin and Pentactin because they have the least amount of data available out of the seven categories of spicules used and the images were being distorted. They are similar in appearance with the main difference being that the structure of Hexactin is straight while Pentactin is curved. In the case of Oxea, Strongyle, and Style, the model predicted Style reasonably well while it had issues with predicting Oxea and Strongyle. There are multiple factors that contribute to this observation. First, the model was exposed to about three times the amount of training data for Style compared to the other two categories. This resulted in the model most likely overfitting on Style which explains the model's performance for those three categories. Secondly, the spicules are extremely similar in appearance. The difference in their appearances is that Oxea has two needle like tips at the ends, Strongyle has rounded ends, and Style has one needle like end and one rounded end. Style in a sense is like a hybrid of Oxea and Strongyle since it shares similar features with them. This combined with the distorted images from the train, valid, and test sets would explain why the incorrect predictions for Oxea and Strongyle were predicted to be Style and the incorrect predictions for Style were divided between Oxea and Strongyle.

The model's performance when trained on data where the training data for Oxea and Strongyle were augmented using transformations was worse than expected. The expectation was that the model's accuracy for all three categories would improve because there was more data available in Oxea and Strongyle for the model to learn from. The model ended up performing better on Style, but worse for Oxea and Strongyle. The cause was most likely the distortion that was applied to all the images including the new images created by transformations. The image distortion was not discovered until after I tried applying transformations to the original data. I later trained the model on padded images that were augmented by transformations, but it did not perform as well as the padded images alone, so I did not include the results of that here.

I had two approaches for applying zero padding. My initial method for doing this was to look for the longest dimension across all images and pad them so that they were all squares whose width and height are equal to the longest dimension. The model performed far worse than the initial model with drops in accuracy across all spicule categories, but Sigma and Strongyle. When these images were loaded into the train, valid, and test sets, all of the images were scaled down the same amount. However, the original images were all different sizes, so the larger images were able to retain more of their features when scaled down compared to the smaller ones that lost a far larger amount. With the second approach where the images were adaptively padded individually, the model performed far closer to the initial model, but was slightly worse. A possible explanation for why this happened is that the images had varying amounts of padding, so the model could be learning the amount of background as a factor when predicting the labels of images.

The model performed the best overall when trained with all the data with background padding. This improvement is due to the accuracy of Style having a significant increase in accuracy. Oxea and Strongyle did not improve, but instead got slightly worse. Since the issue with the image distortion was resolved I decided to revisit the possibility that the model was overfitting on Style because it was exposed to far more training samples for Style compared to Oxea and Strongyle. Instead of trying to increase the number of samples for Oxea and Strongyle, I decided to limit the amount of data that the model sees for Style to 300. When trained on this data, the model had a significant drop in accuracy for Style, but the accuracies for Oxea and Strongyle improved significantly. There appears to be a tradeoff between the accuracy of Style and the accuracies of Oxea and Strongyle. I experimented with varying the amount of Strongyle training data that model can see to determine how it affects the accuracies of all three categories. I could not find a trend that leads to reasonable accuracies across all three of these categories.

4.1 Conclusion

This paper demonstrates how to use transfer learning to train a pre-existing CNN on datasets that are too small to train a CNN from scratch. This was done with MobileNet by removing the last few layers of the model and adding a layer that would provide outputs equal to the number of categories in the target dataset. The last few layers of the model were retrained to learn the higher-level features of the data. Based on the performance of this initial mode, transformations, zero padding, and background padding were individually applied to the data before training the model to try to improve the model's performance for the categories

Oxea, Strongyle, and Style. Out of these three methods, background padding achieved the best results for improving the model's accuracy for these three categories with the caveat being that not all of them could be predicted well at the same time. Instead, it required two different methods where one would achieve the best accuracy for Style while the accuracies Strongyle and Oxea would suffer while the other would achieve the best accuracies for Strongyle and Oxea while the accuracy of Style would suffer.

4.2 Next Steps

This project is a part of a larger project to develop a three-part pipeline ("Poriferal Vision") proposed by Philip Heller and Amanda Kahn. Figure 14 shows the three parts of Poriferal Vision.

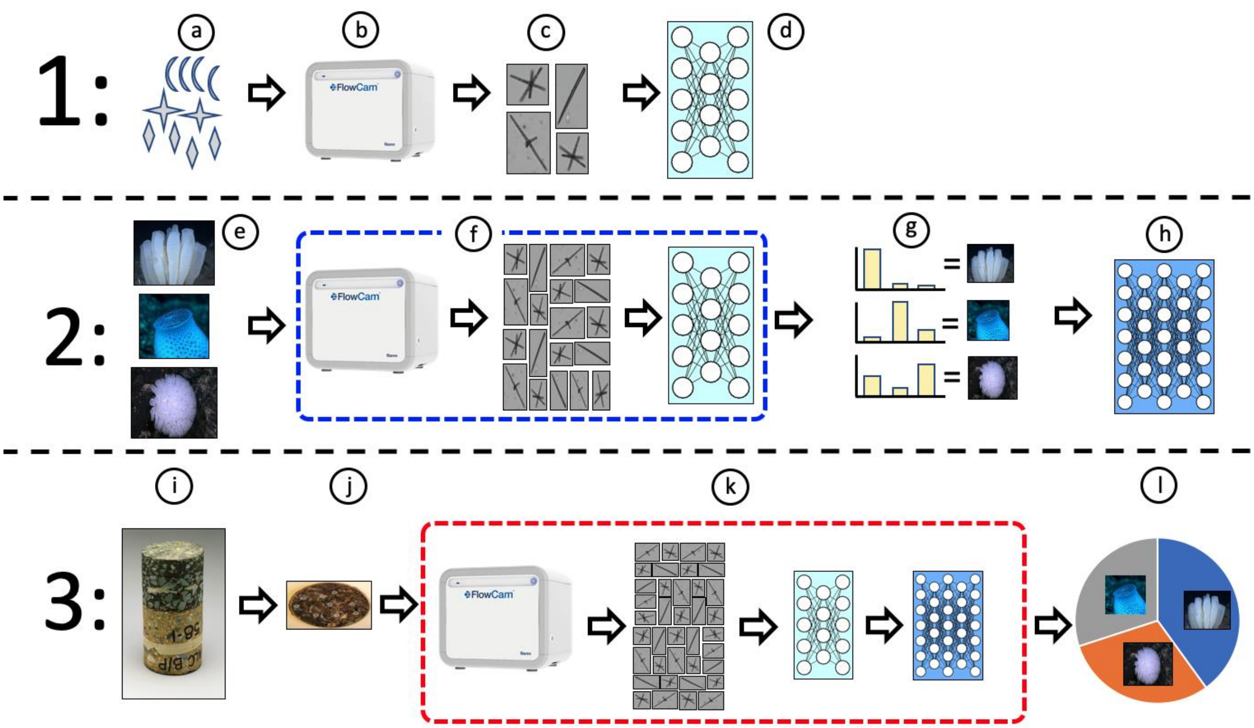

Figure 14: Poriferal Vision Pipeline

The first part of the pipeline, which this project aimed to progress, is developing a model that can classify images of spicules taken by FlowCam. The second part is using the spicules from known sponge species to teach the model to identify the species based on the spicules present in a distribution. The last part is taking the spicules from the seabed samples and using the model to identify what species are present in the sample.

More work is needed on the first part of the pipeline before moving on to the second part. The model was trained to classify 7 of the 26 available categories of spicules since the remaining 19 were deemed to not have enough data. Once more data is acquired, the remaining categories can be incorporated into the model and any adjustments to improve the performance can be made from there. If the accuracies of any single trained instance of the model is unable to classify all spicule categories with reasonable accuracies then there are a couple of possible approaches to deal with this issue.

One possible approach is to have multiple variations of the model make predictions on the unlabeled data. The predictions for each category can be based off which of the models can predict that category the best. For example, if the two variations of the model that were trained on background padded data made predictions, the spicules that are predicted as Style by the variant that was trained on all of the training data should be assumed to be Style while the spicules predicted to be Oxea or Strongyle by the variant that was trained with limited training data should be assume to be Oxea or Strongyle. If there are spicules that are predicted to be Style by the variant that predicts Style well but are predicted to be Oxea or Strongyle by the one that predicts both well, then those spicules should be examined manually. The limitation to this approach is that it should be done only if the model has issues with only a small portion of the categories that its learning since it would waste time and space if too many variations of a model need to be trained to ensure that all categories can be covered.

Another approach is to combine spicule categories based off an ontology. An ontology is a grouping of categories and the relationships that relate them to each other. Figure 15 shows a hierarchical ontology for a subset of spicule categories.

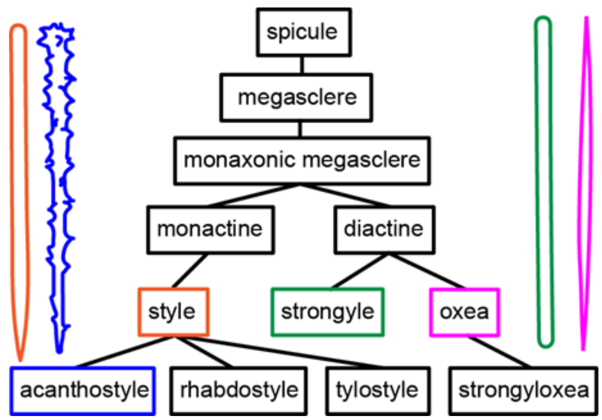

Figure 15: Spicule Ontology

In the spicule ontology, Acanthostyle is related to Style since it has the same general shape where it has one rounded end and one pointed end, but it has an extra feature of spikes which Style doesn't have. Oxea, Strongyle, and Style share the common category of Monaxonic Megasclere which would be the label of the category if they were joined into a single category. Figure 16 shows the performance of the model when it is trained on data where Oxea, Strongyle, and Style are consolidated into Monaxonic Megasclere.

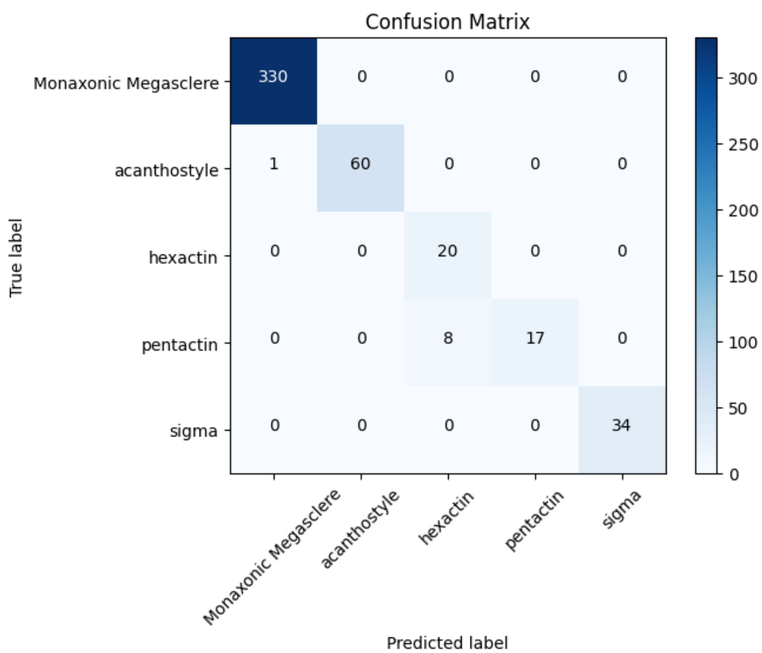

Figure 16: Result of Training Model Using Monaxonic Megasclere as a Category

The model does extremely well in this case. However, this approach requires knowledge of the spicule makeup of the different species of sponges to determine which spicule categories to consolidate. The spicules that are being combined should not be the sole differentiating factor between certain species of sponges. For example, if there are three sponge species that share the same spicule makeup except that one has Oxea, another has Strongyle, and the last has Style, then saying that one of three possible species of sponge is present does not provide as significant amount of information as narrowing it down to just one.

4.3 Expanding the Scope

This pipeline can be applied for training the model on other datasets. To do this, the necessary Python libraries need to be installed and a data folder needs to exist in the same directory as the program. The data folder should contain labeled folders of images for each category that the model needs to learn. The program will be able to divide up the data accordingly and train the model from there. If zero padding or background padding are used and the original images need to extracted from the padded images, then the image names should end with their dimensions in the format "*name_width*x*height*" where *name* is the image name, *width* and *height* are the dimensions of the original image.

References

[1] N. Knowlton, "The future of coral reefs".

[2] A. E. Douglas, "Coral bleaching––how and why?," *Mar. Pollut. Bull.*, vol. 46, no. 4, pp. 385–392, Apr. 2003, doi: 10.1016/S0025-326X(03)00037-7.

[3] J. C. Orr *et al.*, "Anthropogenic ocean acidification over the twenty-first century and its impact on calcifying organisms," *Nature*, vol. 437, no. 7059, pp. 681–686, Sep. 2005, doi: 10.1038/nature04095.

[4] J. J. Bell, S. K. Davy, T. Jones, M. W. Taylor, and N. S. Webster, "Could some coral reefs become sponge reefs as our climate changes?," *Glob. Change Biol.*, vol. 19, no. 9, pp. 2613–2624, Sep. 2013, doi: 10.1111/gcb.12212.

[5] R. W. M. Van Soest *et al.*, "Global Diversity of Sponges (Porifera)," *PLoS ONE*, vol. 7, no. 4, p. e35105, Apr. 2012, doi: 10.1371/journal.pone.0035105.

[6] A. S. Kahn, G. Yahel, J. W. F. Chu, V. Tunnicliffe, and S. P. Leys, "Benthic grazing and carbon sequestration by deep-water glass sponge reefs: Deep-water glass sponge reefs," *Limnol. Oceanogr.*, vol. 60, no. 1, pp. 78–88, Jan. 2015, doi: 10.1002/lno.10002.

[7] J. Vicente *et al.*, "Unveiling hidden sponge biodiversity within the Hawaiian reef cryptofauna," *Coral Reefs*, vol. 41, no. 3, pp. 727–742, Jun. 2022, doi: 10.1007/s00338-021-02109-7.

[8] L. Rix *et al.*, "Coral mucus fuels the sponge loop in warm- and cold-water coral reef ecosystems," *Sci. Rep.*, vol. 6, no. 1, p. 18715, May 2016, doi: 10.1038/srep18715.

[9] J. M. De Goeij *et al.*, "Surviving in a Marine Desert: The Sponge Loop Retains Resources Within Coral Reefs," *Science*, vol. 342, no. 6154, pp. 108–110, Oct. 2013, doi: 10.1126/science.1241981.

[10] J. L. Carballo and J. J. Bell, "Climate Change and Sponges: An Introduction," in *Climate Change, Ocean Acidification and Sponges*, Springer International Publishing, 2017. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-319-59008-0_1

[11] M. Łukowiak, "Utilizing sponge spicules in taxonomic, ecological and environmental reconstructions: a review," *PeerJ*, vol. 8, p. e10601, Dec. 2020, doi: 10.7717/peerj.10601.

[12] K. O'Shea and R. Nash, "An Introduction to Convolutional Neural Networks." arXiv, Dec. 02, 2015. Accessed: Dec. 10, 2022. [Online]. Available: http://arxiv.org/abs/1511.08458

[13] S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," in *2017 International Conference on Engineering and Technology (ICET)*, Antalya: IEEE, Aug. 2017, pp. 1–6. doi: 10.1109/ICEngTechnol.2017.8308186.

[14] M. Sheng, H. Zeng, J. Li, and W. Sun, "Pooling and Convolution Layer Strategy on CNN for Melanoma Detection," in *2021 3rd International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI)*, Taiyuan, China: IEEE, Dec. 2021, pp. 153–161. doi: 10.1109/MLBDBI54094.2021.00038.

[15] S. Arora and M. Sharma, "Deep Learning for Brain Tumor Classification from MRI Images," in *2021 Sixth International Conference on Image Information Processing (ICIIP)*, Shimla, India: IEEE, Nov. 2021, pp. 409–412. doi: 10.1109/ICIIP53038.2021.9702609.

[16] T. Treebupachatsakul and S. Poomrittigul, "Bacteria Classification using Image Processing and Deep learning," in *2019 34th International Technical Conference on*

Circuits/Systems, Computers and Communications (ITC-CSCC), JeJu, Korea (South): IEEE, Jun. 2019, pp. 1–3. doi: 10.1109/ITC-CSCC.2019.8793320.

[17] T. Shaily and S. Kala, “Bacterial Image Classification Using Convolutional Neural Networks,” in *2020 IEEE 17th India Council International Conference (INDICON)*, New Delhi, India: IEEE, Dec. 2020, pp. 1–6. doi: 10.1109/INDICON49873.2020.9342356.

[18] S. Saxena, P. Heller, A. S. Kahn, and I. Aiello, “Poriferal Vision: Classifying Benthic Sponge Spicules to Assess Historical Impacts of Marine Climate Change,” in *Foundations of Intelligent Systems*, vol. 12117, D. Helic, G. Leitner, M. Stettinger, A. Felfernig, and Z. W. Raś, Eds., in Lecture Notes in Computer Science, vol. 12117. , Cham: Springer International Publishing, 2020, pp. 205–213. doi: 10.1007/978-3-030-59491-6_19.

[19] M. Hashemi, “Enlarging smaller images before inputting into convolutional neural network: zero-padding vs. interpolation,” *J. Big Data*, vol. 6, no. 1, p. 98, Dec. 2019, doi: 10.1186/s40537-019-0263-7.

[20] S. R. Burri, S. Ahuja, A. Kumar, and A. Baliyan, “Exploring the Effectiveness of Optimized Convolutional Neural Network in Transfer Learning for Image Classification: A Practical Approach,” in *2023 International Conference on Advancement in Computation & Computer Technologies (InCACCT)*, Gharuan, India: IEEE, May 2023, pp. 598–602. doi: 10.1109/InCACCT57535.2023.10141701.

[21] S. Domala, “PORIFERAL VISION: Deep Transfer Learning-based Sponge Spicules Identification & Taxonomic Classification,” Master of Science, San Jose State University, San Jose, CA, USA, 2022. doi: 10.31979/etd.9psx-933e.

[22] A. G. Howard *et al.*, “MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications”.