Regrading Degrading

A Machine Learning Approach to Coral Reef Health Classification

Nate Leary, Audrey Moessing, Sam Lee, Andrew Goldston, Aidan Quigley*

April 2025

1 Introduction

Coral reefs are some of the most biologically diverse ecosystems on the planet but also some of the most threatened by increasing temperatures, ocean acidity, and other chemical or physical disruptions to their environment. To protect these rare and valuable aquatic communities, conservationists need to understand the extent of the damage that has occurred as well as identify any possible patterns between reef bleaching and any number of possible damage variables. Our project seeks to answer the following research question: *How can machine learning models be used to classify coral reef health status based on image data, and how can these models be leveraged to generate new labeled data sets for monitoring coral degradation?*

2 Background

Coral reefs are sub-aqueous calcium carbonate structures formed by marine invertebrates that play a crucial role in preserving biodiversity, preventing coastal erosion, and supporting marine economies through tourism and trade (Burke et al., 2011; Spalding et al., 2017). However, these ecosystems are under severe ecological threat due to rising sea temperatures, ocean acidification, over-exploitation, destructive fishing practices, and marine pollution (Hughes et al., 2017; Pandolfi et al., 2003). Mass coral bleaching events have already been observed in Australia's Great Barrier Reef, serving as an early warning sign of widespread ecological collapse (Hughes et al., 2018). Given these urgent environmental challenges, there is a growing need for scalable and automated solutions to monitor reef degradation. This project aims to address this gap by using machine learning techniques to classify coral health based on image data, with the additional goal of using our trained models to generate new labeled datasets from previously unlabeled sources.

3 Data

We partnered with researchers at the Woods Hole Oceanographic Institution to generate our own preliminary labeled dataset for training: 37 coral reef images manually annotated to yield approximately 800 individual coral masks (Woods Hole Oceanographic Institution, 2025). These images along with the annotation labels provided were then used to fine-tune the masking model. Additionally, we used this annotated data to train a preliminary model for classifying images for WHOI. The WHOI data consists of coral reef images in Majuro, Marshall Islands. These images were collected by the Yellowfin Surfzone ASV1, an autonomous surface vehicle specializing in gathering oceanographic data. Photographs are taken approximately every 10 seconds, with a range of 4,000 to 13,000 images a day, with a total of 25,000 images to process. Using these WHOI images, we apply our trained models to segment and classify real-world data, generating new labeled data sets for future coral monitoring efforts.

Our coral classification algorithm that we develop to classify the health of the coral was trained first on data from Hugging Face's Coral Health Classification Dataset (Esahit, 2023). We used this data as a proxy

*Brigham Young University. We claim all original work unless explicitly stated it was AI-generated. We used ChatGPT to help us write up documentation to code and convert written work to LaTeX such as tables and mathematical derivations.

for the WHOI data until we received labeled data with appropriate clearance. These data consist of labeled coral reef images categorized into three health conditions: healthy, unhealthy, and dead. Table 1 presents the summary statistics for the dataset, including the mean and standard deviation for each RGB channel, the proportion of white pixels, and the number of observations for each coral health category. While not particularly enlightening, these initial summary statistics show that, in general, saturation is correlated with the health of coral. The high amount of white pixelation (relative to healthy coral) may reflect the bleaching that occurs in unhealthy and dead coral classifications. We give a visual sample of the three classifications in the Hugging Face data set in Figure 1.

(a) Dead Coral (b) Unhealthy Coral (c) Healthy Coral

Figure 1: Sample images from the Hugging Face dataset across three health categories: dead, unhealthy, and healthy coral.

Additionally, we explore using the CoralMASK data set from CoralSCOP for segmentation training—an open-source collection of over 40,000 densely labeled coral images (Huang et al., 2023). Although we compare CoralSCOP's performance to other segmentation alternatives, we do not use the CoralMASK data set in our classification pipeline due to persistent formatting issues with the annotation files.

Category	Red	Green	Blue	White Pixel Proportion	Observations
Dead Coral	0.3625 (0.1443)	0.4710 (0.1556)	0.3898 (0.1521)	0.0158 (0.0570)	430
Unhealthy Coral	0.2740 (0.2477)	0.3807 (0.2700)	0.3842 (0.2732)	0.0368 (0.0454)	508
Healthy Coral	0.2554 (0.2521)	0.2935 (0.2328)	0.3206 (0.2632)	0.0091 (0.0272)	661

Table 1: Summary statistics for coral health categories based on rescaled (128 $\times$ 128) image pixel values. We report the mean of each RGB channel, with standard deviations in parentheses. The proportion of white pixels is defined as the percentage of pixels where all RGB channels exceed 200. Red, Green, and Blue values are standardized to the range [0,1] by dividing raw pixel values by 255.

4 Methodology

In this section we present our machine learning pipeline (see Figure 2). Our framework includes three main operations: Segmentation and Data Preprocessing, Model Training, and Ensemble Optimization. We present each one in turn.

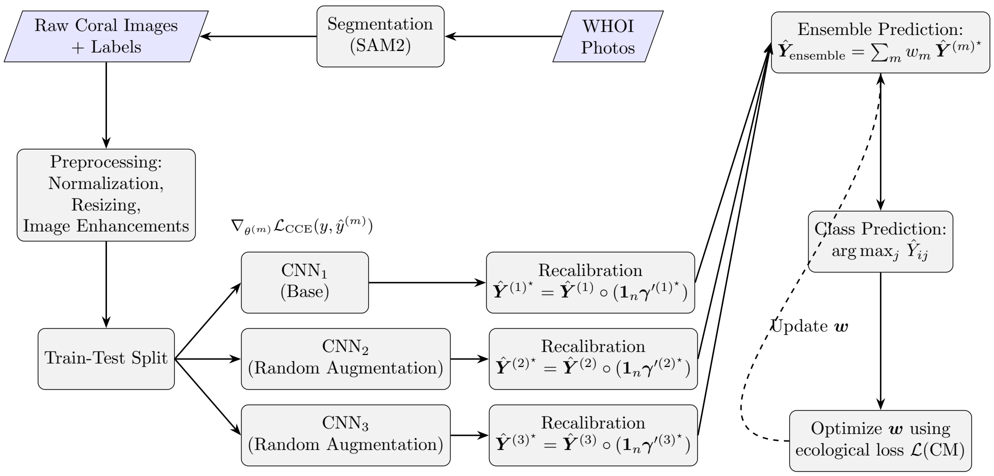


Figure 2: Overview of the Coral Classification Pipeline: raw images are preprocessed, split, and passed through three CNN models with differing augmentation strategies. Their outputs are ensembled using a weighted sum optimized to minimize an ecologically sensitive loss.

4.1 Segmentation

To enable our convolutional neural network (CNN) models to make health classifications on specific coral instances within an image, we isolated individual corals through image segmentation. Segmentation allows us to extract object-level annotations—pixel-specific masks—around discrete coral bodies, which are then passed to the classification pipeline. This process is essential given the visual density and frequently overlapping structure of coral formations. Additionally, we researched whether segmentation accuracy was dependent on depth, with room to further investigate dependencies on light and turbidity (vision clarity lost due to suspended particles).

We began by evaluating CoralSCOP, the leading segmentation model specifically developed for coral reef research. While CoralSCOP demonstrated significant value, it is built on the original Segment Anything Model (SAM1) developed by Meta AI, which has been succeeded by SAM2 (Kirillov et al., 2023). Unfortunately, CoralSCOP underperformed in our use case compared to even the baseline version of SAM2. Consequently, we adopted SAM2 as our primary segmentation tool.

SAM2 is a deep learning model based on transformer neural networks, designed for image segmentation. It was trained on over 30 million masks and excels at identifying object boundaries with minimal input. Compared to its predecessor SAM1, SAM2 offers improved performance and generalizability through enhanced mask prediction, a more responsive prompt-based system, and access to a larger training dataset. SAM2 is also modular and supports fine-tuning on specific domains, making it well-suited for specialized tasks like coral reef segmentation.

Because SAM2 is computationally intensive—especially when run at scale and across high-resolution imagery—we relied on GPU resources made available through BYU's Office of Research Computing. These enabled us to run the CUDA-dependent model and in the future will allow us to batch-process large quantities of data, including the CoralMASK database.

4.2 Data Preprocessing

After segmentation, we continue along the pipeline by constructing a dataset suitable for convolutional neural networks from the raw coral segments. By reducing the images down to blocks of $128 \times 128 \times 3$ RGB-valued data points, we build a lightweight, iterable, and memory-efficient abstraction. During this stage, we also extract annotations from the images to obtain a labeled training data set. Each annotation belongs to one of k # of distinct classes. For our purposes, we use training data obtained from the Hugging Face repository which consists of coral images belonging to $k = 3$ classes: $\{Dead, Unhealthy, Healthy\}$.

Next, we apply biologically motivated enhancements: We increase saturation and contrast to elicit the features of the coral that may prove useful in identifying whether a coral is bleached. We normalize the

RGB values to fall between $[0, 1]$ via per-pixel scaling. The resulting processed dataset consists of $\mathbf{X} \in \mathbb{R}^{n\times128\times128\times3}$ and integer labels $\mathbf{y} \in \{0, 1, 2\}^n$ corresponding to the ordinal ranking of our classes.

4.3 Model Training

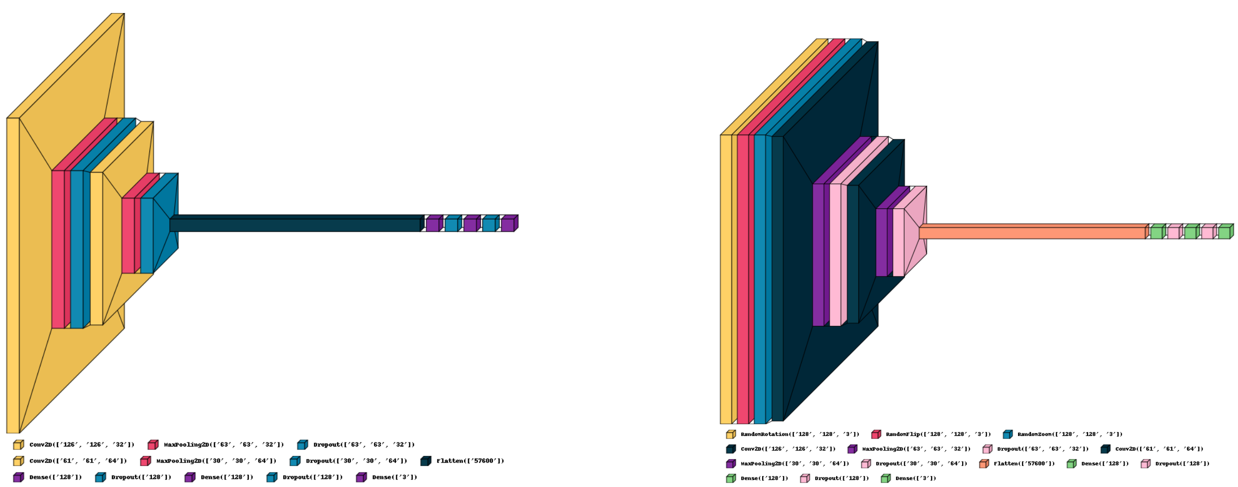
4.3.1 Training the CNN

To classify coral health status we trained a convolutional neural network. The model architecture comprises a sequence of convolutional and pooling layers followed by fully connected dense layers, and then finally using a softmax output for multiclass classification. The input layer receives the RGB images of size 128×128, which are processed through two successive convolutional blocks. Each block incorporates dropout regularization to mitigate overfitting and encourage generalization (see Figures 3a and 3b).

Following the convolutional part of the architecture, the feature maps are flattened and passed through a dense hidden layer with ReLU activation. A final dropout layer precedes the softmax classifier. We train the network using the Adam optimizer and minimize categorical cross-entropy loss. In the next section we explore an *ecologically sensitive* loss function. We prefer training our preliminary CNN models using a categorical cross-entropy (CCE) loss namely because its derivative exists: For a single observation with one-hot encoded true label $y = [y_1, \ldots, y_k]$ and predicted class probabilities $\hat{y} = [\hat{y}_1, \ldots, \hat{y}_k]$ where $\hat{y}_j = \frac{e^{z_j}}{\sum_{\ell=1}^{k} e^{z_\ell}}$ and z_j are the pre-softmax logits, the CCE loss[1] is defined as:

$$\mathcal{L}_{\text{CCE}}(y, \hat{y}) = -\sum_{j=1}^{k} y_j \log \hat{y}_j \tag{1}$$

To further regularize training and improve the model's robustness to natural variation in coral orientation and scale, we apply data augmentation to the input layer within the CNN itself. This includes randomized image transformations such as rotation, flipping, and zooming. Stochastic augmentations help us classify coral in settings where images are distorted through turbidity and low resolution.



(a) CNN_1 displays the architecture of our CNN without augmentation.

(b) CNN_2 shows the CNN architecture with data augmentation layers.

Figure 3: A visual depiction of our convolutional neural network models. Visuals generated from thee `visualkeras` package. Python code from Gavrikov (2020).

[1]The gradient of this loss with respect to the logits z_j is given by:

$$\frac{\partial \mathcal{L}_{\text{CCE}}}{\partial z_j} = \hat{y}_j - y_j$$

It can be shown, after an application of the chain rule, this gradient arises from utilizing the fact that $\sum_{j=1}^{k} y_j = 1$. The simplicity of this expression makes the CCE loss function well-behaved and also computationally efficient for deep learning optimization.

4.3.2 Loss Function

We propose a unique loss function for this project motivated by the ecological stakes of coral classification. Our goal is not necessarily to maximize overall accuracy but to minimize what we term *catastrophic ecological errors*—misclassifications that carry disproportionate ecological consequences. For instance, falsely classifying a healthy coral as dead may lead to misallocated intervention resources. Our solution to this was to increase sensitivity in the model by specifying our loss function.

To this end, we introduce an ecologically sensitive loss metric derived from a confusion matrix, CM $\in \mathbb{R}^{k\times k}$, and a $(k \times k)$ weight matrix Ω encoding the severity of each possible misclassification.

$\Omega =$

True \Predicted	Dead	Unhealthy	Healthy
Dead	ω_{11}	ω_{12}	ω_{13}
Unhealthy	ω_{21}	ω_{22}	ω_{23}
Healthy	ω_{31}	ω_{32}	ω_{33}

For a given $k \times k$ (3×3 in this case) confusion matrix (CM), We create an *ecologically sensitive* loss function ($L : \Delta_{k^2} \to [0, 1]$) by specifying ω_{ij} s.t. $\omega_{13} > \omega_{31} > \omega_{12} = \omega_{23} > \omega_{21} = \omega_{32} > \omega_{11} = \omega_{22} = \omega_{33} = 0$. Though we specify these weights specifically for the coral problem, this can be generalized to any classification problem of this sort.

$$\mathcal{L}(\text{CM}) = \frac{\sum_{i=1}^{k}\sum_{j=1}^{k} \text{CM}_{ij}\omega_{ij}}{(k^2-k)^{-1}\sum_{i=1}^{k}\sum_{j=1}^{k}\omega_{ij}} \tag{2}$$

The loss function[2] $\mathcal{L}$ serves as a *weighted accuracy* function. Note that L becomes the normal overall accuracy function for $\omega_{ij} = 1 \; \forall i \neq j$. Importantly, this loss function is not (really) smoothly differentiable with respect to model parameters due to the rigidity of the confusion matrix. While there may be suitable alternatives to the confusion matrix proposed here (such as perhaps a "soft" confusion matrix), we consider a way to minimize this loss function through an ensemble approach in the next section.

4.4 Ensemble Optimization

For this project, we trained three convolutional neural networks and combined the predictions to create an optimal ensemble CNN, although this could be generalized to include a greater number of models in the ensemble. Although individually, each CNN was trained through a computationally motivated CCE loss function, we consider here an approach to minimize the loss function proposed in the previous section.

4.4.1 Recalibration

Since each model was trained using a CCE loss function, we introduce our method here to *recalibrate* the model to approximately *resemble* a model that was trained on our desired (non-differentiable) loss function. While we do not assert that this is the only way to accomplish this task, nor is it perhaps the best way, we argue that it is computationally efficient and tractable from the perspective of optimization theory.

Suppose we omnisciently observe the outcome matrix, $\boldsymbol{Y}$, consisting of n observations and each observation consisting of a probability distribution governing the outcome of the random variable, Y_i, defined over Δ_k (although, as we will see, Y_i need not be a random variable at all; it could simply a collection of k elements). Also suppose we predict $\boldsymbol{Y}$ with a suboptimal matrix, $\boldsymbol{X}$, where we predict the probabilities of each outcome in Y_i with some other distribution defined over Δ_k. We propose that we can always achieve (weakly) a better prediction for $\boldsymbol{Y}$, denoted as $\hat{\boldsymbol{Y}}$ if we reweigh, and as we coin the term, *recalibrate*, the distribution governing the predicted outcomes of Y, Δ_k, by optimizing some weight vector $\boldsymbol{\gamma}$ over the space of $\boldsymbol{X}$.

Proposition. *Let $Y, X \in \mathbb{R}^{n\times k}$, and define $\hat{Y} = X \circ (\mathbf{1}_n\boldsymbol{\gamma}^\top)$ for $\boldsymbol{\gamma} \in \mathbb{R}^k$. The solution to*

$$\min_{\boldsymbol{\gamma}} \left\| Y - X \circ (\mathbf{1}_n\boldsymbol{\gamma}^\top) \right\|_F^2 \equiv \min_{\boldsymbol{\gamma}} \sum_{i=1}^{n}\sum_{j=1}^{k}(y_{ij} - x_{ij}\gamma_j)^2 \quad s.t. \quad \sum_{j=1}^{k}\gamma_j = 1$$

[2] Note that the integer constant $(k^2 - k)^{-1}$ is simply the normalizing constant to have $\mathcal{L}$ map to $[0, 1]$.

is uniquely given by

$$\gamma_l^* = \left(\frac{\lambda}{2} + \sum_{i=1}^{n} x_{il} y_{il}\right)\left(\sum_{i=1}^{n} x_{il}^2\right)^{-1}, \quad \textit{for } l = 1, \dots, k$$

where

$$\lambda = \frac{2\left[1 - \sum_{j=1}^{k}\left(\sum_{i=1}^{n} x_{ij} y_{ij}\right)\left(\sum_{i=1}^{n} x_{ij}^2\right)^{-1}\right]}{\sum_{j=1}^{k}\left(\sum_{i=1}^{n} x_{ij}^2\right)^{-1}}.$$

Proof. *See Appendix*

The implication is simple. We can *always* get better predictions by reweighting the estimated probability distribution post hoc to match more closely the classifications we observe. Note that we discuss a potential criticism of this method in Section 6.2. In practice, we obtain prediction matrices, $\hat{\boldsymbol{Y}}^{(1)}, \dots, \hat{\boldsymbol{Y}}^{(M)}$, for M number of total models we want in our ensemble (three in our project). Then, using the training data $\boldsymbol{Y}$, which only consists of the *realized* outcomes of $Y_i \in \{0,1\}^k,\ i = 1, \dots, n$, where $\sum_j Y_{ij} = 1\, \forall i$ we recalibrate our models by finding $\boldsymbol{\gamma}^{(1)}, \dots, \boldsymbol{\gamma}^{(M)}$ such that we minimize the ecologically sensitive loss function through out-of-sample k-fold cross-validation: We choose the set $\boldsymbol{\gamma}^{(1)^\star}, \dots, \boldsymbol{\gamma}^{(M)^\star}$ that best minimizes the left-out fold's ecological loss.

4.4.2 Ensemble Building

Once all M number ($M = 3$) CNN models are trained, we construct an ensemble classifier by forming a convex combination of the individual model predictions. Let $\hat{Y}^{(m)} \in [0,1]^{n \times k}$ denote the predicted probability matrix of model m for n samples and k classes. We define the ensemble prediction matrix, for a collection of scalars $(w_1, w_2, \dots, w_k), w_m \in [0,1]$, as,

$$\hat{Y}_{\text{ensemble}} = \sum_{m=1}^{M} w_m \hat{Y}^{(m)} \tag{3}$$

where, for convexity, we constrain,

$$\sum_{m=1}^{M} w_m = 1$$

The final predicted class for sample i is then given by,

$$\hat{y}_i = \arg \max_{j \in \{1,\dots,k\}} \hat{Y}_{\text{ensemble},ij}$$

To determine the optimal weights $\boldsymbol{w}^\star$, we minimize the ecological loss function $L(\text{CM}(\boldsymbol{w}))$,

$$\boldsymbol{w}^\star = \arg \min_{\boldsymbol{w} \in \mathbb{R}^M} L(\text{CM}(\boldsymbol{w})) \quad \textit{s.t.} \quad \sum_{m=1}^{M} w_m = 1$$

4.4.3 Optimization via Simulated Annealing

Because the ensemble prediction involves taking an $\arg\max$ over a weighted sum of softmax outputs, the resulting loss surface $L(\text{CM}(\boldsymbol{w}))$ is non-differentiable with respect to $\boldsymbol{w}$. Traditional gradient-based optimization methods such as standard Newton-Rhapson methods are thus ill-equipped for this situation[3]

To address this, we employ a stochastic global optimization algorithm that does not require a gradient computation. While there are many in this class of algorithms, we use a *simulated annealing*[4], implemented

[3]Note: Similarly to the recalibration algorithm proposed in the previous section, a lagrangian can be defined to minimize a Frobenius loss function (as we did with the recalibration). The resulting first-order conditions for a given weight, w_m, however, depend on the optimized parameters, $\boldsymbol{w}^\star_{(-l)}$, and thus require a value function iteration algorithm to solve the optimal weights—the simulated annealing algorithm may work just as fine in this case.

[4]From ChatGPT when prompted, *"Why is it called an annealing algorithm?"*: The algorithm simulates the annealing process in metallurgy, where a material is heated and then slowly cooled to allow it to settle into a minimum energy configuration. At high "temperature" levels, the algorithm is permitted to accept uphill moves with the objective of escaping local minima. As the temperature decreases, the algorithm becomes increasingly greedy, converging toward a local or global minimum.

via the `dual_annealing` function in `SciPy`. This algorithm combines a probabilistic search over the parameter space with a deterministic local optimizer. In the context of this problem, we choose the weight vector $\boldsymbol{w}$ that minimizes the left-out fold's ecological loss in a k-fold cross-validation routine.

5 Results

5.1 Results from Segmentation

We conducted a series of comparative evaluations between CoralSCOP and SAM2. These tests examined segmentation performance across a range of coral reef images varying in depth. The evaluations focused on three key metrics: precision, recall, and false positive count (FPC).

Initial segmentation with the CoralSCOP model yielded moderate performance. Precision scores were low across all depths—primarily due to the model's tendency to generate duplicate masks around the same coral instance. Recall was somewhat higher, but still inconsistent, as the model frequently missed individual corals, especially in deeper images where contrast and clarity were reduced. We present these results in Figure **??**. While we separated our evaluation by depth (0–3m, 3–5m, and 5m+), an ANOVA test revealed no statistically significant differences in performance between these groups—likely a result of small sample size and potential inconsistency in our hand-labeled grading. One key limitation of our evaluation process is that none of our team members are trained marine biologists; labeling errors, especially in complex scenes or deeper reef photos, may have affected our count accuracy.

Following CoralSCOP's limitations, we transitioned to the Segment Anything Model 2 (SAM2). Initial visual analysis suggested a noticeable improvement: SAM2 avoided the duplicate issue and seemed to capture a greater number of distinct coral bodies. These observations informed our hypothesis that SAM2 would outperform CoralSCOP in recall, but might increase the rate of false positives due to its not being specialized for coral identification. These early comparisons were primarily qualitative (Figure 4)

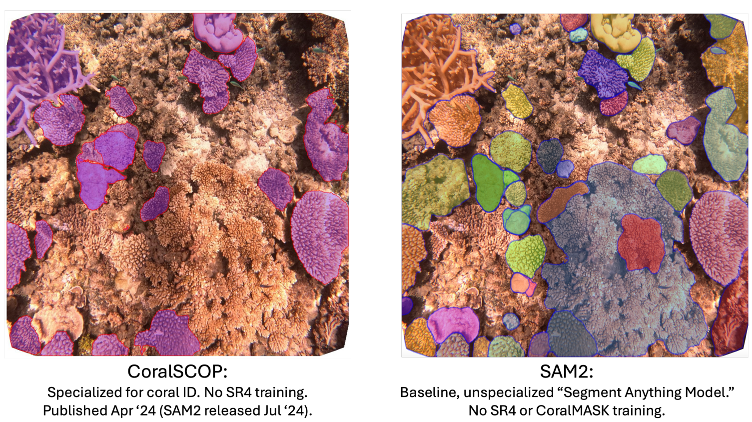


Figure 4: Side-by-side visualizations CoralSCOP and SAM2 mask outputs

To further improve SAM2's domain-specific accuracy, we fine-tuned the model using a small labeled dataset (35 images, 875 coral masks) created in collaboration with the Woods Hole Oceanographic Institution. Due to persistent formatting issues with the CoralMASK dataset, this WHOI dataset was our only source of annotated training data. Fine-tuning led to modest improvements in model behavior. Tensor Board loss curves indicated that training was directionally effective—evident by downward trends—but also highly variable, perhaps due to the limited quantity and diversity of training data (Figure 9).

Our final comparison shows that SAM2 outperforms CoralSCOP in both precision and recall, with the most notable gains in recall (see Figure 6). However, this comes at the cost of an increased false positive count (FPC), suggesting that SAM2 (at the current training status) over-segments background elements and non-coral structures when uncertain (see Figure 5).

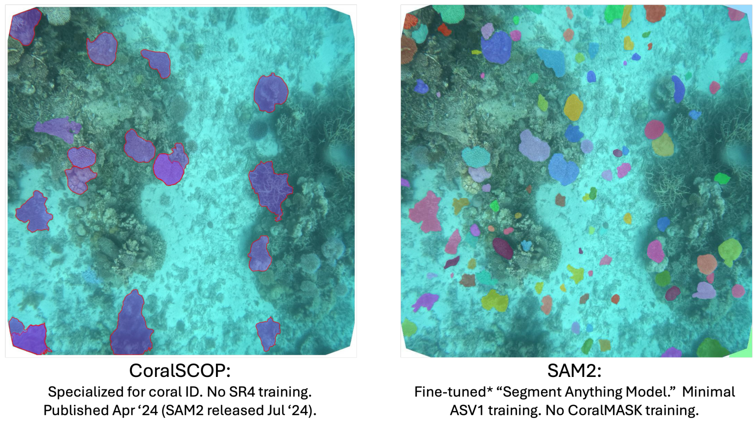


Figure 5: Side-by-side segmentation maps visualizes tradeoffs: CoralSCOP tends to miss corals, while SAM2 often oversegments unrelated features.

While the performance of the current loading of SAM2 is promising, we believe further improvements are achievable with larger, high-quality labeled datasets. Future work should include resolving issues with the CoralMASK formatting and increasing the quantity and diversity of labeled data from WHOI to better represent edge cases and unusual coral types.

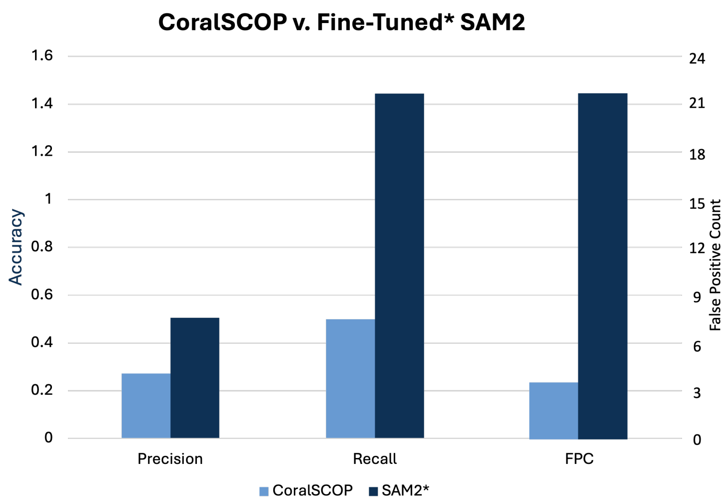


Figure 6: Comparison of segmentation model performance across three metrics—precision, recall, and false positive count (FPC). Fine-tuned SAM2 significantly outperforms CoralSCOP in recall and precision, but with a much higher incidence of false positives. Again, discrepancies in hand grading photos likely compromised some of these measures, as evidenced by the Recall value for SAM2 being > 1.

5.2 Results from Hugging Face Training Data

The labeled Hugging Face data categorized coal heath into three labels (Healthy, Unhealthy, Dead) as discussed. The confusion matrix below (Figure 7) shows the respective results of our machine learning model on the out-of-sample test set (as the training data results approach 100 percent accuracy). The greatest discrepancies are dead coral incorrectly labeled as unhealthy and healthy coral labeled as unhealthy. Our model achieved a 94.7% out-of-sample accuracy and 94.99% out-of-sample weighted accuracy (corresponding to our ecological loss function) for the Hugging Face data.

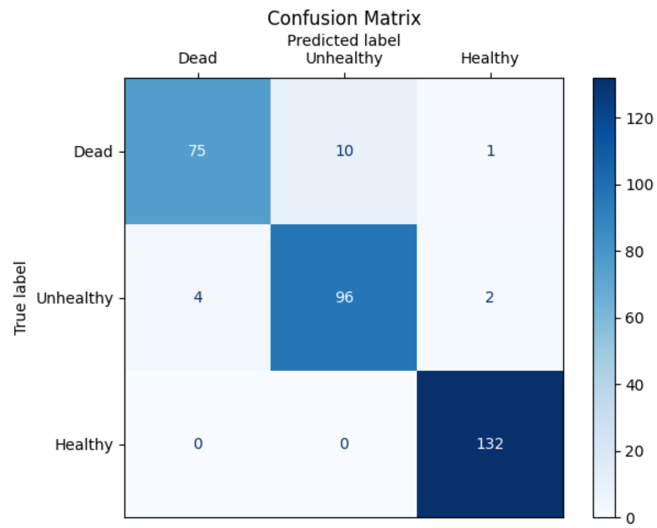


Figure 7: Confusion matrix on three ordinal classifications using our optimized ensemble model comprised of three convolutional neural networks.

5.3 Results from Labeled WHOI Training Data

The WHOI data we received categorized coral into only two bins: healthy and unhealthy (bleached coral). Since our original model was trained using the Hugging Face proxy data, our pipeline as used with the WHOI data may be suboptimal. Processing the WHOI data through the pipeline still functionally sorts the masked data. The out-of-sample confusion matrix depicts a 82.9% overall accuracy and 93.59% weighted accuracy for the WHOI data (Figure 8).

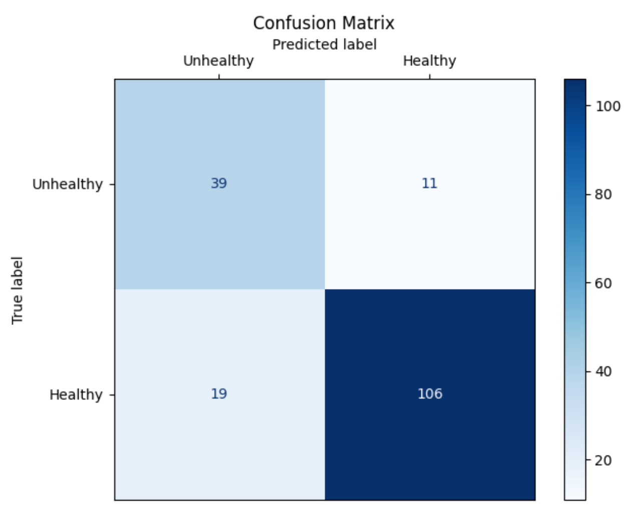


Figure 8: Predictions for the model trained on the WHOI data. Trained on 700 images; predicted on the remaining 175.

6 Conclusions

6.1 Main Findings

With our proposed machine learning pipeline, we take raw images taken by an autonomous research boat, modify them to fit into our algorithms, mask the existing corals, and then classify them as healthy or unhealthy. This process answers our question of how machine learning models can be used to monitor coral degradation. Through iterating through our segmentation process and classification algorithm, we trained our models to be ecologically sensitive to the various levels of misclassification error corresponding to the health of the coral. We recognize that our final model was trained on the sparsely available, hand-labeled WHOI images, and thus we propose several avenues for future research. In addition to the modeling suggestions already discussed throughout, such as adding more CNNs to the ensemble, increasing model depth, and increasing regularization, future research will evaluate our proposed machine-learning pipeline on more annotated data from WHOI over a larger classification space (potentially up to four or more classes in addition to the 'unhealthy' and 'healthy' classifications). We are confident that with greater refinement and more training data, our model will be an effective and efficient replacement for current hand-labeling methods. We are hopeful that our model will augment scientists' reef conservation efforts.

6.2 Comments Regarding our Optimization Methods

We proposed a *recalibration* algorithm to optimize individual models and calibrate the model predictions towards optimizing an *ecologically sensitive* loss function. Why this may be novel in its own right, we still would regrettably admit that the rigidity in the loss function and its non-differentiable nature make it hard to work with. As a result, we were forced to create appendices to our ML pipeline in a post hoc fashion, making our pipeline perhaps more complicated than it needed to be. Going forward, we plan to define a smooth loss-function that will be both computationally tractable and able to maintain the ecological sensitivity that our loss function accomplishes.

A valid critique of our recalibration technique is that the loss function that each $\boldsymbol{\gamma}$ vector minimizes is the squared Frobenius loss function which induces a higher loss the less accurate our predictions are. In other words, recalibrating the model, as we present it, does *nothing* explicitly directed towards solving the issue of minimizing ecological loss. Only through a cross-validation technique do we choose $\boldsymbol{\gamma}$ that minimizes the out-of-sample ecological loss. This is only partially beneficial. We are simply choosing the best $\boldsymbol{\gamma}$ vector that, by construction, was chosen to maximize accuracy (not minimize ecological loss). The resulting $\boldsymbol{\gamma}^\star$ is the vector that just so happens to minimize ecological loss within the synthetic out-of-sample folds. Hence, while we proved $\boldsymbol{\gamma}^\star$ minimizes a uniform accuracy loss, it is only a local minimum over the space of ecological loss.

6.3 Code

All code for this project is well-documented in our GitHub repository[5]. On the GitHub you will find the `READMD.md` that gives instruction on how to reproduce our models and main results. We note here that due to data privacy we do not include the raw image files generously obtained from WHOI.

6.4 Acknowledgments

We want to give a special thank you to Calvin Quigley and Anne Cohen from the Woods Hole Oceanographic Institution for motivating this project and giving us guidance in the work they do. We also thank our machine learning professor, Dr. Brigham Frandsen[6], for providing the framework for starting this project.

[5]See https://github.com/SamLeeBYU/CoralReefClassification if the link doesn't work.
[6]Professor of Economics, Brigham Young University

7 Appendix

7.1 Proof for Section 4.4.1

Proof. *Define the Lagrangian:*

$$\mathcal{L}(\boldsymbol{\gamma}, \lambda) = \sum_{i=1}^{n} \sum_{j=1}^{k} (y_{ij} - x_{ij}\gamma_j)^2 + \lambda \left(1 - \sum_{j=1}^{k} \gamma_j\right).$$

Taking first order conditions,

$$\frac{\partial \mathcal{L}}{\partial \gamma_l} = -2 \sum_{i=1}^{n} x_{il}(y_{il} - x_{il}\gamma_l) - \lambda = 0,$$

$$2\gamma_l \sum_{i=1}^{n} x_{il}^2 - 2 \sum_{i=1}^{n} x_{il} y_{il} = \lambda,$$

$$\gamma_l^\star = \left(\frac{\lambda}{2} + \sum_{i=1}^{n} x_{il} y_{il}\right) \left(\sum_{i=1}^{n} x_{il}^2\right)^{-1}.$$

$$\frac{\partial \mathcal{L}}{\partial \lambda} \implies \sum_{j=1}^{k} \gamma_j = \sum_{j=1}^{k} \left(\frac{\lambda}{2} + \sum_{i=1}^{n} x_{ij} y_{ij}\right) \left(\sum_{i=1}^{n} x_{ij}^2\right)^{-1} = 1.$$

$$\implies \frac{\lambda}{2} \sum_{j=1}^{k} \left(\sum_{i=1}^{n} x_{ij}^2\right)^{-1} + \sum_{j=1}^{k} \left(\sum_{i=1}^{n} x_{ij} y_{ij}\right) \left(\sum_{i=1}^{n} x_{ij}^2\right)^{-1} = 1.$$

$$\implies \frac{\lambda}{2} = \frac{1 - \sum_{j=1}^{k} \left(\sum_{i=1}^{n} x_{ij} y_{ij}\right) \left(\sum_{i=1}^{n} x_{ij}^2\right)^{-1}}{\sum_{j=1}^{k} \left(\sum_{i=1}^{n} x_{ij}^2\right)^{-1}}.$$

Substitute into $\gamma_l^\star$ to obtain the closed form. Since γ_l was an arbitrary element $\in \boldsymbol{\gamma}$, this holds for any $l = 1, \ldots, k$.

Now, to show that $\boldsymbol{\gamma}^*$ is a global minimizer, we verify the second-order conditions. The Hessian of $\mathcal{L}$ with respect to $\boldsymbol{\gamma}$ is diagonal, with entries,

$$\frac{\partial^2 \mathcal{L}}{\partial \gamma_j^2} = 2 \sum_{i=1}^{n} x_{ij}^2, \quad \textit{and} \quad \frac{\partial^2 \mathcal{L}}{\partial \gamma_j \partial \gamma_\ell} = 0 \textit{ for } j \neq \ell.$$

Let $H = \nabla^2_{\boldsymbol{\gamma}\boldsymbol{\gamma}} \mathcal{L} = \operatorname{diag}(2\sum_{i=1}^{n} x_{i1}^2, \ldots, 2\sum_{i=1}^{n} x_{ik}^2)$. The constraint Jacobian is $A = [1 \ \cdots \ 1] \in \mathbb{R}^{1\times k}$. Its nullspace is

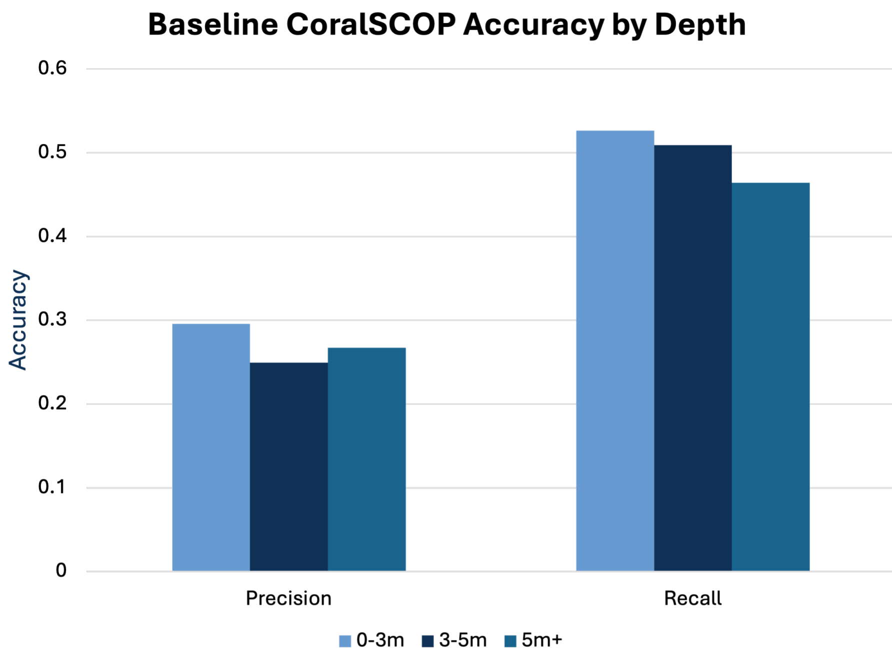
$$\mathcal{N}(A) = \left\{ \boldsymbol{v} \in \mathbb{R}^k : \sum_{j=1}^{k} v_j = 0 \right\}$$

For any $\boldsymbol{v} \in \mathcal{N}(A)$, we compute

$$\boldsymbol{v}^\top H \boldsymbol{v} = \sum_{j=1}^{k} 2 \left(\sum_{i=1}^{n} x_{ij}^2\right) v_j^2 \geq 0.$$

Hence, H is positive semi-definite on the nullspace of the constraint, and the second-order conditions for equality-constrained minimization are satisfied. Therefore, $\boldsymbol{\gamma}^*$ is a unique minimizer. □

7.2 Additional Figures



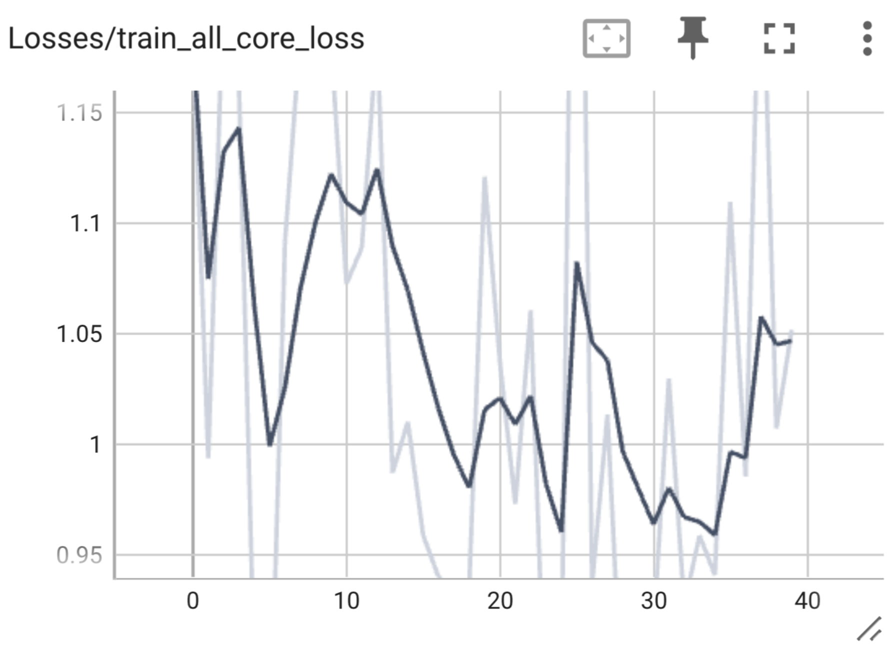


Figure 9: SAM2 loss curve during fine-tuning. High variance highlights the need for larger, more diverse annotated training sets.

References

Burke, L., Reytar, K., Spalding, M., and Perry, A. (2011). *Reefs at risk revisited*. World Resources Institute.

Esahit (2023). Coral health classification dataset. Accessed: March 15, 2025.

Gavrikov, P. (2020). visualkeras. https://github.com/paulgavrikov/visualkeras.

Huang, Y., Wu, T., Wang, D., and Zhang, Z. (2023). CoralSCOP: Segment any coral image on this planet. https://coralscop.hkustvgd.com/CoralSCOP_files/CoralSCOP__Segment_any_COral_Image_on_this_Planet.pdf. Accessed: 2025-04-11.

Hughes, T. P., Barnes, M. L., Bellwood, D. R., Cinner, J. E., Cumming, G. S., Jackson, J. B., Kleypas, J., van de Leemput, I. A., Lough, J. M., Morrison, T. H., et al. (2017). Coral reefs in the anthropocene. *Nature*, 546(7656):82–90.

Hughes, T. P., Kerry, J. T., Baird, A. H., Connolly, S. R., Chase, T. J., Dietzel, A., Eakin, M. C., Heron, S. F., Hoey, A. S., Hoogenboom, M. O., et al. (2018). Global warming transforms coral reef assemblages. *Nature*, 556(7702):492–496.

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., C. Berg, A., Lo, W.-Y., Dollár, P., and Girshick, R. (2023). Segment anything. *arXiv preprint arXiv:2304.02643*.

Pandolfi, J. M., Bradbury, R. H., Sala, E., Hughes, T. P., Bjorndal, K. A., Cooke, R. G., McArdle, D., McClenachan, L., Newman, M. J., Paredes, G., et al. (2003). Global trajectories of the long-term decline of coral reef ecosystems. *Science*, 301(5635):955–958.

Spalding, M., Burke, L., Wood, S. A., Ashpole, J., Hutchison, J., and zu Ermgassen, P. (2017). Mapping the global value and distribution of coral reef tourism. *Marine Policy*, 82:104–113.

Woods Hole Oceanographic Institution (2025). Woods hole oceanographic institution. https://www.whoi.edu. Accessed: 2025-04-11.