Solution Design

Solution Description

With the main objective of our proposed system being object detection there are three main objectives that we need to achieve.
Firstly processing the data that we are going to be using will mainly consist of breaking the video based datasets into frames, then adding the right annotations to each frame, this usually includes highlighting the desired object, any extra processing will depend on the model chosen for the training. As for the testing data set there is a lot less processing,at most converting the data set from videos to images. The steps mentioned above only cover a base implementation as the data sets can be modified further in order to better test / train the model this modifications can be anything from changes in image brightness to adding noise to certain parts of the image all this data manipulation can lead to improved performance they are however a consideration for the later stages of development.
Secondly, choosing or building a model that works best with the task at hand. In this situation we would firstly have to consider which type of model we will be using and after this if we want to adapt an existing open source model in order to accomplish our objective or if we want to build it from the ground up. We settled on using a model based on deep learning more specifically a model based on YOLO v3.
Thirdly, the training and testing phases. which will consist of training the model on the aforementioned annotate images the testing performance for a deep learning model is usually measured by average loss with lower values being better.This will result in a set of weights that will then be used for testing on a data set that has not been seen by the model yet. The performance of the testing will be measured in the success rate and the precision rate of the model the

success rate representing the amount of objects successfully identified whilst the precision rate is a representation of how well the object is framed by the model.

Dataset Description

With the the proposed system being designed for underwater fish detection the datasets that we need to do our models training and testing on has to include firstly a large variety of fish species as well as other marine wildlife. Secondly there in order to ensure our systems robustness we need to account for multiple underwater environments with different levels of visibility, different backgrounds and different lighting conditions, another consideration especially when testing should be having images with clutter or images where no target objects are present. In terms of choosing the datasets one of the more relevant and widely used set for marine wildlife is Fish4Knowledge which gives a 3 year view of coral reef habitats of the coast of Thailand having monitored over 3000 species of fish this is the most extensive dataset when it comes to underwater environments. When it comes to our model the main drawbacks this set has are its sheer size, due to hardware limitations we will only be able to process a fraction of the entire set, as well as the fact that all the recordings take place in similar environments, all being focused on habitats near coral reefs. Another dataset that we will be considering is Marine Habitats and Species available on data.gov.uk under an open license, which is a collection of record of habitats and species issued by Natural England and collected by Defra. The inclusion of this dataset is done in order to diversify the training and testing data both in terms of the recorded species as well as the environments present in the scenes. When it comes to considerations since we are limited to using mainly open sourced datasets we may not be able to train our model on every type of underwater environment, this is especially true for recordings taking place in flowing waters, as they represent a perfect testing environment due to the high velocity of the objects and the opacity of the waters, testing and training on these will greatly improve the robustness of the model but such datasets are harder to produce thus harder to come by.

Solution Motivation

The motivation behind using a deep learning model mainly comes from studying other underwater based object detection models in our literature review where we found out that models that use and adaptation of YOLO v3 have great results and are somewhat easy to implement, YOLO v3 is especcialy god at identifying moving targets which makes it a great choice for underwater marine wildlife detection. Another consideration when it comes to the model is using a mixed model this can help overcome certain drawbacks, such as not being able to identify an object when collision occurs. There are quite a few drawbacks that come with using model based on neural networks mainly they have to do with overfitting the data and the time they need for training, the first point is a non issue in our case since there are plenty of data sets recording underwater habitats and wildlife such as the ones mentioned in our dataset description and when it comes to performance using Google Colab can greatly speed up the training process. One more drawback that needs to be mentioned when it comes to neural networks is not being able to understand the reasoning behind the predictions made and how each layer influenced the final result, although this is a major consideration when deploying such models in certain fields due to the nature of the problem at hand being mostly risk free this is cause for concern only when trying to optimize the hidden layer.
When it comes to implementing the model we will most likely be using Keras as it is used by most projects that incorporate YOLO V3 along with this in terms of image processing we will be defaulting either to OpenCV or Scikit-image. Another positive aspect to using a deep learning model is the flexibility of weights as they can be used to "kick start" a model and also by storing them they can be used to easily roll back to an earlier state of development if needed.