

Proposal of
Aquatic Identification and Sorting System

A
project
submitted by
John Schulz and Reiner Lintag

For
Senior Capstone Project



Department of Electrical Engineering

Peoria, IL

12/5/18

Table of Contents

| | |
|-----------------------|-----------|
| Abstract | 3 |
| Commercial Uses | 3 |
| Design | 3 |
| General | 3 |
| Power System | 3 |
| Simulation Tank | 4 |
| Imaging | 5 |
| Database | 6 |
| Neural Network | 7 |
| Results | 8 |
| Database | 8 |
| Neural Network | 9 |
| Conclusion | 10 |
| Appendix I | 11 |
| Appendix II | 11 |
| References | 12 |

Abstract

The proposal of this senior capstone project looks to provide a much-needed solution to monitoring aquatic life. The project will involve three main areas of development: specie identification, mobility, and power management. The identification side of this project will involve machine learning algorithms and image processing to calculate data such as type of species and approximate dimensional measurements. Whereas the mobility side of the project involves creating a power system to support the various digital systems in a remote environment for an extended period of time. Lastly, the management portion of the project is to implement a proactive approach to harnessing the data collected from the identification portion of the project to be able to sort the system's target object.

Commercial Uses

Two major commercial uses exist for such a product: management of natural aquatic populations and aquaculture, which was rated to be a \$128 billion-dollar industry in 2012 [1]. The targeted use case for the first stated commercial segment would allow for the sorting and removal of invasive species. A problem so large and expensive that in the case of keeping Asian Carp out of the Great Lakes it is estimated to cost from \$15 to \$18 billion and take 25 years to complete using traditional man-made barriers, as estimated by the Army Corps of Engineers [2]. The second commercial use is to allow the on and offshore aquaculture industries to systematically farm fish based on an individual size rather than on the collective batch size, thus helping to reduce inefficient harvesting that amounts to a \$50 billion dollar a year loss globally, as stated in a 2013 report by The World Bank.

General Design

The prospective system will be designed in a modular fashion to accommodate for dimensional scaling depending on the targeted environment, needs of the operator, and the size of the targeted specimens. The end design of this project is to contain no openly moving parts because safety for the target object and human involvement must be held to the highest regard. Ease of deployment is also key to the success of the device, it must be able to survive and function in a remote environment where access to the power grid is not always viable. The system must be built around the prospect of zero noise and vibrations as not to disturb the environment that it is operating in.

Power System

The power system that is going to be designed for this project will focus on mobility and management of the attached load. To power the system the user will need a power source that can be easily transported. The battery should last at least twelve hours or more during its deployment. Power to the system will go

to the development board and its corresponding sensors and a servo that will be outside of the sealed container filled with water and a fishing lure. With that being said, a method to recharge the power storage unit will be created. The system will utilize a camera and a ZYNQ 7000 series FPGA for image processing and neural network operation. The 23-watt pump system and LED lighting strip will be powered from their included power supplies since they are added parts used for simulation purposes only to create the fish database made of fake lures.

The success of the system depends on answering six key points:

1. What is it?
2. What size is it?
3. Will a successful sort be possible?
4. Execute the mechanical sorting hardware.
5. Will the device be scalable and moveable?

Simulation Container

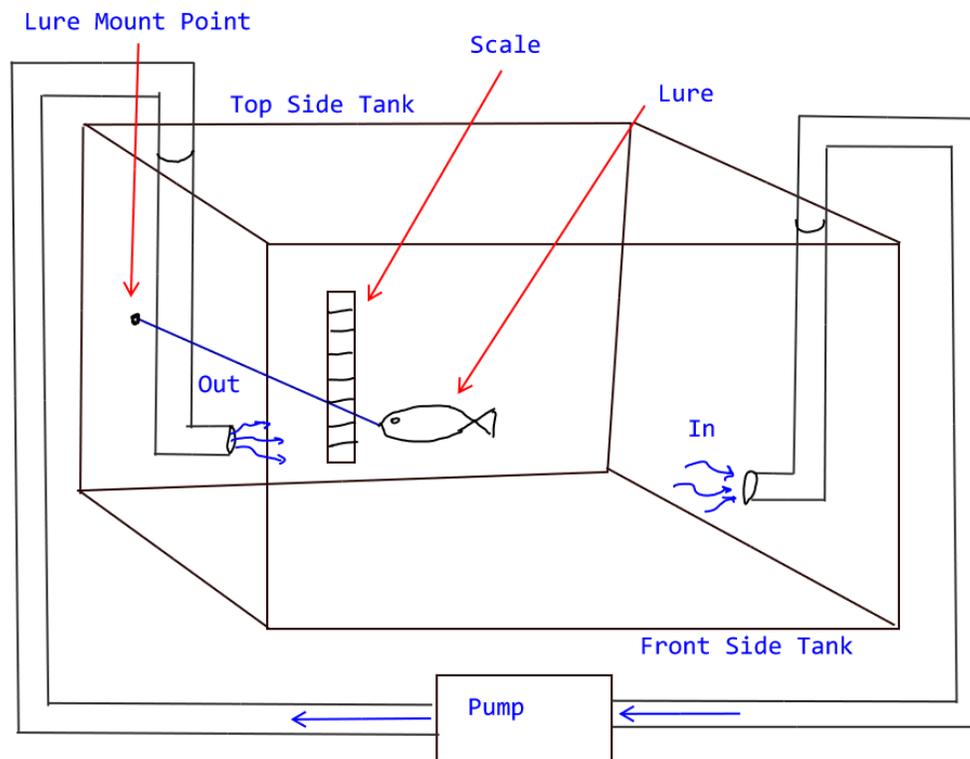


Figure 1 – Simulation Design

Utilizing a large 18"x26"x15" clear box made from food grade plastic to resist staining, this sealable container will house the environment used to create the database of lures that resemble real fish. As shown in figure 1, the only thing entering the container is the input and output hoses from the external pump and possible internal lighting. The pump system is used to make the fake lure swim as water passes over it and to circulate fine sediment debris that will be added to the tank in stages of measured amounts. The imaging processes will entail keeping the same distance from lens to subject, but the subject will be captured at a total of three different angles to improve neural network training. The distance from lens to subject will be based off of the minimal focus distance of the camera lens. Once the desired quantity of images has been shot for all 20 lures, then fine sediment debris will be added to reduce water visibility, thus the image capturing process will repeat.

A layering of black heavy-duty trash bags will be cut and applied to half of the surface area of the container to reduce undesirable affects from external lighting, reflective glare on the subject and container, and to limit background details that might confuse the neural network during testing. The rest of the surface will be left uncovered for the subject to be imaged. The use of a controlled and constant background resembles the type of constant background that would be used in a fish sorting machine. The lures will be attached by clear fishing line to the top of the tank on the end where the water exists the hose, as shown in figure 1. This will allow the lure to freely swim in the stream of water as would occur when reeling a lure in from a lake.

Imaging

The imaging system will be a Canon 70D DSLR camera using a 35-50 mm zoom lens provided by John Schulz. The subjects will be shot at a resolution of standard 720p at 60 frames per second to help reduce blur of the lures swimming. An internal LED strip mounted to the lid of the container will be used to light the subject from above. This allows for imaging to be done in a dark room to eliminate reflections from the camera and external environment. The focal length of the lens will remain fixed at a value that works best and will not be changed to reduce irregularity in our lure database. Images will be stored on a solid-state drive, with a folder existing for each lure to separate the image classes for neural network training purposes. Once the network is trained and successfully on the FPGA, images from the Canon 70D will be fed directly into the development board via an HDMI connection and processed locally on the board. After being processed, the data will be sent through the neural network for it to classify which lure is present.

Database

The database will consist of 20 different classes of lures that the neural network will train on. The process of receiving the data from the camera and producing a trainable dataset can be seen below in figure 2.



Figure 2 – Creation of Database from Camera to Training.

Ideally, the database should consist of at least 10,000 images per class. Thus, if every 6th image is used from the video recorded at 60 frames per second, then each class must be filmed for at least 14 minutes each. Additionally, the subjects will be shot at different water clarities and angles to increase the networks recognition accuracy. Then the original images will be duplicated by MATLAB code and applied Gaussian white noise and randomly selected for a left-right and/or up-down flipping operation.

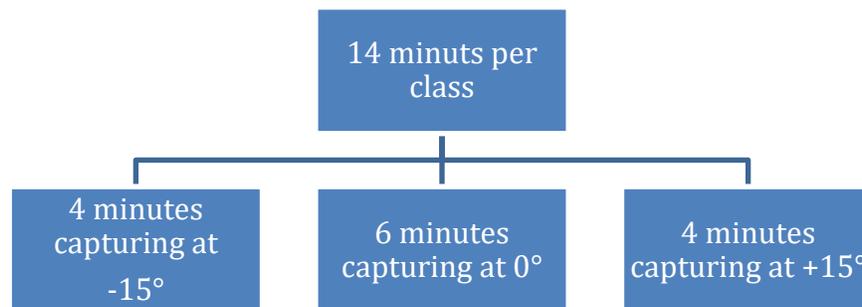


Figure 3 – Capturing Time Schedule.

The capturing time schedule in figure 3 provides details on the amount of time each class should be filmed per level of debris in the water. Consequently, after all 20 classes have been filmed, the process repeats itself for the next amount of debris that will be added. Ultimately the guidelines for time in figure 3 should produce a consistent database to train from. The estimated size of the database with a duplicated set of images with noise and image rotation added combined with a total of four debris levels would result in a database of approximately 80,000 images per class and a total database size of 1.61 million images.

Machine Learning and Network Structure

Training of the neural network will be done on the lure database using a Nvidia GTX 1060 graphics card. The Matlab Machine Learning Toolbox will be utilized to create and train the network on the graphics card. The resulting network structure and weights will be exported in the machine learning standard *.onnx* compressed package file type. From there, the *.onnx* package will be imported into MATLAB where software is currently being developed by the team that will analyze the neural networks structure and generate the VHDL code for the network to run on the FPGA. Images will be processed on the FPGA and passed into the network where the results from the network will be hot-key encoded and transmitted over UART to a computer to be stored.

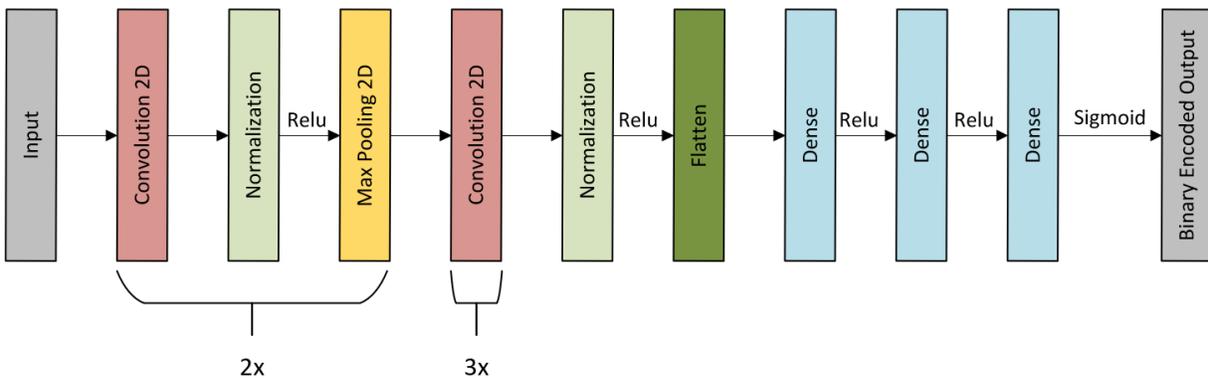


Figure 4 – Convolutional and Feedforward Network [4][6].

The structure of the network used in this project will be similar to AlexNet, which is a well-known machine learning classifying network that famously won the 2012 ImageNet LSVRC-2012 competition [5]. The structure of AlexNet best suits our use case because it is a small enough neural network that should be able to fit onto a mid-level performing FPGA. The structure of AlexNet shown in figure 4, showcases five convolutional layers in red, two normalization layers in light green, two max pooling layers in yellow, one flattening layer in dark green, three dense layers, and one binary encoded output in gray at the end. Once on the FPGA, the neural network will utilize a signed-magnitude 12-bit integer bus. Bus size is important because the smaller the bus, the less area that the network will require on the FPGA.

Results

The current size of the database contains five and half hours of recorded footage consisting of all 20 image classes under clear conditions. So far only the footage for the first four lure classes have been split and processed in the lure database used for network training. Thus, the current size of the processed database is approximately 103k images with a resolution of 200 by 150 RGB pixels. At minimum each class consists of 16,000 training images with the rest separated for testing and validation data.

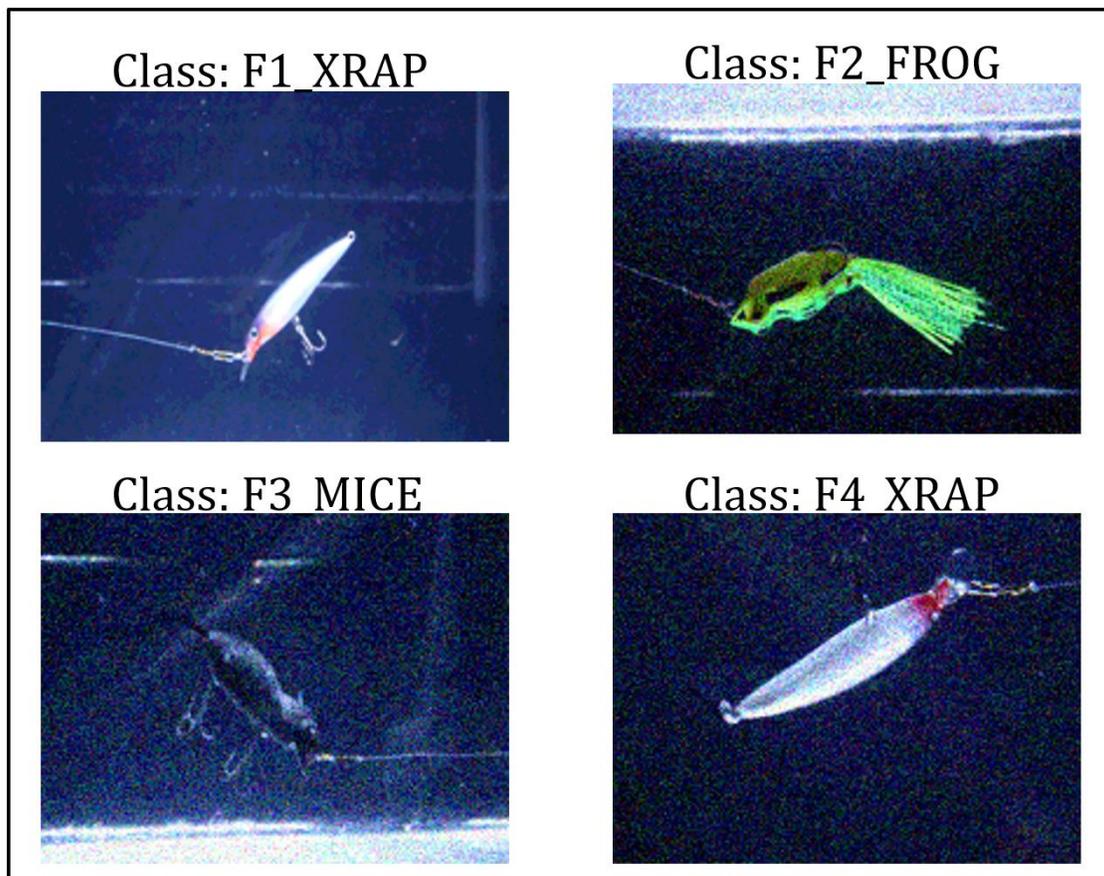


Figure 5 - Processed Training Data Samples

The above figure contains a picture from each of the four classes used to train the initial neural network to evaluate its accuracy and performance. Classes F1 and F4 were chosen as test classes because they have the same size and body coloring, but slightly different face coloring. F2 was chosen as a strong outlier that should be an easily recognizable lure class for the neural network, while F3 was selected to test our image intensity scaling and whether the neural network could distinguish a black lure from an all-black background. All three image processing operations used to obscure the dataset of duplicated images can be seen in classes F2 through F4, which is adding a combination of noise and a left-right or up-down flip.

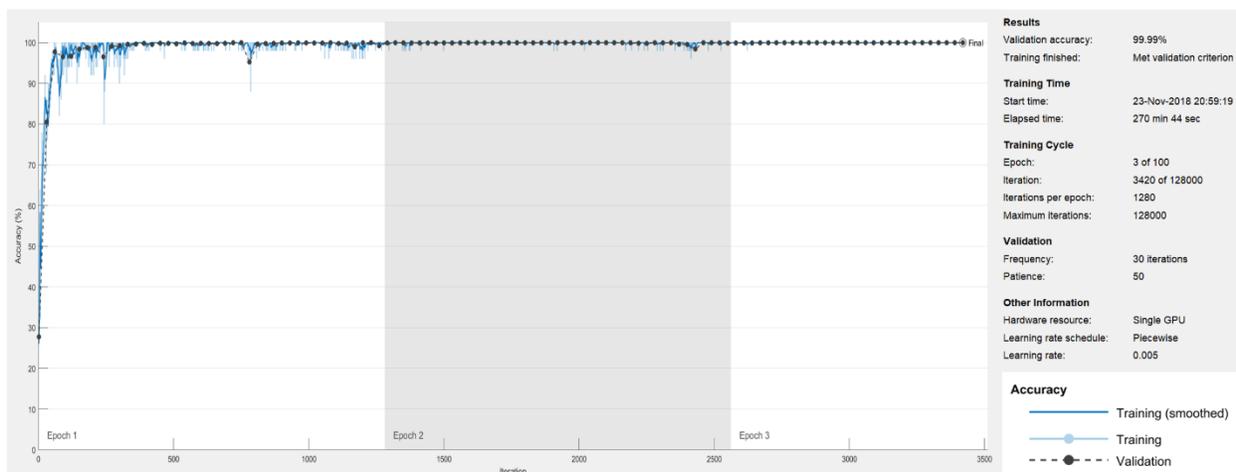


Figure 6 – Accuracy of 4 Class Network

Figure 6 contains the training results from the network design shown in figure 4. The goal of the 4-class network was to analyze the design of the neural network model against the data to make sure that there should be no issue when expanding the model to more classes. The data in the accuracy plot above shows that the network can recognize its classes with 99.99% accuracy. The network went through 3,420 iterations consisting of batches with 100 images loaded to the GPU at one time. Every 30 iterations the network was setup to test itself against validation data that it will not perform back propagation on. Checking validation data is the most important metric in the accuracy plot, shown by the black dotted line, because it provides a real-world look into how the machine learning model would perform on untrained images. The reason the plot stops after the third epoch is because the neural network hit its patience limit of 50. A patience limit is a design attribute setting that causes a training model to stop learning once the accuracy stops increasing after a set number of iterations, which is 50 in the case of the 4 Class network.

Discussion

To be clear, a problem such as the Asian Carp in the Mississippi River Interbasin, in our opinion has been approached wrong, it is not a physical barrier or an elimination of a species problem, it is a population control issue. Secondly, altering the DNA to control a species is not yet well understood and incredibly dangerous considering, the DNA of Asian Carp has already been found in the DNA of local fish populations in the Great Lakes, which implies that Asian Carp are already in the Great Lakes [3]. Therefore, we believe a population control approach utilizing machine learning and image processing is the only plausible solution to this problem; one that has not been explored or created yet.

Conclusion

Designing, building, and accomplishing the proposed project will have a wide impact on the success and survival of native species and will be paramount in its effect on the aquaculture industry. Bradley University will be on the forefront of solving a local and global problem facing the long-term sustainability of aquatic populations.

Appendix I - Milestones

- 8/30/2018 – Senior capstone officially approved by faculty
- 9/12/2018 – First meeting with adviser, Dr. Imtiaz
- 9/23/2018 – Finalize design and hardware components
- 10/5/2018 – Start testing image intensity levels for neural network
- 10/11/2018 – Submit order form for parts
- 10/19/2018 – Received simulation tank parts
- 10/22/2018 – Finish image intensity testing
- 10/27/2018 – Begin building image databases for neural network.
- 10/29/2018 – Submit rough outline of neural network structure
- 11/09/2018 – Finalize stats of image database
- 12/4/2018 – Demonstrate neural network: accuracy & structure
- 12/4/2018 – Demonstrate system power usage
- Winter Break
- 3/20/2018 – Demonstrate a functioning neural network on an FPGA
- 4/10/2018 – Demonstrate the power delivery system

Appendix II – Parts and Materials

- Tank – Cambro 12”x18”x9” Clear Food Grade Storage Box
- Lid – Cambro Multipurpose 12”x18” Clear Lid
- Pump – Jebaco DCP-2500, Max: 23 [W]
- LED Light Bar – Mingdak Max: 4 [W]
- Lures – Rapala, Quantity: 20
- FPGA – Avnet Zedboard with a Xilinx ZYNQ 7000 chipset
- Camera – Cannon 70D DSLR with 21 M-Pixel Sensor
- Lens – 35-75 mm with STM (silent-stepper motor) and OS (optical stabilization)
- Background – Hefty Black Trash bags

References

- [1] World bank, “FISH TO 2030 Prospects for Fisheries and Aquaculture,” Rep. no. 83177-GLB, Dec. 2013. [Online]. Available: <http://www.fao.org/docrep/019/i3640e/i3640e.pdf>. Accessed on: April 23, 2018.
- [2] News Desk, “Keeping Asian carp out of the Great Lakes will cost billions and take decades,” *PBS NewserHour Productions LLC.*, Jan. 6, 2014. [Online]. Available: <https://www.pbs.org/newshour/science/keeping-asian-carp-out-of-the-great-lakes-will-cost-billions-and-take-decades>. Accessed on: April 23, 2018.
- [3] Vice News, “The Worst Fish in America: Asian Capocalypse,” Sep 24, 2014. [Online]. Available: <https://www.youtube.com/watch?v=YnZp1jtOhR0>. Accessed on: April 23, 2018.
- [4] S. Raje, “AI Acceleration”, *presented at Xilinx XDF Silicon Valley 2018 Conf.*, Xilinx Inc., Oct. 10, 2018, San Jose, CA. [Online]. Available: <https://www.youtube.com/watch?v=qWFREOeRiqo>. Accessed on: Oct. 25, 2018.
- [5] H. Gao, “A Walk-Through of AlexNet”, Medium Corp., Aug. 7, 2017. [Online]. Available: <https://medium.com/@smallfishbigsea/a-walk-through-of-alexnet-6cbd137a5637>. Accessed on: Oct. 25, 2018.
- [6] Y. Ayster, “Lecture: Deep Learning for Vision”, *MIT Computer Vision* (Course 6.869: Advances in Computer Vision). Cambridge, MA., USA. [Online]. Available: <http://6.869.csail.mit.edu/fa16/lecture/6.869-Lecture19-DeepLearning.pdf>. Accessed on: Oct. 29, 2018.