Using AI Enabled Robotic Fish to Combat Environmental Waste

Matthew R. Ganis, Matthew Crespo, Matt Brown, Vineetha Alluri,
Robert Consigliere, Janaya Cooper, Avery Leider, and Charles C. Tappert
Pleasantville, NY, USA
{mganis, mc90468n, mb18974p, jc45506p, va89781, rc58462n, aleider, ctappert}@pace.edu

***Abstract*—The combination of Robotic Fish and Artificial Intelligence (AI) is becoming one of the most prominent research discoveries of our time. The advanced technology, software engineering, big data and extensive trials has led to their success. Many robotic fish research are being found all over the web. Each helps cater to our environmental, cyber warfare, and machine learning needs. However, Robotic Fish are not to be linked to underwater drones, but are linked to reducing of global warming, greenhouse gases, and the extinction of other underwater life. Robotic Fish and Artificial intelligence provide a special outlook on intellectual implementation results for educational resources.**

***Index Terms*—Robotic Fish, Artificial Intelligence, Software Engineering, Big data, Machine Learning, Amazon, Underwater Environmental Clean up, Cleaner Fish, IoT**

I. Introduction

The most important thing to do in order to address underwater environment waste is try to better understand the environment we are in. As we are in an environmental crisis, different environmental factors contribute to the cleaning the underwater habitat for underwater life. When placed in a body of water, robotic fish can gather and provide us with different kinds of information about the waste from its surroundings. For example, a swarm of fish can be dispatched into a large body of water to monitor different fish behavior, water quality, and underwater pollution. Scientists can then collect and analyze this data to help us better understand environmental monitoring. There are currently a variety of robots being used in the oceans. Robots with artificial intelligence are important, and they provide us with a wide range of advantages; they help reduce the loss of human life in risky environments, and provide quality assurance, efficiency, and consistency. Liquid robotics, a startup in San Francisco, has created wave-riding robots which can be used to track fish, Predict tsunami's and discover oil leaks around drilling platforms. A prototype of robot fish has been created to detect pollution, find out pH of water and report to a computer. [1] Robots are also used for large scale water quality monitoring, environment monitoring and event detection [2]. Richard Hardiman of RanMarine is using small robots in Rotterdam called waste sharks that catches garbage before it works it's way into the sea [3]. Robots are also used to facilitate deeper understanding of marine life with minimally disruptive observations. SoFi, an untethered soft-bodied robotic fish that demonstrates prolonged and consistent underwater remote-controlled operations was built by MIT researchers [4]. Our research is to design, develop and control a robotic fish using artificial intelligence and analyzing it's environmental cleaning behavior when it is subjected to a test in an artificial water body.

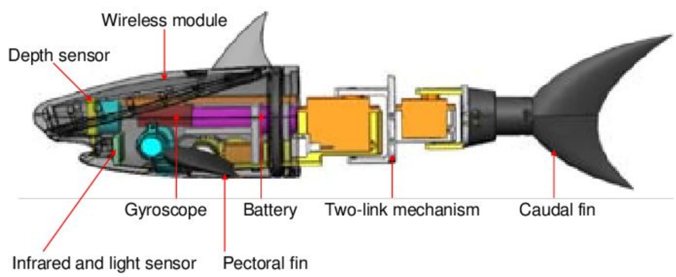

Figure 1: Conceptual design

II. Literature Review

A. Current Design and Model Configurations for AI Enabled Robot Fish

1) Experimental Modeling and Motion Control for Underwater Robotic Fish [5]: The main concept of the AI project is to combine and implement intelligence for education through Machine learning, Software Engineering, Artificial Intelligence, IoT, Big Data, and environmental clean up. This gives an importance and reasoning behind keeping the waters safe because a hand full of the aquatic animals are eating the waste that is being thrown into the waters and are dying from it. This is why there is less fish and underwater mating being present. The uncleanliness of our environment is disgusting and as a result, it impacts human resources as well. The fact is that, we have to make our environment not only clean for the human species but for the underwater life as well.

Figure 2: Robotic prototype

In order to get a proper environmental cleanup, we modeled the robotic fish after the cleaner fish. Cleaner fish are known to have a cleaning behavior. Although their behavior consists of cleaning other fish by eating infections, parasites, and dead skin tissue. However, our Robotic fish would grasp a similar concept of the cleaning behavior, which allows it to clean the underwater environment for the other fish to live in a toxic free habitat. The majority of concern when it comes to designing a 3D robotic fish is its technology. Special algorithms and programming languages like Python help with the movement and the simplified robotics of the design [5]. The robotic fish's movement will be monitored through a remote camera on the fish's fin. The findings will be analyzed on the computer to enable the results and see how it functions using its body to propel and motion itself in the water in order to retrieve the underwater waste. The data collected will be based on the 3D robotic fish's interaction with other fishes as well, including its ability to camouflage itself when danger is near, and how much time and speed it can acquire to clean up certain sections while underwater. This would depict if the robotic fish has what it takes to clean the underwater pollution for other fish to live safely in their "natural" habitat.

Artificial intelligence is one of the most complex subjects and has many difficulties when being applied to robotics. The artificial intelligence will be encrypted on a chip and inserted within the head of the robotic fish. Therefore, when applying artificial intelligence to the 3D robotic fish, there needs to be careful caution with its necessity. It is easy to replace the robotic fish, but developing the artificial intelligence is not so easy. The Robotic fish's artificial intelligence will provide its nature of posing as a fish, but functioning as a real "cleaner fish". The robotic fish would become one with nature to provide cleaner waters for the natural habitat. The mechanical system and its now "fish instinct" will help it to adapt and evolve like other fishes. The 3D robotic fish would have the potential to dive about 3 meters under water to enable its fish like movement [6]. Any time of motion patterns will then be included in the data research. There would also be a GPS on the tail of the robotic fish to ping each location the fish swim to. Also by creating a formula and using the GPS the number of meters the robotic fish swam will also be used as significant data [7]. Furthermore, these results will be used as an education guide for University based research and experiment.

2) Analysis of Robotic Fish Using Swarming Rules with Limited Sensory Input [8]: Sensory features are going to be a main benefit to building a robotic fish. Fish are known to depend on their skill to sense the environment around them and to swarm together with other fish to strengthen their chance of survival in wild sea life. For robotics, researchers have tried to copy fish's ability and learn the environment around them to make the robotic fish as realistic (behavior wise) as possible. The effects of their different sensory capabilities have been paired with a few basic swarming "rules" to figure out what the robotic fish will need to not only interact with by act like real fish. The goal of this research is to create an environmental-like fish that will eventually evolve to monitor and study life of fish underwater. Although it would have been beneficial to have and test a physical model/prototype, because of time and cost constraints, a virtual environment and model was done to further the experiment. Having the robotic fish created and tested in a virtual simulation permits limitations like; which sensory skill will be used at any given time and how quickly it can adjust to switching between the skills necessary for the situation or environment the robotic fish I placed in. Creating the testing subject and environment virtually has also presented the advantage of being able to make modifications to both the design of the fish and the interacting or sensory features much simpler. By testing the different sensory types that cleaner fish have on the robotic fish, it shows how effective the robot fish will be in blending in with real swarms of fish and how they respond to a controlled and continually changing environment.

3) Insight of a Six Layered Neural Network along with other AI Techniques for Path Planning Strategy of a Robot [9]: It is key to find a suitable artificial intelligence algorithm to steer our robotic fish in familiar/unfamiliar territories. Several prior researches show us that different techniques can be used for robot maneuvering. Layered artificial neural networks can be used so that the input layer captures the distance between the fish and its surroundings and the output layer gives the fish a steering angle in which it should move. [9] Central pattern generators are also used to control postures of robot fish. CPG is a biological neural network that can generate motion pattern used to control fish postures [10]. Achieving fish-like capabilities in maneuverability and efficiency in an automotive vehicle will require the tight integration of both design and control development. [11]

B. Cooperative Optimal Collision Avoidance Laws for a Hybrid-Tailed Robotic Fish [12]

Collision avoidance is an important feature for developing a robot fish. This paper discusses how a hybrid tailed robotic fish addresses collision avoidance using the collision cone approach [13] [14] [15] [16]. It is a one-step look-ahead method suitable for online implementation.

In this paper, Collision cone approach for two arbitrarily shaped objects is disused and then a cooperative optimal guidance laws for collision avoidance are derived. Within the collision cone approach Lyapunov-based methods are employed to determine analytical expressions of nonlinear guidance laws with which cooperative collision avoidance can be achieved. The structure of the fish described is unique. The caudal fin of the hybrid fish is driven by a double jointed mechanism. One joint is driven by an actuator that allows for precise control of angular or linear position, velocity and acceleration. The other joint is driven by an ionic polymer-metal composite actuator which is called an artificial muscle. A NodeMCU WiFi microcontroller is used to control the fish and communicate with the station. A battery is used to provide electricity to the robotic fish, and an H bridge delivers the IPMC voltage. The servomotor performs a constant amplitude flapping. To control the fish remotely, the control signal is transmitted by the user datagram protocol (UDP) between the control station and onboard microcontroller at a frequency of 5 Hz.

1) Model Predictive Control for Underwater Robots in Ocean Waves [17]: Based on the research paper "Model Predictive Control for Underwater Robots in Ocean Waves" [17], wave displacement and water disturbances can be effectively resisted and even counteracted with the implementation of the model predictive control method (MPC). The component fluid dynamics underneath ocean wave fields can be forecasted/estimated using linear wave theory (LWT) which is employed by MPC. The method suggested in this article is thought to increase the confidence in robotic observations in shallow ocean water and can even reduce damage done to robotic fish stations by water disturbances. MPC utilizes a sequence of input controls to minimize global cost function (deviation/squared distance from the target state). An optimized projected trajectory is modeled and then executed using thrusters which counteract the negative effects of the majority of water disturbances.

III. Project Requirements

A. The project requirements needed are as follows:

1) A control unit with microprocessors and peripherals: A "control unit" with microchips,"microprocessors, remote, peripherals, mini suction vacuum and a computer [5]. The Robotic fish's artificial intelligence of a cleaner fish would be inserted in it's head with a AI micro chip and a 32 bit microprocessor. A peripheral, a computer, and a mini cam are needed to on the user side to monitor what is happening in the water and to execute further codes if necessary. This would give us total control over the robotic fish's movement and how much garbage it needs to pick up in a maximum set of time of three to five minutes per location. The waste will be stored in the Robotic Fish's airbag stomach.

The control unit will also have a sector for the "communication unit", which includes wireless mic, sensors, and algorithms [5]. The wireless mic will tell the fish to wake up and to begin collecting garbage. The sensors would indicate the surrounding area of the robotic fish position.The Robotic Fish would have five sensors built within. It will contain a Proximity sensor, MYTHINGS sensor, Acceleration sensors, Gyroscope sensor, and an Acceleormeter sensor.The Proximity sensor will be used to detect if any waste objects are near for the Robotic fish to pick up. The MYTHINGS sensor is used to capture the data from the GPS tracking. The acceleration sensor will be used to indicate the Robotic fish's initial velocity and final velocity. The gyroscope sensor would determine the displacement of the Robotic Fish around the X-axis and Y-axis and its turning points. The acceleormeter sensor is used to detect the motion of the other fish near by and promotes a danger signal to the Robotic Fish.

The algorithm is significant in the speed and control of the Robotic's fish cleaning capabilities. The robotic fish's " support unit" is from its soft exoskeleton body (head and body) which is made of silicon [5]. There is also an "activation unit" that requires batteries, wiring, GPS, mini camera, gas tube chamber(breathing mechanism) , five rotating motors and the robotic fish's tail and fin [18]. The gas tube chamber is a breathing mechanism for the Robotic fish to stay underwater for a prolonged time, exactly 10 hours.

2) Robotic Fish Movement: In order for the robotic fish body to move from location to location to vacuum up the underwater waste, we would need a formula for the entire body movement. A little more than half of the body is being moved at an altitude so that the robotic fish can swim at both surface level and deeper into the water to collect any unknown objects in the water with its suction vacuum. Approximately of the body including the tail does most of the movements and the rest of motions for turning is based on the fins and the head in the The expectancy for the altitude is equal to or less than zero. A Robotic fish's motion can be divided into two separate parts. One part is the flexible silicone body and the second part is its two oscillating fins, and six joints, which is used for the robotic fish's high performance and self containment [5]:

$$ybody(x,t) = [(c1x + c2x2)][Sin(kx + wt)])$$

The formula above indicates direction of the robotic fish's movement. The interval ybody is equal to the diagonal displacement of the robotic fish as it is moving side to side mimicking the cleaner fish's movement. Also the diagonal displacement also includes the robotic fish going across the water to suck up the garbage with its vacuuming mouth. The x variable represents the displacement as the robotic fish moves vertically in the water to get to the other waste that might be hiding in the reefs or sea weed. The k variable shows the body wave amount (k=2/lambda). Lambda indicates the robotic fish's wavelength while swimming [6].

B. Speed Control Algorithm and Robotic Kinetics:

As the robotic fish swims in the water, its body speeds up by rotating the pattern of the motors in the tail and the fins, which produces oscillating frequency. The suction vacuum is also produced by the Robotic fish's motors as well. The momentum is built up as the fish interacts with the waters and changes its frequencies. The fish should be able to accelerate and also decelerate when cleaning the area underwater going from its initial velocity (Vi) to its final velocity (Vf). However, with the inertial force and hydrodynamic force, the robotic fish should be able to control its speed and slow down to make any sudden turns or stop if it senses danger or sees more garbage to clean up [5].

The potential energy in the robotic fishes movement needs to be manipulated in order to promote change. The maximum acceleration is represented by the symbols Am , this can be restricted by the inertia force. Therefore the robotic fish should swim only from its initial velocity (Vi) to its final velocity (Vf). While it decreases in velocity, so does its frequency (f). Then the robotic fish's speed would become zero and it will drifts as the water current moves its body. Since the speed becomes zero then that implicated the hydrodynamic control is achieved [19].

Table I: Technical Specifications for a Prototype Robot Fish

Specification Type	Specification Value
Size (L x W x H)	13" x 3" x 5"
Weight	1kg
Number of joints	6
Oscillating Frequency	4 Hz
Length of oscillating tail & fins	7 inches
Max speed	1 m/s
Minimum turning speed	0.45 m/s
Max torque	44.1 psi (pascals)
Drive Mode	Tail & fin motors
Control Mode	Remote

Algorithm formula [5]:

$$v(t) = \{0.5Vf\,(1 - cos(\tfrac{\pi t}{T}))\} \qquad 0<t\leq T$$

$$\{0.5Vf(\,1 - cos\,(\tfrac{\pi(t-Td-T}{T}\} \qquad T<t\leq T_d$$

$$\{0\} \qquad T_d<t\leq T_d + T$$

$$t>T_d + T$$

IV. Methodology

Coding is very significant when implementing the artificial intelligence of a real cleaner fish to be programmed and used to mimic in the robotic fish. It will enable the robotic fish to have a false instinct so that it can move around, turn, flip, swim slow, swim fast, and vacuum the trash that is consuming the Fish's habitat. Nearly all of the codes are from Python script on the Tensor Flow Platform, the Cifar 10 code to identify the waste, the displacement and acceleration formula for the Robotic fish motion and speed control. This will enable the capture of data on how fast the fish moves, its wavelength, and its control to stop.

The robotic fish will identify and cleanup trash that would consist of plastic, rubber, and algae. The cleaning will last up to a maximum of 9 hours and given at least one hour to make it back to the surface. The codes are executed from the computer and then transferred into the robotic fish using a chip and a microprocessor. Coding will also enable the robotic fish to not remotely bump into any objects when searching for trash and not be destroyed in the process of the experiment. Both code and algorithms would work together to provide stability in movement and control of cleaning the environment. Our project will be sought with two methodologies for code and they are experiments and observations. These methodologies would help determine which codes are resourceful for the robotic fish to have the proper artificial intelligence as a vacuum cleaner fish and allow for the correct replica of fish-like movement. Trials are significant in the creating algorithm and kinematics, fish design, and therefore, it will be carried out in testing the codes. There would be three trial runs for the code [20]. Each of the codes are written out in Github before the execution process begins. Also there will be an evaluation each time the robotic fish cleans the water and how long it takes. The performance of each movement would be compared in each of the trials to see which codes are the best to use to clean underwater at a maximum speed of 1 m/s and a maximum torque of 44.1 psi [20].

The code will be encrypted so that it cannot be intercepted and decrypted by any hackers. Another reason for encrypting the code is so that the code cannot be manipulated by anyone other than the teammates that are writing the codes. The codes would also initiate the instinct of having camouflage mode in case of any predators. The camouflage mode does not require the robotic fish to temporarily disappear or change colors like an Octopus. However, the camouflage mode would consist of hiding behind seaweed or the rocks in the water. From here, this would determine the survival skills of the robotic fish and that if it can coexist among the other fishes in the water.

V. Background

Several open source libraries were used for testing and validation. A background on some of the open source libraries are listed and summarized as follows:

A. *Tensorflow [21]*

TensorFlow enables anyone without machine learning experience to develop deep learning models with an API and python script. It allows the Robotic fish research and experimental model to become a virtual model to run the programmed codes. Since the robotic fish has to swim in water a natural flow control is needed to use for the experiment of testing the dynamic movement of the Robotic Fish. This method would easily and quickly collect the data on the three trial runs. As opposed to building the model from scratch and running into runtime error. With TensorFlow runtime errors can be fixed.

Since Cifar 10 is one of the datasets, the Robotic fish will analyze at least 1,000 of the most common underwater waste images and pick up about 20-40 of those waste items. The training model will operate on the given variables to execute the movement. By using TensorFlow it provides more control over the model and virtually displaying the possibilities of what can happen with the robotic fish being three meters underwater. It can provide the losses, impact, metrics, and optimizers.

B. Keras [22]

Keras has been found to adhere to best practices for decreasing cognitive load. It provides dependable basic APIs, it reduces the amount of user actions necessary for frequent use cases, and it offers comprehensible and actionable feedback for user error. One of the great things about the ease of use does not negatively affect the adaptability of Keras because it integrates well with basic level deep learning languages, it permits users to implement anything that may have been built in the base language. Keras has become very popular as it has over 250,000 individual users as of the middle of 2018. Keras has a powerful presence in both the industry and research community than all other deep learning frameworks except for Tensorflow. It allows users to constantly interact with its built in features. Apps such as; Netflix, Uber, Yelp, Instacart, Zocdoc, and Square use Keras in their frameworks, in addition to several others. It is particularly prevalent for startups that put deep learning as a core feature in their products because of how powerful the tool is. Keras has also been deemed desirable by deep learning researchers, which put it at the 2 spot of preferred dependencies mentioned in scientific papers uploaded to arXiv.org.

C. Python Imaging Library (PIL) [23]

The Python Imaging Library (PIL) has been very useful in both image archives and batch processing applications. Users are able to use PIL to create print images, thumbnails and can be converted across different file formats. The histogram method offered by PIL gives the user the ability to obtain statistical data from an image and use it for automatic contrast enhancement as well as global statistical analysis. The database includes basic image processing capabilities, point operations color space conversions and filtering with a set of built-in convolution kernels. It can also support image resizing, rotation and transformation without the loss of quality. The most up to date version include features such as; Tk PhotoImage and BitmapImage interfaces. It has a Windows DIB interface that is compatible with Python WIN in addition to other Windows-based toolkits. A large amount of GUI toolkits usually has some type of support by PIL.

D. Open Source Computer Vision Library (OpenCV) [24]

Open Source Computer Vision Library, or OpenCV, is an open source project developed by Intel. OpenCV contains a library of programming functions and methods used to train computers to automate tasks that the human visual system typically performs. These libraries can also be used to develop real-time computer vision applications. With the use of digital images or videos and programming methods that acquire, process, and analyze such digital content. OpenCV is written in C/C++/Python and can be implemented on multiple computing platforms. Applications such as facial recognition, motion tracking, mobile robotics, human computer interaction, and augmented reality all utilize OpenCV technology.

E. MatPlotLib [25]

Matplotlib offers an object-oriented API containing a plotting library for Python and NumPy. Using GUI toolkits such as GTK+ and Qt, Matplotlib users are able to easily embed their plots into their applications. Users can represent data in their applications by creating static, interactive, and animated plot visualizations using this open source API. These visualizations can be customized using various line styles and font properties, and they can also be updated and modified in real-time by both the application developer and end-user. In addition to the original Matplotlib library, third party/external packages can be also be used in conjunction with the original API provided through GitHub to further enhance the developers control over the visualizations contained in the application. The Matplotlib library is written in Python and can be used across various operating systems.

VI. Preliminary Findings and Research

The team explored various options on how to implement an AI enabled robotic fish. Of these options, garbage collection was chosen as the most viable option.

A. Source Code Repository Findings

The team explored and researched various source code repositories from various sources (i.e. GitHub, GitLab, Azure Repos, etc.), and found that this one was the most viable option: The Waste-classification repo created by techShash [26] provides open-source code that is able to distinguish waste as organic or recyclable which can be useful in many applications such as maintaining clean water sources and improved waste testing by the government and/or private companies (classifying "water junk" as true waste or something that can be recycled and reused in society). The Deep Learning Keras model was employed, containing six Convolutional Layers and two Dense Layers which attributed to 91% model accuracy. Convolutional Layers methodology effectively take subsets of input data and spreads it across the entire spectrum of input volume to compute products between the filter entry points and the actual input, whereas Dense Layers do not use filters and is a linear operation giving specific weights to each of the inputs and outputs.

B. Dependencies and Libraries Used

Since the team has chosen coding to be the main method in which we program the robotic fish, some dependencies were needed in order to make the code run properly. The dependencies chosen include; Keras [22], TensorFlow [27], OpenCV [24], PIL [23] and matplotlib [25]. Keras is an API (application programming interface) that is easy to learn and reduces the amount of user actions that are necessary for common use cases. It also integrates with other learning languages like TensorFlow and enables the user to implement anything that can be built in the base language. TensorFlow is an end-to-end platform that allows users to build and deploy machine learning models. TF offers many levels of abstraction to ensure the right one fits the needs of the coder. The function ‘eager execution’, allows for easy and quick, prototyping and debugging. OpenCV is an open source computer vision library used for machine learning. It contains more than 2,500 optimized algorithms that include, but are not limited to, a wide-ranging set of standard and state of the art computer vision and machine learning algorithms. Python Imaging Library, also known as PIL, adds image processing to out Python interpreters. This library offers a variety of file formatting’s support, and it provides a stable foundation for an overall image processing tool. Matplotlib is a library for creating 2D plots of arrays in Python. Its intended use was to simply create plots with as little as one command to a few. It can x also, create a histogram of the data that is being used.

C. Evaluated Dataset

Training a machine learning model with the right dataset play a significant role. The dataset decides how efficiently the model is trained and then how accurately it gives results. Dataset that was used to train the model contained a total of 70 images out of which 35 images were organic waste items and 35 images were recyclable waste items. For the purpose of testing 15 organic waste images and 15 images of recyclable waste were used making the total images that were used for testing 30. Approximately 70% of the data was used for training and 30% was used for testing.

D. Training Model

In order for the Robotic fish to classify underwater if it is suction vacuuming garbage or not is based on the back end of TensorFlow. This will authorize the machine learning capabilities for the Robotic fish. The robotic fish will be programmed with code from the TensorFlow platform to pick up “foreign “ objects in the water. This means that objects that are not from the underwater habit. These foreign objects would be classified as plastic straws, plastic bags, sunglasses, and paper which are easy for the vacuum to suck up without getting clogged.

CNN (Convolutional Neural Network) is the training model or prediction model used to replicate the results from preliminary findings in distinguishing if the "foreign" objects are garbage or not. The training model is also constructed from the TensorFlow platform as well. The waste would be sorted into different categories in the TensorFlow library for the robotic fish to discover, predict, classified and identify trash with images. The images of the trash are programmed into the artificial intelligence to help the robotic fish identify the waste it has to clean up. CNN will use CIFAR 10 code as a data set to contain at least “60,000 “images of garbage and underwater trash. These images would be placed into “10 categories with an amount of “ 6,000” images in each category [27], as shown in the code snippet below:

```

from __future__ import absolute_import, division,
    print_function, unicode_literals
import tensorflow as tf
from tensorflow.keras import datasets, layers,
    models
import matplotlib.pyplot as plt
# CIFAR 10 Code
(train_images, train_labels), (test_images,
    test_labels) = datasets.cifar10.load_data()
# Normalize pixel values to be between 0 and 1
train_images, test_images = train_images / 255.0,
    test_images / 255.0
```

VII. Final Results

By utilizing a similar model as the one developed by GitHub user techSash [26], the team was able to successfully train and test an image classification model with high accuracy.

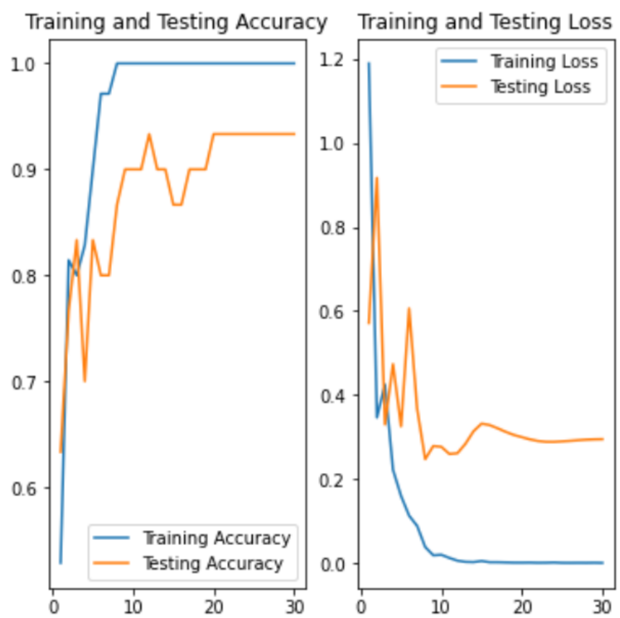

Figure 3: Final Results of the image classification model the team developed

The team chose a sample quantity of 100 images; 70 images used to train the model, and 30 images used to test it. With these parameters set, an accuracy of 93% was achieved by the 20th Epoch (the 20th pass through the full set in the neural network).

```

"Epoch 1/30\n",
"3/3 - 38s - loss: 1.1894 - accuracy: 0.5286 -
    val_loss: 0.5715 - val_accuracy: 0.6333\n",
"Epoch 2/30\n",
"3/3 - 0s - loss: 0.3464 - accuracy: 0.8143 -
    val_loss: 0.9170 - val_accuracy: 0.7667\n",
"Epoch 3/30\n",
"3/3 - 0s - loss: 0.4249 - accuracy: 0.8000 -
    val_loss: 0.3298 - val_accuracy: 0.8333\n",
"Epoch 4/30\n",
"3/3 - 0s - loss: 0.2219 - accuracy: 0.8286 -
    val_loss: 0.4737 - val_accuracy: 0.7000\n",
"Epoch 5/30\n",
"3/3 - 0s - loss: 0.1597 - accuracy: 0.9000 -
    val_loss: 0.3254 - val_accuracy: 0.8333\n",
"Epoch 6/30\n",
"3/3 - 0s - loss: 0.1128 - accuracy: 0.9714 -
    val_loss: 0.6063 - val_accuracy: 0.8000\n",
"Epoch 7/30\n",
```

```
"3/3 - 0s - loss: 0.0883 - accuracy: 0.9714 -
    val_loss: 0.3668 - val_accuracy: 0.8000\n",
"Epoch 8/30\n",
"3/3 - 0s - loss: 0.0381 - accuracy: 1.0000 -
    val_loss: 0.2473 - val_accuracy: 0.8667\n",
"Epoch 9/30\n",
"3/3 - 0s - loss: 0.0184 - accuracy: 1.0000 -
    val_loss: 0.2787 - val_accuracy: 0.9000\n",
"Epoch 10/30\n",
"3/3 - 0s - loss: 0.0197 - accuracy: 1.0000 -
    val_loss: 0.2768 - val_accuracy: 0.9000\n",
"Epoch 11/30\n",
"3/3 - 0s - loss: 0.0121 - accuracy: 1.0000 -
    val_loss: 0.2597 - val_accuracy: 0.9000\n",
"Epoch 12/30\n",
"3/3 - 0s - loss: 0.0057 - accuracy: 1.0000 -
    val_loss: 0.2614 - val_accuracy: 0.9333\n",
"Epoch 13/30\n",
"3/3 - 0s - loss: 0.0028 - accuracy: 1.0000 -
    val_loss: 0.2840 - val_accuracy: 0.9000\n",
"Epoch 14/30\n",
"3/3 - 0s - loss: 0.0022 - accuracy: 1.0000 -
    val_loss: 0.3138 - val_accuracy: 0.9000\n",
"Epoch 15/30\n",
"3/3 - 0s - loss: 0.0048 - accuracy: 1.0000 -
    val_loss: 0.3319 - val_accuracy: 0.8667\n",
"Epoch 16/30\n",
"3/3 - 0s - loss: 0.0019 - accuracy: 1.0000 -
    val_loss: 0.3285 - val_accuracy: 0.8667\n",
"Epoch 17/30\n",
"3/3 - 0s - loss: 0.0019 - accuracy: 1.0000 -
    val_loss: 0.3207 - val_accuracy: 0.9000\n",
"Epoch 18/30\n",
"3/3 - 0s - loss: 0.0014 - accuracy: 1.0000 -
    val_loss: 0.3123 - val_accuracy: 0.9000\n",
"Epoch 19/30\n",
"3/3 - 0s - loss: 0.0010 - accuracy: 1.0000 -
    val_loss: 0.3050 - val_accuracy: 0.9000\n",
"Epoch 20/30\n",
"3/3 - 0s - loss: 9.7703e-04 - accuracy: 1.0000 -
    val_loss: 0.2997 - val_accuracy: 0.9333\n"
```

VIII. Conclusion

Climate change and environmental waste are one of the world's biggest problems today. With the collective plastic in oceans across the world totaling 4.8 - 12.7 million tonnes [28], utilizing the technology available to us is vital to improving the way we go about solving this problem. Throughout the paper, the team has shown that there are ways to implement current technologies in artificial intelligence and robotics to take on the challenge of environmental waste. The team utilized various open source libraries in software development to design, train, and test a deep learning image classification algorithm to identify potential plastic waste, and distinguish it from other objects in an ocean environment (i.e. seaweed or other fish). In applying these tools and algorithms, the team was able to accomplish an accuracy of 93% in correct image classification when testing against a data set of randomly images of plastic waste in water.

By building a robot fish with the capability to pick up and clean waste in oceans around the world, the team hopes to contribute to the ongoing effort to keep our oceans and environment clean.

References

[1] Fatima Jaber, Abrar Abdulsalam, Jawaher Jafar, Batool Moosa, Amna Khalil, Salama Bilal, and Teye Brown, "Autonomous robot fish for underwater pollution detection," in *2018 Advances in Science and Engineering Technology International Conferences (ASET)*. IEEE, pp. 1–6.

[2] Matthew Dunbabin and Lino Marques, "Robots for environmental monitoring: Significant advancements and applications," *IEEE Robotics & Automation Magazine*, vol. 19, no. 1, pp. 24–39, 2012.

[3] Aisling Irwin, "How to solve a problem like plastics: Everybody is talking about plastic waste. but we should guard against quick – and potentially flawed – fixes, warns aisling irwin.," *New Scientist*, vol. 238, no. 3178, pp. 25 – 31, 2018.

[4] Robert K Katzschmann, Joseph DelPreto, Robert MacCurdy, and Daniela Rus, "Exploration of underwater life with an acoustically controlled soft robotic fish," *Science Robotics*, vol. 3, no. 16, pp. eaar3449, 2018.

[5] Junzhi Yu, Min Tan, Shuo Wang, and Erkui Chen, "Development of a biomimetic robotic fish and its control algorithm," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 34, no. 4, pp. 1798–1810, 2004.

[6] Jindong Liu, Ian Dukes, and Huosheng Hu, "Novel mechatronics design for a robotic fish," in *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2005, pp. 807–812.

[7] Xiaobo Tan, Drew Kim, Nathan Usher, Dan Laboy, Joel Jackson, Azra Kapetanovic, Jason Rapai, Beniamin Sabadus, and Xin Zhou, "An autonomous robotic fish for mobile sensing," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2006, pp. 5424–5429.

[8] Jack Connor, Jordan Nowell, Benjamin Champion, and Matthew Joordens, "Analysis of robotic fish using swarming rules with limited sensory input," in *SoSE 2019: Internet of Things as system of systems: Proceedings of the 2019 14th Annual Conference System of Systems Engineering*. Institute of Electrical and Electronics Engineers, 2019, pp. 69–74.

[9] Swadhin Sambit Das, Dayal R Parhi, and Suranjan Mohanty, "Insight of a six layered neural network along with other ai techniques for path planning strategy of a robot," *Emerging Trends in Engineering, Science and Manufacturing (ETESM-2018), IGIT Sarang, India*, 2018.

[10] Long Wang, Shuo Wang, Zhiqiang Cao, Min Tan, Chao Zhou, Haiquan Sang, and Zhizhong Shen, "Motion control of a robot fish based on cpg," in *2005 IEEE International Conference on Industrial Technology*. IEEE, 2005, pp. 1263–1268.

[11] J Edward Colgate and Kevin M Lynch, "Mechanics and control of swimming: A review," *IEEE journal of oceanic engineering*, vol. 29, no. 3, pp. 660–673, 2004.

[12] Vishwamithra Sunkara, Animesh Chakravarthy, Xiongfeng Yi, Wenyu Zuo, and Zheng Chen, "Cooperative optimal collision avoidance laws for a hybrid-tailed robotic fish," *IEEE Transactions on Control Systems Technology*, 2019.

[13] Animesh Chakravarthy and Debasish Ghose, "Obstacle avoidance in a dynamic environment: A collision cone approach," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 28, no. 5, pp. 562–574, 1998.

[14] Animesh Chakravarthy and Debasish Ghose, "Collision cones for quadric surfaces," *IEEE Transactions on Robotics*, vol. 27, no. 6, pp. 1159–1166, 2011.

[15] Animesh Chakravarthy and Debasish Ghose, "Generalization of the collision cone approach for motion safety in 3-d environments," *Autonomous Robots*, vol. 32, no. 3, pp. 243–266, 2012.

[16] Vishwamithra Reddy Sunkara and Animesh Chakravarthy, "Cooperative collision avoidance and formation control for objects with heterogeneous shapes," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 10128–10135, 2017.

[17] Daniel C Fernandez and Geoffrey A Hollinger, "Model predictive control for underwater robots in ocean waves," *IEEE Robotics and Automation letters*, vol. 2, no. 1, pp. 88–95, 2016.

[18] "Autonomous, self-contained soft robotic fish at mit," Mar 2014.

[19] Henrik M Schmidt-Didlaukies, Asgeir J Sørensen, and Kristin Y Pettersen, "Modeling of articulated underwater robots for simulation and control," in *2018 IEEE/OES Autonomous Underwater Vehicle Workshop (AUV)*. IEEE, 2018, pp. 1–7.

[20] Junzhi Yu, Kai Wang, Min Tan, and Jianwei Zhang, "Design and control of an embedded vision guided robotic fish with multiple control surfaces," *The Scientific World Journal*, vol. 2014, 2014.

[21] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng, "Tensorflow: Large-scale machine learning on heterogeneous systems," [Online], https://www.tensorflow.org/, 2015.

[22] Keras, "Why use keras?," [Online], https://keras.io/why-use-keras/.

[23] Pillow, "Pillow overview," [Online], https://pillow.readthedocs.io/en/stable/handbook/overview.html.

[24] OpenCV, "About opencv," *OpenCV*, [Online].

[25] MatPlotLib, "Matplotlib history," [Online], https://matplotlib.org/3.1.1/users/history.html, 2019.

[26] TechSash, "Techsash/waste-classification," [Online], https://github.com/techSash/Waste-classification/blob/master/README.md, Jan 2019.

[27] "Why tensorflow," [Online], https://www.tensorflow.org/about.

[28] Jenna R Jambeck, Roland Geyer, Chris Wilcox, Theodore R Siegler, Miriam Perryman, Anthony Andrady, Ramani Narayan, and Kara Lavender Law, "Plastic waste inputs from land into the ocean," *Science*, vol. 347, no. 6223, pp. 768–771, 2015.