ECE 6254 Magnet Localization for the Tongue Drive System

Xue Yang, Ghaith Matalkah, Yuanda Zhu, *Georgia Institute of Technology*

***Abstract*—The Tongue Drive System (TDS) is a brain-tongue-computer interface system that allows severely disabled people to control an actuator, such as a wheelchair driving system or computer input device using their tongue [1]. The system comprises a small magnet that is attached to the user's tongue and a set of magnetometers placed near the user's cheeks using headsets. The traditional method of tracking location and movement of a magnet is to solve a triangulation problem that basically solves an inverse magnetic equation for the position and orientation of the magnet. However, this method is very computationally intensive and not suitable for real-time embedded systems like TDS. Alternatively, machine learning offers classification and regression algorithms that can be much less computationally intensive and - if modeled properly - can achieve high accuracy. In this project, my team and I investigated what went wrong in [3], isolated and identified the cause of the degraded performance, proposed and evaluated fixes to the problems, and evaluated alternative regression and classification algorithms.**

***Index Terms*—TDS, machine learning, magnet tracking, linear regression, neural networks, SVR/Support Vector Machine**

I. INTRODUCTION

Initially, the tongue drive system is built as an assistive technology to enable people with severe disability to control their environment such as a computer and a wheelchair [1]. In this research, the tongue drive system is an assistive technology to train paralyzed patients to recover their arm mobility by using the tongue to perform preset moves to control a robot for movement of human arms. A small magnet is attached to the patient's tongue during a human subject trial, and the patient needs to wear a headset with magnetic field receivers to obtain signals from the tongue magnet and send data to the computer. To determine if the user has performed preset tongue movements, a comprehensive data processing and machine learning algorithm has been developed to analyze collected data. The tongue drive system already can issue discrete commands. The purpose of this research is to enable the tongue to issue continuous control in a similar way as people use the trackpad, such as realizing a simplified version of the tongue drive system by tracking a magnet location on a 2D trackpad.

Since a mathematical model of a permanent magnetic dipole has already been developed, and it reached a precision of $4.17*10^{-6}$ mm [2]. The main problem with this magnetic dipole equation model is that it requires very heavy computation power and timing. Therefore, machine learning algorithms are developed to bypass such high computation requirements. In this paper, we will mainly research into the simulation data results and actual data results from three machine learning algorithms: neural networks, support vector machine and LASSO linear regression.

In [3], the authors investigated the same classification and regression solutions for the tongue location and motion tracking with TDS as mentioned above. Since the experimenters did not test the machine learning algorithms on simulated data beforehand, it is hard to discern if the testing result has coding or hardware issues. One way to isolate and identify actual causes of performance degradation is to test machine learning algorithms both on simulation and experimental data, to identify both software and hardware related problems. Ultimately, my goal in this project is to identify the potential sources of errors, then implement alternative classifiers and predictors that will improve the performance by order of magnitude.

Also, the previous team trained three machine learning algorithms on data collected in Figure 2 [3] on a 5 mm precision training data set and tested on 5mm precision testing data set, while my team trained on 10mm

Xue Yang is with the Georgia Tech Bionics Lab, Georgia Institute of Technology, Atlanta, GA 30309 USA in Spring 2017 (e-mail: xue.yang0224@gmail.com).

resolution and tested on 5mm resolution.

The main apparatus of the tongue drive system is illustrated in Figure 1 [1]. Since it would not be easy to collect precision data with a resolution of 0.5 cm in a human mouse, a simplified experimental model emulating the tongue–attached magnet is implemented for this research (Figure 2 [3]).

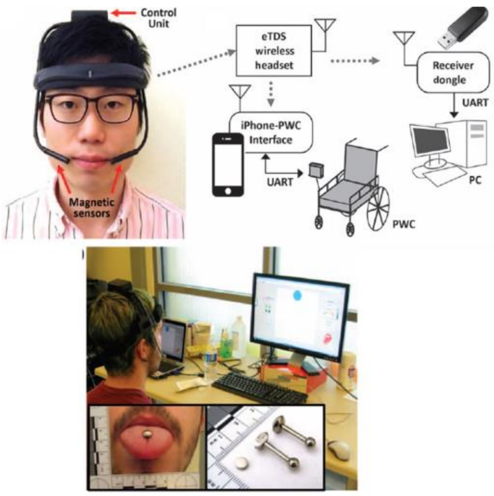


Figure 1. Tongue Drive System

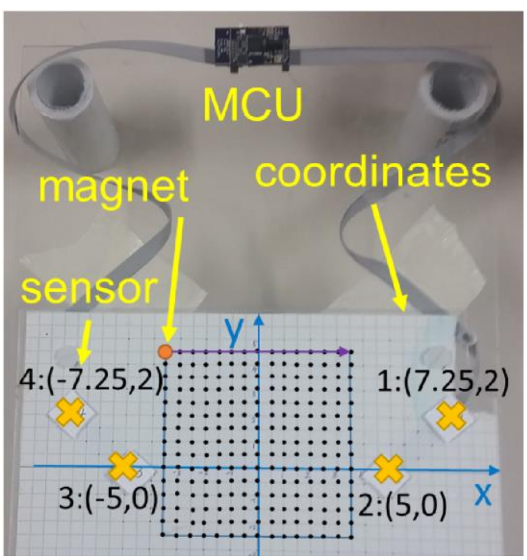


Figure 2. Experimental model emulating the tongue-attached magnet.

II. TASKS COMPLETED

A. Preparation and Lab Related Work

When the project just started at the beginning of the semester, James Zhang prepared me with online machine learning classes from Coursera. Then, I familiarized with the tongue drive system by being the testing subject, as well as the operator recording lab data. We found out that Labview would lose signals from the headset every half an hour. An electromagnetic shielding was later installed on the motors of the tongue drive system to reduce signal interferences. Other than that, I also staffed a few tongue drive system clinical trials.

B. Preprocessing on Data

1) Simulation Data

The simulated data was generated from the sensor datasheet [4]. It was calculated based on magnetic sensitivity and noise density information.

a. Magnetic Distribution Field

The sensors LSM303D in Figure 2 can handle a maximum of 20 gauss disturbance field each. If more than that, the sensor may not function properly. 20 gauss is a large value, because the most of the earth magnetic field is 70000nT [5], which is 0.7 gauss. Same thing applies to another sensor LSM9DS1.

b. Magnetic Sensitivity

The measured magnetic FS (field/ flux strength) is from -12 to 12 gauss, and it was converted to a 16-bit signed integer. Its typical sensitivity is 0.479 mgauss/LSB. This can be understood from below:

if the output signed integer has a 1 LSB value increase, the measured magnetic flux strength is increased by 0.479 mgauss. For example, see TABLE I [6].

TABLE I
TABLE TO DEMONSTRATE MAGNETOMETER OUTPUT AND MAGNETIC FLUX STRENGTH RELATIONSHIP

Magnetometer Output Signed Integer	Magnetic Flux Strength
0000000000000000	0 gauss
000000000000001	0.479 mgauss
000000000000010	0.958 mgauss
000000000000011	1.437 mgauss
000000000000100	1.916 mgauss

c. Magnetic Noise Density

From LSM303D datasheet [4], the magnetic noise density is 5 mgauss/RMS, which is 33 mgauss/pk-pk.

The noise has a Gaussian distribution from [7]. A conversion rate of 6.6 is used to convert RMS to peak to peak values [8]. This covers 99.9% of the Gaussian distribution.

Noise is converted to magnetometer output values in simulated data. The magnetometer output is affected by noise of (33 mgauss/pk-pk) / (0.479 mgauss/LSB), which leads to 68.89 LSB/pk-pk or 10.44 LSB/RMS.

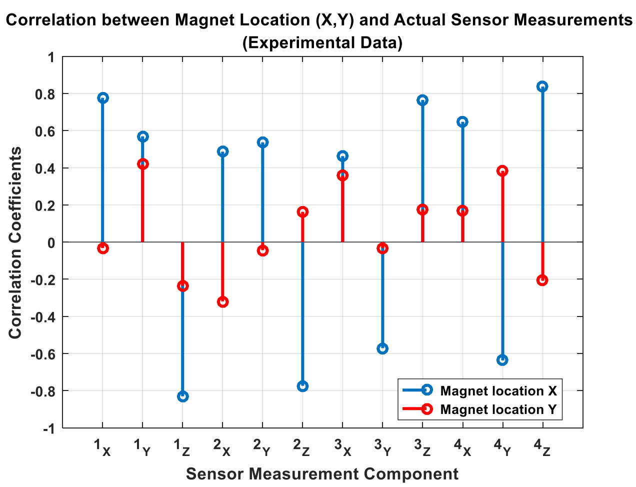
2) *Machine Learning Algorithms*

a. Artificial Neural Network

Artificial neural network is implemented in python by using built-in function multi-layer perceptron, MLPClassifier. This classifier is a supervised learning algorithm that uses backpropagation.

3) *Testing Process*

On the machine learning project side, key observations were made when initially analyzing data. The data collected was from magnetic flux received by four magnetic sensors located on the marked places of Figure 2. They were configured to sample the magnetic flux generated by a magnet placed within the grid at 100 samples/sec, where each sample is a 16-bit signed integer representing a magnetic flux value between -12 and +12 gausses [3]. The data collected at each point has 12 total dimensions: the x, y, and z axis measurements from four sensors (z = 0 here since the data was collected on a 2D plane). When initially analyzing the data, we found that Y is less correlated to the sensor measurements. The correlation analysis graph of the training dataset was attached below:



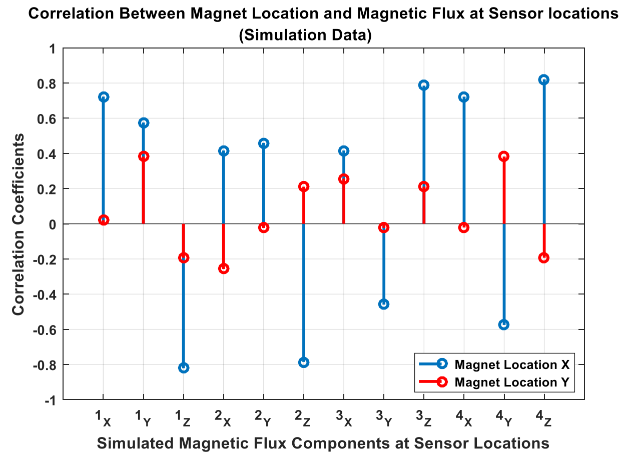
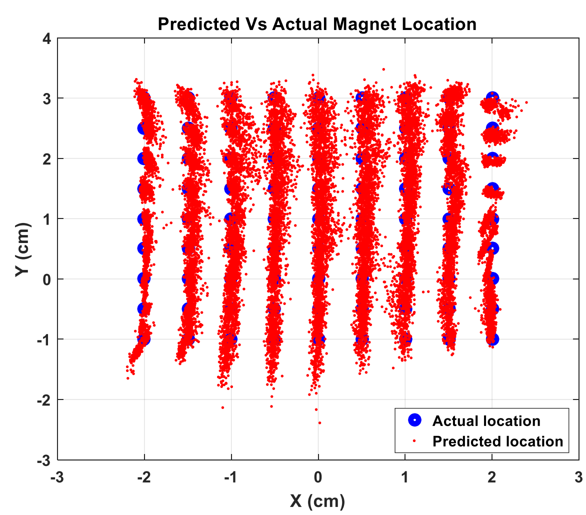


Figure 3. Correlation Analysis of Both the Simulation Data and the Experimental Data

It can be interpreted from Figure 3 that the sensor measurements are a lot less sensitive to magnet movements along the Y axis than movements along the X axis.

When analyzing predicted locations of both X and Y predictions, it can be understood from Figure 4 that Y prediction error is the main contributor to the RMSE (Root Mean Square Error); Y prediction error is increasing the overall magnet location prediction error.

This pattern is applied to not only linear regression algorithm, but also to support vector machine and neural network algorithms.



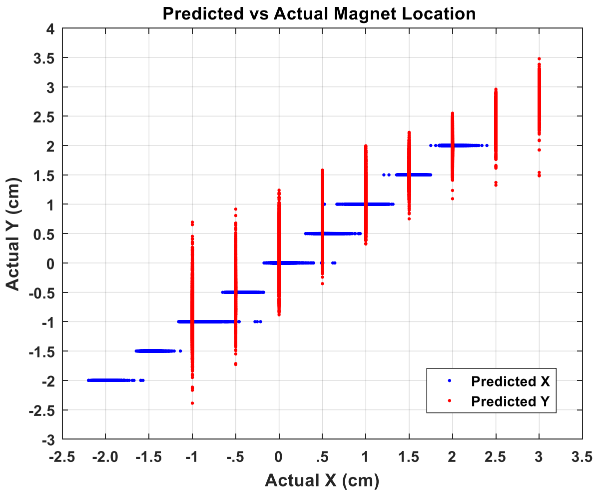


Figure 4. Top Graph: Predicted v.s. Actual Magnet Location, RMSE = 0.19 cm; Bottom Graph: Predicted X and Y Locations Separately, RMSE(X) = 0.07 cm, RMSE(Y) = 0.28 cm.

One solution proposed is to drop the lowest correlated components when predicting Y. This method is not only more accurate but also saves time when running linear regression algorithm.

Another solution to these two issues is to train and predict X and Y separately. My team was able to achieve better results this way. Below is TABLE II showing the best results from all machine learning algorithms.

TABLE II
PREDICTION ERROR -RMSE (CM)

Regression Method	Kernelized Ridge Regression	SVR	Neural Networks
Test 1	*0.21*	*0.32*	*0.076*
Test 2	*0.74*	*0.81*	*0.5*
Test 3	*0.27*	*0.32*	*0.348*
Test 4	*0.25*	*0.29*	*0.563*
Test 5	*0.19*	*0.25*	*0.792*

The parameters in each algorithm were tuned separately for X and Y training data and the combined RMSE for each test was much better, compared to training X and Y together. The parameters for kernelized ridge regression is X: α =0.001, γ =0.01, Y: α =0.01, γ =0.1; the radial basis function (RBF) kernel SVR has parameters of C=10000, γ=0.0001, Y: C=100, γ=0.1; and neural networks, whose parameter alpha equals to 1e-5, are used with 250 hidden layers for reaching optimum performance.

The ridge regression and SVR parameters are tuned on the whole dataset while neural network L2 regularization parameter alpha value is not.

It is worth mentioning that the neural networks take the least amount of training time, while SVR is still relatively fast. Linear regression (such as Linear LASSO regression) takes the longest time. When training with seven trials of data and 80 points collected per point on the grid, it can take up to 36 hours to process data. After dropping the least correlated features of Y during training, the linear regression training time has dropped.

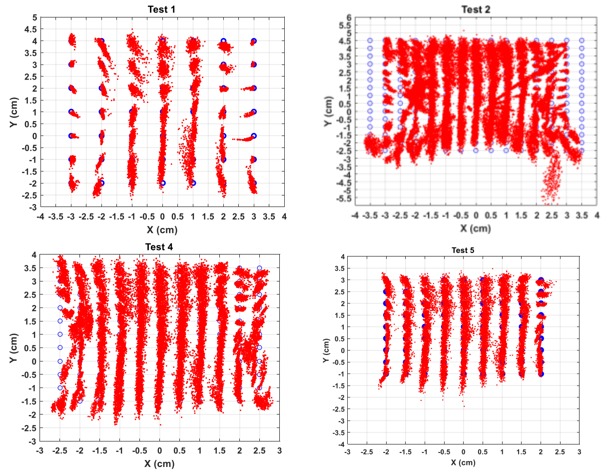


Figure 5. Linear Regression Testing Results (Test 3 and 4 results were similar)

4) Conclusion

When X and Y are predicted independently, kernelized linear regression results are 50% better than all previous results. SVR results are almost the same. Neural Network performs extremely well on some test, but poorly on other tests.

5) Recommendation for Future Work

One potential improvement is to install sensors along the Y axis of the grid (lateral to X) or use better sensors with sensitivity to changing in magnet location along Y axis.

Also, the neural network's alpha parameter can be tuned for the whole span of data to for better prediction results.

C. Simulation Work Requested by Zhang

Similar experimental data results were reproduced from Zhang's paper. Also, simulation data were run with the same machine learning algorithms. It can be seen from the simulation data prediction results that better predictions were produced from most of the simulation data.

III. Conclusion

Hardware limitations is a potential source of error for the machine learning algorithms. It would be helpful to install sensors along the Y axis of the grid (lateral to X) or use better sensors with higher sensitivity to change in magnet location along Y axis.

References

[1] J. Kim, et al, "The tongue enables computer and wheelchair control for people with spinal cord injury," *Science Translational Medicine,* vol. 5(213), no. 166, p. 166, Nov. 2013.

[2] A. Farajidavar, J. M. Block and M. Ghovanloo, "A comprehensive method for magnetic sensor calibration: A precise system for 3-d tracking of the tongue movement," in *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2012.

[3] Z. Zhang, M. Kagie, S. Shah and F. Jafri, "2D Regression and Classification for Tongue Drive," ECE 6254 Final Project Report, 2016.

[4] STMicroelectrnics. *Ultra-compact high-performance eCompass module: 3D accelerometer and 3D magnetometer* [Online]. Available FTP: http://www.st.com/content/ccc/resource/technical/document/datasheet/1c/9e/71/05/4e/b7/4d/d1/DM00057547.pdf/files/DM00057547.pdf/jcr:content/translations/en.DM00057547.pdf

[5] SBG Systems. (2014, Jan 14). *Magnetic Field and Disturbances* [Online]. Available: http://support.sbg-systems.com/customer/en/portal/articles/1408684-magnetic-fields-and-disturbances

[6] rberek. (2012, April 27). [Online]. Available: http://www.edaboard.com/thread250221.html

[7] Honeywell. (2008, August). *1- and 2-Axis Magnetic Sensors HMC1001/1002/1021/*1022 [Online]. Available: https://neurophysics.ucsd.edu/Manuals/Honeywell/HMC%201001%20and%20HMC%201002.pdf

[8] Analog Devices, Inc. (2010, Jan 19). *Calculating RMS Noise to Peak-to-Peak Noise* [Online]. Available: https://www.youtube.com/watch?v=-KcODSYXiZA