# Capstone Project Proposal

## Nature Conservancy Fisheries Monitoring

This project is based on an active competition on Kaggle that ends April 12, 2017 -- https://www.kaggle.com/c/the-nature-conservancy-fisheries-monitoring. The information provided in the background and problem statement and objective sections are summarized from the competition description.

## Background

The oceans supply a large but limited resource of protein for the world's population. A large fraction of the fish extracted and consumed are harvested from the Central and Western Pacific.[1] In order to provide good stewardship of this resource, alliances have been struck between many countries to regulate fishing so that excessive harvesting does not deplete the fish population unsustainably and thereby pose a threat to fragile marine ecologies or local fishing economies.

While signed treaties are a good first step, implementing these regulations poses a big challenge. Capturing images of the catch off-loaded at every port can provide a good amount of data about the size, species and quantity being harvested; however, processing these images manually would prove to be an expensive and time consuming task and makes the deployment of video surveillance systems a non-scalable solution.

The sponsors of the competition are seeking the help of the ML community to create an automated system for processing images to speed up the video review process so that surveillance systems can be deployed world-wide to ensure a sustainable future for all.

## Problem Statement

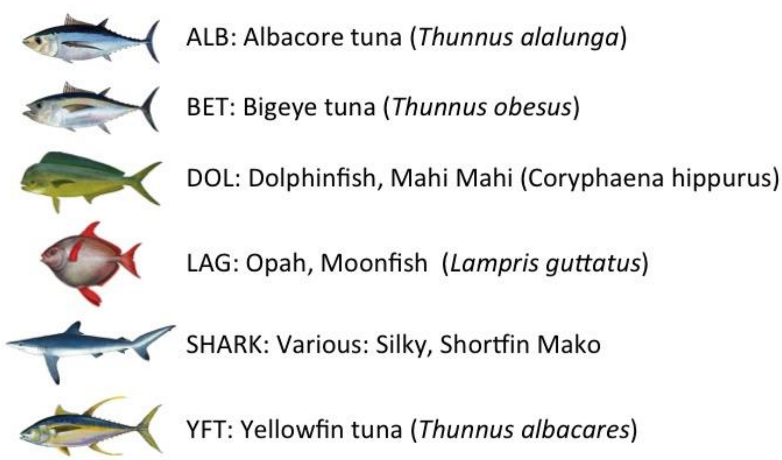
The challenge for this competition is to identify a computational method that can reliably detect and identify (classify) species of tuna or sharks that were captured in a set of 1000 test images. The images can contain no recognizable fish (too small) or other species of fish that are also part of the catch but not regulated.

For each test image, the probability of it belonging to one of the eight classes will be calculated. The computed multi-class log loss over the set of all test images will be the single-valued performance metric that will be used to gauge the success of the proposed algorithm.

## Datasets

The dataset made available for the competition was compiled by The Nature Conservancy in partnership with Satlink, Archipelago Marine Research, the Pacific Community, the Solomon Islands Ministry of Fisheries and Marine Resources, the Australia Fisheries Management Authority, and the governments of New Caledonia and Palau. Test and training datasets have been provided separately. Since we will not have access to the labels (actual classes) for the test set, we will use only the training set images for this project.

The training set consists of nearly 4000 images grouped into eight folders with varying number of images for each class. The folder designates the class label. The images are to be classified as one of eight classes – six of them drawn from a short list of species to be identified (see images below), one class for images where there were no fish present (or small fish) and one class for images where a fish was identified but did not match one of the six species of tunas and sharks.




Fish images are not to scale with one another

## Solution

Convolutional neural networks (conv-nets) have been very successful in recognizing objects in an image. In this project we will apply conv-nets and other supervised and pick the one that provides the best reliability.

## Benchmark

The competition provides a sample submission file where each image is tagged with the same set of probabilities that the image belongs to a particular class - 0.45500264690312336, 0.05293806246691371, 0.03096876654314452, 0.017734250926416093, 0.12308099523557438, 0.07914240338803599, 0.046585494970884066, 0.1942826892535733. Since this baseline is for the set of test images, it is most likely the random probability that the image is from a particular class in the test set and would not be a good baseline for this project where we are only using images from the training set. For our baseline, we will use the probability of an image belonging to the particular class as the probability that an image selected at random from the training set belongs to that class.

I will submit the predictions on the test set from the competition from our final model. The public leaderboard for the competition will also provide additional feedback on the ranking of my solution relative to other teams.

## Evaluation Metrics

The multiclass logarithmic loss function is given by:

$$logloss = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} yij \,.\, \log(pij)$$

where N is the number of images in the test set, M is the number of image class labels, $y_i$ is the vector-valued true label (a value of 1 for $y_{ij}$ implies that the *i*-th image belongs to the *j*-th class) and $p_{ij}$ is the probability (predicted) that the *i*-th image belongs to the *j*-th class.

## Project Design

There has been a lot of previous work done on object recognition in images including several other projects and previous competitions on Kaggle.[2-6] This project falls into the same broad category of problems.

The challenge in this project lies in differentiating between some classes where the species look very similar (albacore vs. big eye tuna), varying camera angles and partial profiles (image of fish obscured by other objects, carcasses with head removed).

I will explore the application of deep learning techniques (convolutional networks) to this problem using the TensorFlow and Keras libraries[7] and attempt to design a conv-net that gives the highest accuracy (lowest logloss) varying elements such as number of nodes, hidden layers, and activation functions. We will also compare the results obtained by some well-known networks (LeNet[4], AlexNet[7], VGG16[8]) applied to this problem.

## References


1. Kaggle Competitions - https://www.kaggle.com/c/the-nature-conservancy-fisheries-monitoring/data.
2. Kaggle Competition - https://www.kaggle.com/c/leaf-classification
3. Kaggle Competition - https://www.kaggle.com/c/dogs-vs-cats-redux-kernels-edition
4. Implementing LeNet with python and Keras - http://www.pyimagesearch.com/2016/08/01/lenet-convolutional-neural-network-in-python/
5. "*Object Recognition with Gradient Based Learning*", Yan LeCun, Patrick Haffner, Leon ottou and Yoshua Bengio, http://yann.lecun.com/exdb/publis/pdf/lecun-99.pdf
6. "*ImageNet Classification with Deep Convolutional Neural Networks*", Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, University of Toronto, http://www.cs.toronto.edu/~fritz/absps/imagenet.pdf
7. AlexNet - http://www.cs.toronto.edu/~guerzhoy/tf_alexnet/
8. VGG16 - http://www.cs.toronto.edu/~frossard/post/vgg16/