Viability of Indoor Robotic Air Aquariums

Lazaro Herrera, Francisco Peleato, Jong-Hoon Kim
Discovery Lab, School of Computing and Information Science
Florida International University, 11200 SW 8th St, Miami, Florida 33199
lherr012@fiu.edu, fpele001@fiu.edu, kimj@cis.fiu.edu

ABSTRACT

Zoos and aquariums are visited by 700 million people every year; they are popular tourist locations all around the world and generate billions of dollars of revenue yearly. More than nine million people in the US own a small household fish tank, which indicates a desire to bring some of the aquarium features into their homes.

We considered the viability of real-life indoor aquariums. We found that the deficiencies that lead to the problem are closely related to costs of operation (maintenance), spacing (for housing) and personnel (for upkeep). In this paper, we analyze the problem, provide our current solution to the issue, and discuss future uses of the solution developed in order to tackle this problem.

Our solution is to bring smart robotics and smart algorithms in the form of autonomous robotic fish aquariums into the every-day home in order to deliver awe- inspiring simulated aquariums. The solution tackles the cost of spacing (by moving the aquarium into the air), the personnel (by removing the humans and replacing the fish with robots) and the costs of operation (by requiring just air space and helium).

For this solution, we created environment-aware fish simulators that were capable of replicating basic fish functions (such as swimming) along with full usage of avoidance maneuvering, ultrasonic sensing systems, and Bluetooth wireless networking. A key part of our solution is attempting to improve the existing user interaction between the current generation of robotics and humans from our current rudimentary simple button control to more sophisticated algorithms such as Fast Fourier Transform for voice.

Keywords

Keywords: Autonomous Robot, Aerial Robot, Bluetooth

INTRODUCTION

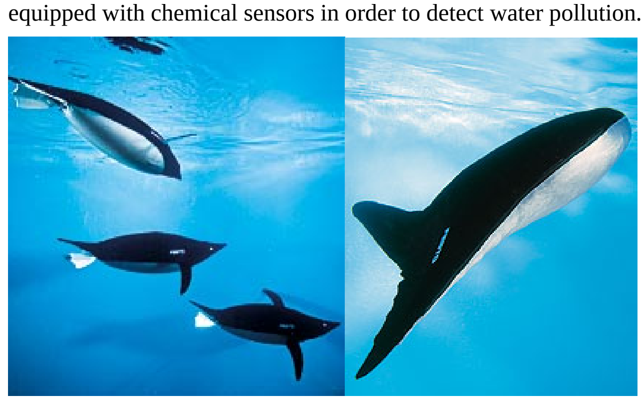
This paper focuses on the development of test platforms, algorithms and wireless Bluetooth control required for an indoor robotic air aquarium. Because of the inherent cost with water-based home aquarium systems that cannot rival the bigger aquariums, normal home aquariums are both smaller and less impressive. We approached the problem from a biological sense, using robotics to replicate the functions of fish. Implementation of basic swimming motions and obstacle avoidance were a key element to imitating the look and feel our robotic air aquarium. This paper gives an overview of how we accomplished our solution, challenges, test data and viability assessment.

BACKGROUND

As seen before, millions of people just in the US are willing to spend the money to bring the aquarium experience into their home. For the reasons of cost detailed previously, many companies and people around the world have resorted to robotics in order to make this a reality. At an average $20 cost per person, current aquariums around the world are capable of making 1.4 billion dollars every year just from the desire of people to see amazing sea creatures in front of their eyes. The average home setup runs between $100 and $1000+ for a small home aquarium system and does not fully utilize the space within the home to maximize the user enjoyment of their “aquarium”.

Robotics to simulate marine creatures is not a new concept. In the past, many companies and individuals have used robotics with marine-life-like form in order to simulate realistic marine behaviors. In the past, a German company, Festo and Essex University have used robotics to simulate sea-creatures with their AquaPenguin[3], AquaJelly[4] and AquaRay[5] prototypes, Essex Robotic Fish[12] and expanding it to air sea creatures as well.

The AquaPenguin[3] is a completely autonomous running on an AVR Mega 128 capable of moving through water at 5km/hr. The AquaRay[4] is a wirelessly-controlled robotic manta ray with a maximum speed of 1.8km/hr with a 40Mhz processing unit. The AquaJelly[5] is a light-controlled underwater jelly fish robot running on 2 ATMega168 chips at 8Mhz each with a ZigBit module for radio communication. It has a temperature and pressure sensors along with a 2-way infrared communication system. The Essex Robotic Fish[12] is a $42,000 robotic fish equipped with chemical sensors in order to detect water pollution.



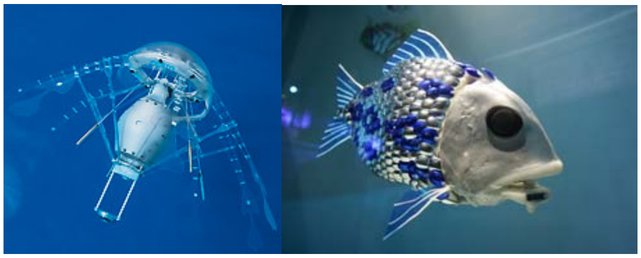


Figure 1. Festo Water Prototypes and Essex Robotic Fish

The AirJelly[1] is a remote-controlled jelly fish robot based on a 3V coreless motor. The AirPenguin[2] is a flying helium filled balloon powered by a 32-bit microcontroller, with a 3-axis compass/accelerometer and wireless 2.4Ghz Zigbee network developed by the German engineering company Festo. It includes other sensors such as a temperature sensor.

We will compare these aerial robots to ours in order to compare viability, cost and features.

Figure 2. Festo Air Prototypes.

STATE OF THE ART

Although the previous robots are great marvels of modern technology, we are going to compare them against our own system in order to show that our system is simply better than theirs in many aspects.

The AirPenguin[1] design is rather innovative and much like ours but is not as cost effective as our expected final design. It also does not implement microphone localization as our finalized product will.

The AirJelly[2] design is only remote controlled and has no autonomous capabilities unlike ours, making it highly unviable for the home entertainment sector.

Our design combines the best of both worlds, implementing an autonomous system along with a direct control system with Bluetooth and the Air Aquarium (AA) Serial communications system. Also, with the advent of the Roomba robotic vacuum cleaning system being used by millions of homes, it is no longer unusual to have robotics within the home.

APPROACH

Our approach to this project first attempted to replicate the basic fish capabilities in order to create a realistic model of the marine life we are attempting to replicate. On this idea, we modeled our robots and our algorithms to accomplish the task.

Robot

Our basic robot prototype is based on the Megafliers™ flying helium fish [9]. The fish is a helium balloon with a guide-rail running down the bottom that allows for altitude changes and a tail to provide propulsion, each one powered by a 1.5v DC motor. Changing the position of the ballast on the guide rail significantly alters the flight dynamics through changing the center of mass causing the fish to pitch up or down; moving the tail while doing this causes the fish to gain or lose altitude. We replaced the stock wireless RF radio control system with our own electronics system powered by an Arduino Nano running at 8MHz powered by a 3.7v 110mah portable battery. Our first stage involved using the microcontroller along with a TB6612FNG Dual Pololu Motor Driver Carrier in order to begin basic movement simulation [10]. The second stage was to interface our sonar sensors to the board as outlined below.

Figure 3. Tail propulsion system

Sensors

For this design, we used two LV-Maxsonar®-EZ1™ MB1010 sonar modules[11]. The modules have an effective range from 6 to 254 inches giving us total environment-awareness capability for our robot models. One of the two devices was attached on the built-in rail for forward range detection as to fulfill one of our primary simulation objectives (basic obstacle detection). The second of the devices was attached as to range the distance to the floor giving us effective height readout of our current position. Our approach was a classic logical approach for the forward sensor, we would monitor if the distance to an obstacle was smaller than our algorithm would allow and if it was, we would trigger an avoidance maneuver. The height sonar was more complex, instead opting for a range of about 70in of "acceptable height" so that the fish was at eye-height; if the fish drifted from that height, it would perform a "dive" or "climb" maneuver.

Figure 4. Assembled Sensor Suite on Guide Rail

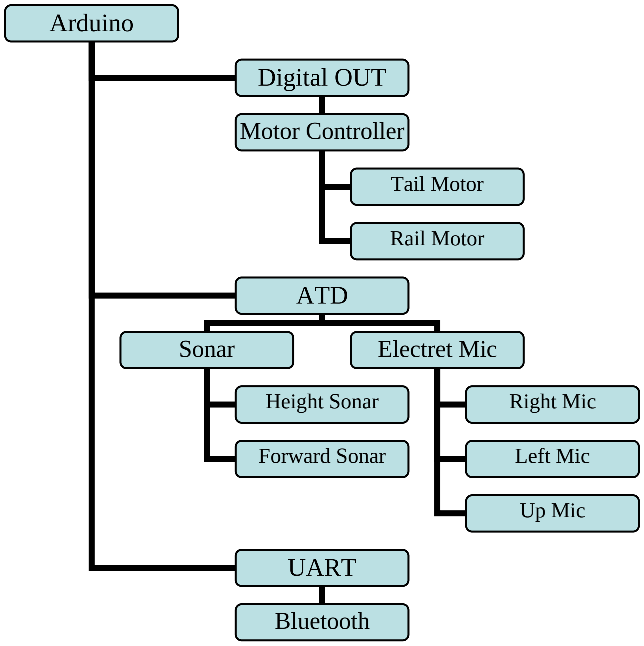


Figure 5. Architecture for final product.

Based on the previous architecture, we built the test platforms in order to test our solution's effectiveness. The main Arduino board uses Digital Out (to control the motor controller), Analog To Digital (for sonars and microphones) and UART (for Bluetooth functionality).

Communications

For the communication stage, we interfaced our Arduino board to a BlueSMIRF Gold Bluetooth device and created code to receive, process and execute serial commands using the FLK serial communication protocol we implemented. The Bluetooth standard is a proprietary open wireless technology for exchanging data over short distances (usually less than 30ft.)

The protocol is designed to be reusable and remappable for any Bluetooth project while having the capability of being extensible through the seventh and eight bit for data length. With that, one can simply add extra hardware such as a LED screen on the side for advertising, or sensors for a data gathering platform; modify the Arduino code to fit the new hardware and it should work with the existing protocol.

The protocol's specification is detailed below. As can be seen, communication begins with "{" and concludes with "}". All data is sent as characters and is delimited by "|" in between each parameter for different commands. A sample command such as requesting a readout of the forward sensor will be sent as follows "{0|1|0|00}" and would return "{2|1|1|2|50}" corresponding to a 50in. readout from the sonar.

	{	___	|	___	|	___	|	___	___	|	___	...	}
Byte →	0	1	2	3	4	5	6	7	8	9	10 thru n		n + 1

Byte #	**Description**	**Function**	**Value**	
	Message Delimiter	Separate Messages	|	
0	Beginning of message		{	
1	Operation type	Get	0	
		Set	1	
		Report	2	
3	Operation to perform	Height sensor	0	
		Forward sensor	1	
		Altitude up	2	0 for OFF / 1 for ON
		Altitude down	3	0 for OFF / 1 for ON
		Tail right	4	0 for OFF / 1 for ON
		Tail left	5	0 for OFF / 1 for ON
		Forward speed	6	0 for STOP / 1 for SLOW / 2 for MEDIUM / 3 for HIGH / 4 for FULL
		Auto	7	0 for OFF / 1 for ON
5	Unused		0	
7 & 8	Length of data		00 thru 99	
10 thru n	Data			-----------------

Figure 6. Architecture for final product.

For user control, we devised a Java-based Bluetooth control system. Java's capability for cross-platform work makes it capable of being run on any 32-bit Macintosh, Linux or Windows machine. Our main focus was to make easy to understand and allow the user to have control of every basic shark function that the built-in avoidance algorithm had access to. In the end, this was our final iteration of the control system.

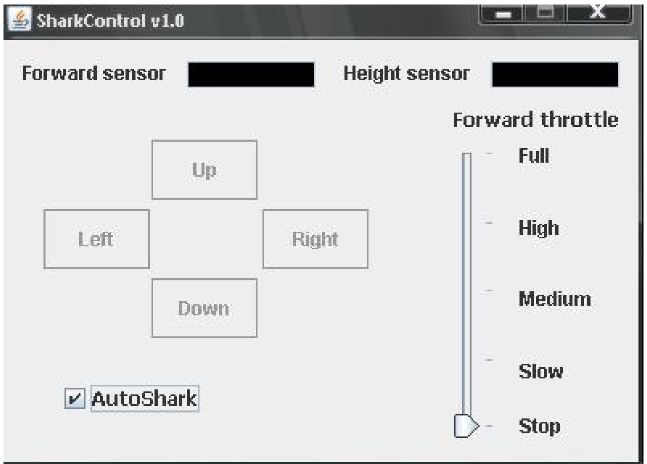


Figure 7. Shark Control GUI

One of the key features is the ability for a user to switch back to autonomous control on-the-fly for any reason as detailed on the following diagram. A simplified logic for our autonomous algorithm is laid out below.

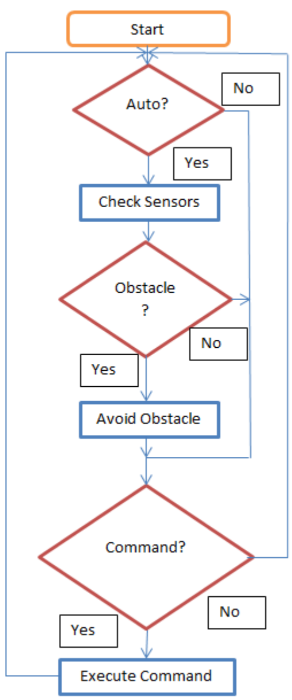


Figure 8. Programming Logic

PROTOTYPING AND EVALUATION

Our test results for the four fish that we created were within the preset test parameters (must avoid obstacles, must replicate fish functionality, and must be controllable). We created three obstacle-avoiding robots that would fly in "pseudo-formation" and avoided obstacles including each other. The fish would stay within 50in. of any obstacles and would fly between 120in. and 160in. off the ground. Our fourth fish, a "shark model" was equipped with a short-range camera and the Bluetooth control system; the system was tested and worked within acceptable limits. Some issues did exist during the building and testing of these robotic sea creatures and as such, they are detailed below. Figure 8 shows our 3 prototypes, bass, nemo, and shark; the bass has basic obstacle avoidance, the nemo has the microphone localization system and basic obstacle avoidance and the shark has the Bluetooth control system, a camera and basic obstacle avoidance.

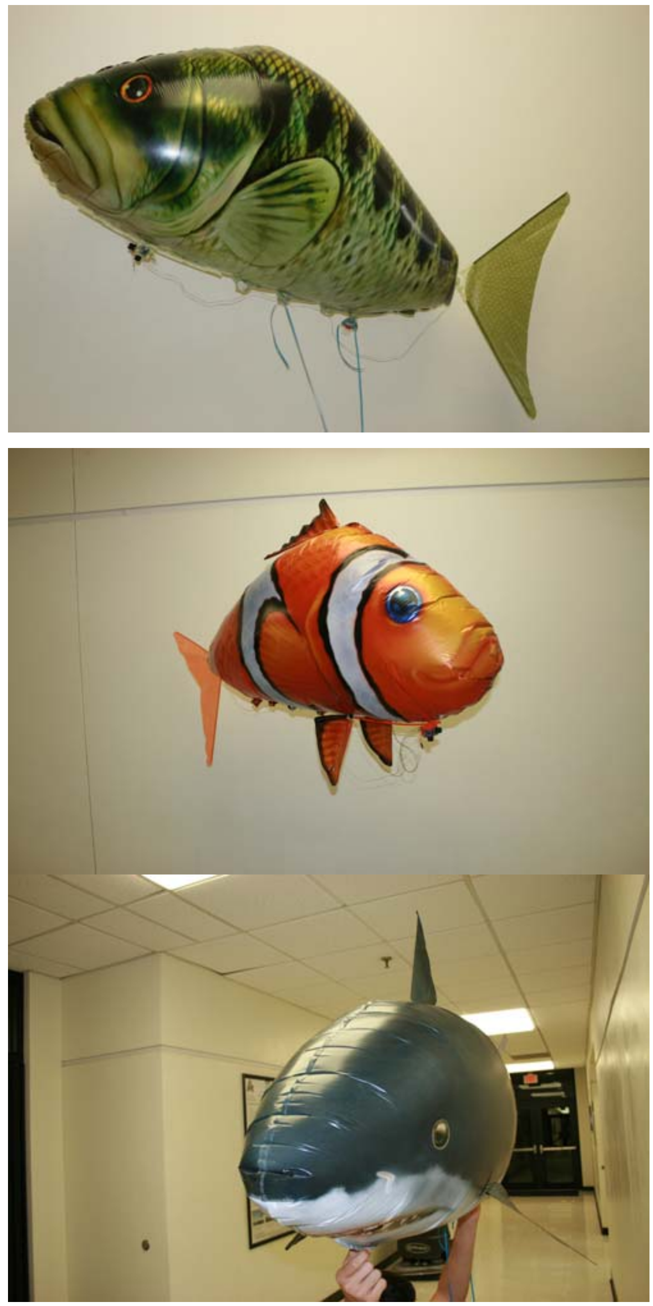


Figure 9. Our Prototypes

DISCUSSION

Solutions

During the preliminary integration of the sensors with the motor, we noticed that during the motor power phases all sensor data would be lost. We later on attributed this phenomenon to a critical voltage drop causing the Arduino to reset due to an inability to provide the necessary 2.5V to the sonars and 3.3V to the motor controller system. The problem was solved by separating the power supply for the Arduino and sensors, and the motor controller with one battery each. The extra weight added by the previous solution to the aforementioned motor controller issue caused the test platform to over the planned weight limit of 28 grams worth of electronics. Our solution to this was to puncture the plastic casing that acted as ballast with a drill until we reached our desired weight. The solution exceeded our expectations and allowed every one of our models to be lighter and capable of carrying more equipment for future test runs.

Issues

During the sensor integration stage, it became apparent that the sensors had an issue when the distance was less than 6in. When this situation presented itself, it would return a maximum 254in, unlike what the datasheet listed; causing some issues for our obstacle and height maneuvering logic [7]. Our second problem came from the Sparkfun Electret Microphones, which required a full Fast Fourier Transform pass in order to become usable for our project[8]. The Arduino's 8Mhz processor could not handle the amount of data even with the Arduino FFT library doing the work. As it stands, this is an unresolved issue. During the final stages, our last set of issues appeared. The first issue was simply a Bluetooth range issue when the device went beyond 25ft. and would instantly disconnect [6]. The disconnection would cause the Java GUI to lock up and become irresponsive. The second issue was with the analog wireless camera we installed on the shark; due to interference, the range was simply cut down to about 20ft. leading to severely minimized usable camera range (after 10ft., the picture would be too grainy to be usable for navigation). The solution to this issue will be to move to Wi-Fi Adhoc networking and having a digital camera use the same networking system.

FUTURE WORK

Our tests with this system are far from done. In the near future, we will implement the full "Follow me" functionality to allow users to interact with the aquarium via the electret microphone array detailed in the wiring diagram above. We also expect to expand into the advertising field by creating larger versions of our existing marine creatures and bringing them to life. We expect this to have real-life use as this sort of advertising medium has not been fully explored in the past or used by any commercial entity. We expect these to be done soon in order to be fully commercialized.

CONCLUSION

In this paper we presented our idea about a cost-effective air aquarium to rival the more expensive prototypes around the world.

We built our three prototype models, tested them for viability in the real world and evaluated the true usefulness and capabilities of our current prototypes, and contrasted them with other systems such as the AirPenguin and AirJelly.

We then located issues such as sonar issues, microphone issues, motor controller issues and others. We tackled issues and fixed as many as possible within our current system in order to improve our system for the future. The remaining issues have to be fixed in the future in order to achieve full commercialization.

REFERENCES

[1] "AirJelly." *Festo Corporation*. Festo Corporation, n.d. Web. 1 May 2012. < http://www.festo.com/rep/en_corp/assets/pdf/AirJelly_en.pdf >.

[2] "AirPenguin." *Festo Corporation*. Festo Corporation, n.d. Web. 1 May 2012. < http://www.festo.com/rep/en_corp/assets/pdf/airpenguin_en.pdf>.

[3] "AquaPenguin." *Festo Corporation*. Festo Corporation, n.d. Web. 1 May 2012.

< http://www.festo.com/cms/en_corp/9777.htm>

[4] "AquaJelly." *Festo Corporation*. Festo Corporation, n.d. Web. 1 May 2012.

< http://www.festo.com/cms/en_corp/9772.htm>.

[5] "AquaRay." *Festo Corporation*. Festo Corporation, n.d. Web. 1 May 2012. < http://www.festo.com/cms/en_corp/9786.htm>.

[6] Bhagwat, P.; , "Bluetooth: technology for short-range wireless apps," *Internet Computing, IEEE* , vol.5, no.3, pp.96-103, May/Jun 2001.

[7] Drumheller, Michael; , "Mobile Robot Localization Using Sonar," *Pattern Analysis and Machine Intelligence, IEEE Transactions on* , vol.PAMI-9, no.2, pp.325-332, March 1987.

[8] McDowell, P.; Bourgeois, B.; Iyengar, S.S.; , "Formation maneuvering using passive acoustic communications," *Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on* , vol.4, no., pp. 3843- 3848 Vol.4, April 26-May 1, 2004.

[9] "Mega Fliers." *Mega Fliers*. WOW Products, n.d. Web. 1 May 2012. <http://66.215.26.25/products.html>.

[10] "TB6612FNG.pdf." *Pololulu*. Pololulu, n.d. Web. 1 May 2012. <http://www.pololu.com/file/0J86/TB6612FNG.pdf>.

[11] "MB1010_Datasheet.pdf." *Matrixbot*. Matrixbot, n.d. Web. 1 May 2012. <http://www.maxbotix.com/documents/MB1010_Datasheet.pdf>.

[12] "Essex University." *Essex University*. Essex University, n.d. Web. 1 May 2012. <http://dces.essex.ac.uk/staff/hhu/HCR-Group.html