Shahjalal University of Science and Technology

Department of Computer Science and Engineering

Air Writing For Bangla Digits and Alphabets Using Android Device

Student:

Soumik Kumar Saha, 2010331003, 4/1, CSE

Mohammed Mostakim Ornob, 2010331009, 4/1, CSE

Watina Malek, 2010331045, 4/1, CSE

Adviser:

Sabir Ismail

Lecturer

Computer Science and Engineering

30th March, 2015

Air Writing For Bangla Digits and Alphabets Using Android Device

A Thesis/Project submitted to the Department of Computer Science and Engineering, Shahjalal University of Science and Technology, in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science and Engineering.

Student:

Soumik Kumar Saha, 2010331003, 4/1, CSE

Mohammed Mostakim Ornob, 2010331009, 4/1, CSE

Watina Malek, 2010331045, 4/1, CSE

Adviser:

Sabir Ismail

Lecturer

Computer Science and Engineering

30th March, 2015

Recommendation Letter from Thesis Supervisor

These Students, Soumik Kumar Saha, Mohammed Mostakim Ornob, Watina Malek whose thesis entitled “Air Writing for Bangla Digits and Alphabets Using Android Device”, is under my supervision and agree to submit for examination.

Advisor :

Sabir Ismail

Date : / /

30th March, 2015

Qualification Form of Bachelor Degree

Student Name : Soumik Kumar Saha, Mohammed Mostakim Ornob, Watina Malek

Thesis Title : Air Writing For Bangla Digits and Alphabets Using Android Device

This is to certify that the thesis submitted by the student named above in March, 2015. It is qualified and approved by the Thesis Examination Committee.

Head of the Dept.	Chairman, Thesis Committee	Supervisor
Prof. Dr. Md. Shahidur Rahman	Abu Naser	Sabir Ismail

Abstract

Our thesis is about Air Writing for Bangla digits and alphabets Using Android device. We used accelerometer and gyroscope sensor of android device to recognize the segmented structures of Bangla alphabets and digits as defined types of strokes.

Our thesis is an ongoing work from our former seniors. We started from where they ended their work. They did it for English numbers and Alphabets. They used data glove consists of sensors, microcontrollers, Bluetooth chip and the power supply.

As future work in next semester, we will implement it for every Bangla digits and alphabets. And we will also implement it in real life application.

Keywords

Air Writing, Stroke, Sensor, Accelerometer, Gyroscope, Noise, Android

Acknowledgments

During this thesis we have faced lots of difficulties and found many people in the way that helped us a lot and without their supports it would be almost impossible to continue our work. So we would like to acknowledge them here. First and foremost, we are deeply indebted to our supervisor, Sabir Ismail sir, for his advice and for being a ceaseless motivator during all the time of thesis.

We are really grateful and our humble regards to our supervisor for continuous support. We have found him all the time with us while we are working in our lab. He expresses his hand of help whenever we faced a problem and gave live recovery.

We are also grateful to our seniors MD. Munif Hasan (2009331020), MD. Asfaq Sufian (2009331030), Rajesh Baidya (2009331038). Their thesis work is the inspiration behind our work.

TABLE OF CONTENTS

Page

List of Tables

List of Figures

Illustration of Symbols

LURB: Left Up Right Bottom

RULB: Right Up Left Bottom

CHAPTER 1

INTRODUCTION

This is the era of smart and efficient technologies. The increasing number of gesture-control technologies being introduced these days mean that a time may actually come when you can control your whole house with just the wave of a hand. The days of reaching for a notepad, phone or tablet in the middle of the night to scrawl down a note to yourself might soon be over thanks to a system that translates what you write in the air into editable text. And this is called Air Writing.

In this thesis we proposed a way in which we can represent air writing for Bangla alphabets and digits using android device. First, we analyzed segmented structure of Bangla alphabets and digits and divided them into strokes. Then, we analyzed the outputs generated by accelerometer of strokes in 2D plane. Then we implement this on some of the Bangla digits on android device.

Bengali alphabets and digits have more curve than English alphabets and digits. So firstly we are doing this for segmented structure for Bengali alphabets and digits. And then we will do it for continuous structure in near future.

1.1 Background

1.1.1 Air Writing

A system that translates what you write in the air into editable text is called Air writing. In air writing, the acceleration due to hand gestures can be translated into geometric strokes, and recognized as characters.

1.1.2 Segmented Characters:

Segmented characters are usually used in the digital segmented displays. The stricter of the characters are developed in such a way so that the structure can be represented with only linear segments. For Bangla characters some researchers have developed segmented structures.

1.2 Motivation

Now we are in a world where everyone wants to make their life easier and save time and this is the era of touch screen and hand gesture. A smart technology has been invented to write in the air using different types of hardware.

We did some background work on it and wanted to try something new with it. We studied some papers about Air writing for English alphabets and digits. And they did it with data glove using sensors and microcontrollers.

But there is nothing for Bangla alphabets and digits. So we decided to do it for Bangla alphabets and digits using android device.

CHAPTER 2

Background Study

We started our thesis from previous work of our senior MD. Munif Hasan (2009331020), MD. Asfaq Sufian (2009331030), Rajesh Baidya (2009331038). Though they used data glove consists of sensors, microcontrollers, Bluetooth chip and the power supply for English Alphabets and Digits, we have used android device for Bangla alphabets and digits.

We studied a published paper *"A Unique 10 Segment Display for Bengali Numerals"* by *Md. Abul Kalam Azad, Rezwana Sharmeen, Shabbir Ahmad and S. M. Kamruzzaman, Department of Computer Science & Engineering, International Islamic University Chittagong, Chittagong, Bangladesh & Department of Computer Science & Engineering, Manarat International University, Dhaka, Bangladesh* to analyze the segmented structures of Bangla digits. They used segments of uniform size and no bent segment is used.

We also studied a published paper *"SEGMENTED DISPLAY SYSTEM FOR BENGALI ALPHABET"* by *Salahuddin Mohammad Masum, Swapon Chandra Dash, Sarwar Morshedul Haque, Kazi Faisal Kabir, Faculty of Science & Information Technology, Daffodil International University, Dhaka 1207, Bangladesh* to analyze the segmented structures of Bangla alphabets. They displayed segmented structures of Bangla alphabets without using Dot Matrix.

We also went through a published paper *"Using Mobile Phones to Write in Air"*. This paper envisions a system called *PhonePoint Pen* on Nokia N95 that uses the in-built accelerometer in mobile phones to recognize human writing. But the values of sensors vary for different devices. So we are doing this using android device for Bangla alphabets and digits.

CHAPTER 3

Methodology

In this chapter we describe our work and our experiment and its result analysis. We are doing all our experiments with Segmented Bangla Characters. As the structure of generalized written form of Bangla characters is little bit complex. So at the first stage we are working with Segmented Bangla Characters. After we implement our desired system for Segmented Bangla Characters we will also implement the system for generalized written form of Bangla characters. At the end of this chapter we added summary of our experiment.

3.1 Analyzing the Structures of Segmented Bengali Digits and Alphabets

Our first experiment is what structure we have found analyzing the segmented Bangla digits and alphabets. We used the reference *"A Unique 10 Segment Display for Bengali Numerals"* for Bangla digits and *"Segmented Display System For Bengali Alphabet"* for Bangla alphabets.

3.1.1 Procedure:

We collected and went through some published journals on segmented Bangla digits and alphabets and analyzed the structure of the characters. We have carefully looked into the structures of the characters to find out any ambiguity.

3.1.2 Data collection:

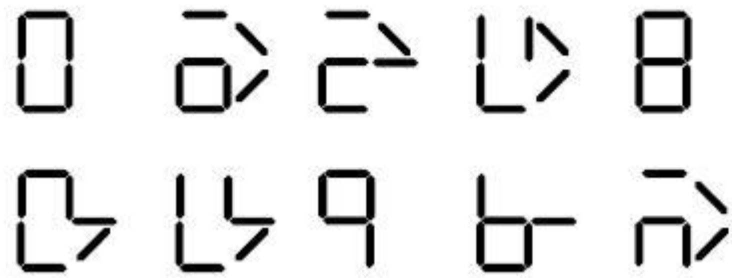

Figure 1: Proposed Segment display of Bengali digits

Table 1: No. of strokes used in each digit

Digits	-	|	\	/
০	2	2	0	0
১	3	2	1	1
২	3	1	1	0
৩	1	2	1	1
৪	3	4	0	0
৫	3	2	0	1
৬	2	2	0	1
৭	2	2	0	0
৮	2	2	0	0
৯	2	2	1	1

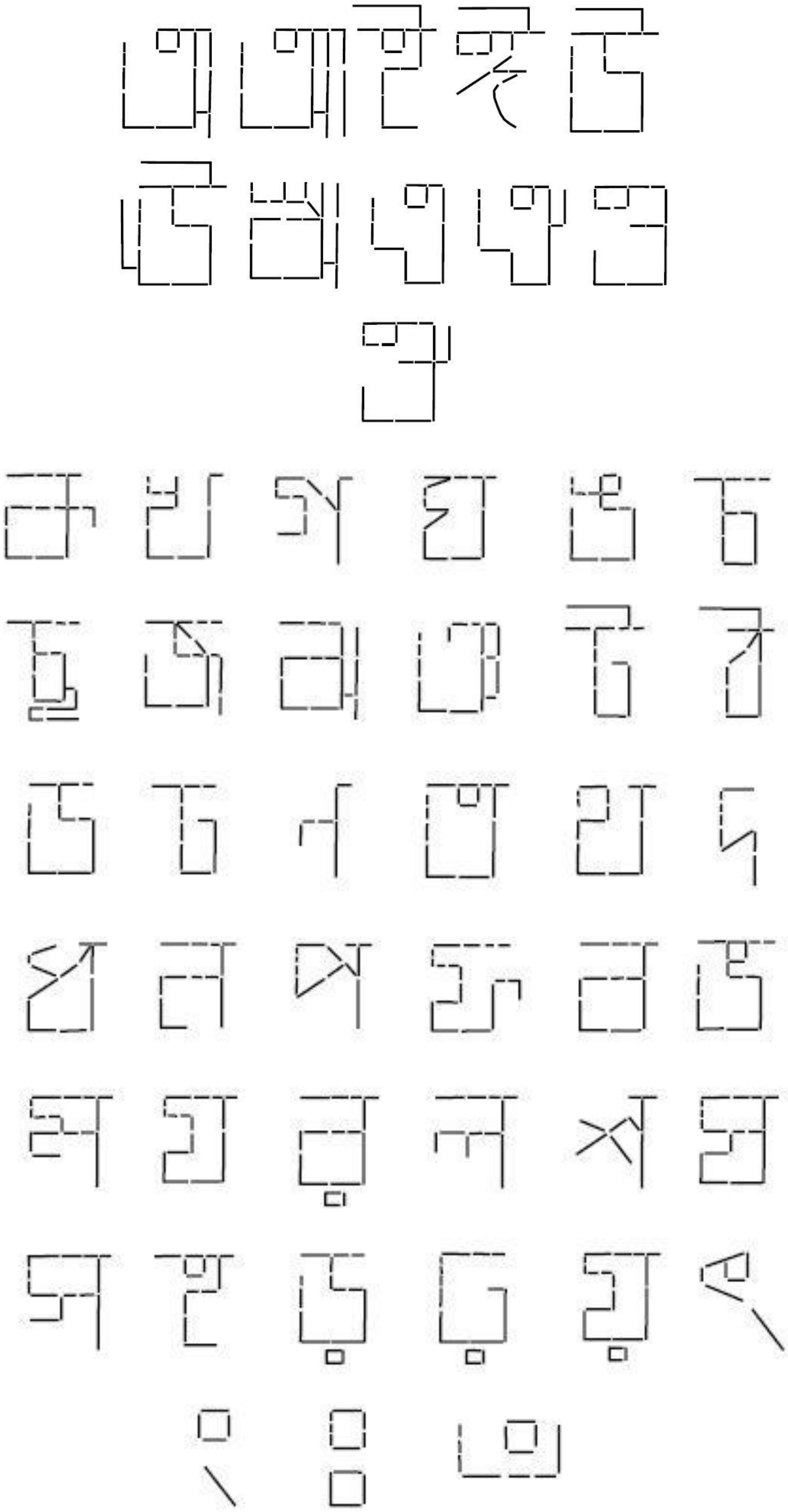

Figure 2: Proposed Segment display of Bengali alphabets

Table 2: No. strokes used in each alphabet

Alphabet	-	|	\	/
অ	4	5	0	0
আ	4	6	0	0
ই	5	5	0	0
ঈ	4	5	1	2
উ	4	4	0	0
ঊ	5	5	0	0
ঋ	4	6	1	0
এ	4	5	0	0
ঐ	5	6	0	0
ও	4	4	0	0
ঔ	5	5	0	0
ক	3	3	0	0
খ	4	4	0	0
গ	4	3	1	0
ঘ	3	3	0	2
ঙ	4	5	0	0
চ	3	2	0	0
ছ	6	4	0	0
জ	4	4	1	0
ঝ	4	3	0	0
ঞ	5	5	0	0
ট	4	3	0	0
ঠ	3	3	0	1
ড	3	3	0	0
ঢ	3	2	0	0
ণ	2	2	0	0
ত	3	4	0	0
থ	4	4	0	0
দ	1	2	0	1
ধ	2	3	1	2
ন	3	2	0	0
প	2	2	1	1
ফ	5	5	0	0
ব	3	2	0	0
ভ	4	4	0	0
ম	4	4	0	0
য	4	4	0	0

Alphabet	-	|	\	/
র	5	4	0	0
ল	2	3	0	0
শ	1	1	2	1
ষ	5	4	0	0
স	3	3	0	0
হ	4	4	0	0
ড়	5	5	0	0
ঢ়	5	4	0	0
য়	6	6	0	0
ৎ	1	4	1	1
ং	2	2	1	0
ঃ	4	4	0	0
৺	3	4	0	0

- Red characters denotes ambiguous character
- Black characters denotes non-ambiguous character

3.1.3 Analysis and Summary

After analyzing the segmented structure we divide each segmented alphabets into four types of strokes:

- Horizontal stroke (-)
- Vertical stroke (|)
- Left up-Right bottom tilted stroke (\ )
- Right up-Left bottom tilted stroke (/)

We also find how many of each stroke is used in the segmented structure of Bengali alphabets and digits.

Then we analyzed the segmented structure of each alphabets and digits to find if there is any ambiguity. We find ambiguity in (০, ৭, ৮) , (অ, এ, ঙ) , (ই, ঊ, ঔ, ঞ, ফ, ড়) , (উ, ও, খ, থ, ভ, ম, য, হ, ঃ) , (ক, ড, স) , (চ, ঢ, ন, ব) , (ঝ, ট) , (ত, ৺) , (র, ষ, ঢ়).

We have seen that Bangla Segmented characters have lots of ambiguity. So it is totally impossible to implement Bangla Air Writing without reducing these ambiguities. These would require a out of the box approach to do so.

3.2 Motion Sensors

3.2.1 Accelerometer

An **accelerometer** is a device that measures proper acceleration ("g-force"). Proper acceleration is not the same as coordinate acceleration (rate of change of velocity). For example, an accelerometer at rest on the surface of the Earth will measure an acceleration g= 9.81 m/s^2 straight upwards. By contrast, accelerometers in free fall orbiting and accelerating due to the gravity of Earth will measure zero.

An accelerometer measures proper acceleration, which is the acceleration it experiences relative to free-fall and is the acceleration felt by people and objects. Put another way, at any point in space time the equivalence principle guarantees the existence of a local inertial frame, and an accelerometer measures the acceleration relative to that frame.[1] Such accelerations are popularly measured in terms of g-force.

3.2.2 Gyroscope

A **gyroscope** is a device for measuring or maintaining orientation, based on the principle of preserving angular momentum.

Mechanical gyroscopes typically comprise a spinning wheel or disc in which the axle is free to assume any orientation. Although the orientation of the spin axis changes in response to an external torque, the amount of change and the direction of the change is less and in a different direction than it would be if the disk were not spinning.

Vibration gyro sensors sense angular velocity from the Carioles force applied to a vibrating element. For this reason, the accuracy with which angular velocity is measured differs significantly depending on element material and structural differences.

3.2.3 Motion Sensors in Android Device:

Sensors used in the android devices are not the mechanical ones. Manufactures use electronic motion sensors in the android devices. Behavior of the electronic motion sensors is different from mechanical motions sensors. So to use the motion sensors in android devices require more study.

3.2.3.1 Accelerometer in Android:

An acceleration sensor measures the acceleration applied to the device, including the force of gravity. Conceptually, an acceleration sensor determines the acceleration that is applied to a device (A_d) by measuring the forces that are applied to the sensor itself (F_s) using the following relationship:

$A_d = - \sum F_s / mass$

However, the force of gravity is always influencing the measured acceleration according to the following relationship:

$A_d = -g - \sum F / mass$

For this reason, when the device is sitting on a table (and not accelerating), the accelerometer reads a magnitude of $g = 9.81\ m/s^2$. Similarly, when the device is in free fall and therefore rapidly accelerating toward the ground at $9.81\ m/s^2$, its accelerometer reads a magnitude of $g = 0\ m/s^2$. Therefore, to measure the real acceleration of the device, the contribution of the force of gravity must be removed from the accelerometer data. This can be achieved by applying a high-pass filter. Conversely, a low-pass filter can be used to isolate the force of gravity.

We need to eliminate the gravitational effect from the raw accelerometer reading. The following mathematical equation is used to do so.

$$linear\_acceleration_{x/y/z} = raw\_acceleration_{x/y/z} – gravity_{x/y/z}$$

Where value of the gravity is

$$gravity_{x/y/z} = 0.8 * gravity_{x/y/z} + (1 - 0.8) * raw\_acceleration_{x/y/z}$$

After eliminating the gravity then we can use the values for further calculations.

3.2.3.2 Gyroscope in Android:

The gyroscope's coordinate system is the same as the one used for the acceleration sensor. Rotation is positive in the counter-clockwise direction; that is, an observer looking from some positive location on the x, y or z axis at a device positioned on the origin would report positive rotation if the device appeared to be rotating counter clockwise. This is the standard mathematical definition of positive rotation and is not the same as the definition for roll that is used by the orientation sensor.

Standard gyroscopes provide raw rotational data without any filtering or correction for noise and drift (bias). In practice, gyroscope noise and drift will introduce errors that need to be compensated for. The drift (bias) and noise are usually determined by monitoring other sensors, such as the gravity sensor or accelerometer.

We need to reduce the noise to use the sensor's readings for further calculations. We can do so by using the following mathematical equations.

$$\text{axis}_{x/y/z} = \text{axis}_{x/y/z} \ / \ \text{omega\_magnitude}$$

If and only if

$$\text{omega\_magnitude} > \text{EPSILON}$$

Where EPSILON represents your maximum allowable margin of error

We can calculate omega_magnitude using the following equation

$$\text{omega\_magnitude} = \sqrt{(\text{axisX} * \text{axisX} \ + \ \text{axisY} * \text{axisY} \ + \ \text{axisZ} * \text{axisZ})}$$

Where axisX = gyroscope X-axis reading

axisY = gyroscope Y-axis reading

axisY = gyroscope Z-axis reading

After eliminating the noise then we can use the values for further calculations.

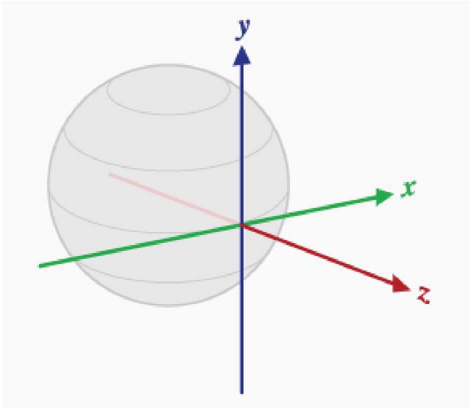

Figure 3: Co-Ordinate System Used by the Motion Sensors

3.3 Analysis of the Outputs Generated by Accelerometer of Strokes in 2-D Plane

Our second experiment is to find out how accelerometer behaves while drawing the strokes in 2-D plane.

3.3.1 Test Bed

We used Walton Primo X1 Android device containing Bosch BMA050 3-Axis Accelerometer to do this experiment.

3.3.2 Procedure

Here we make a comparison between the output of accelerometer in both X and Y axis for various strokes. We developed an android application to gather the accelerometer data and plotted these data on graph and analyzed them. We have used "SensorManager.DELAY_NORMAL" for getting data on sensor value change.

3.3.3 Data collection & Graph plotting

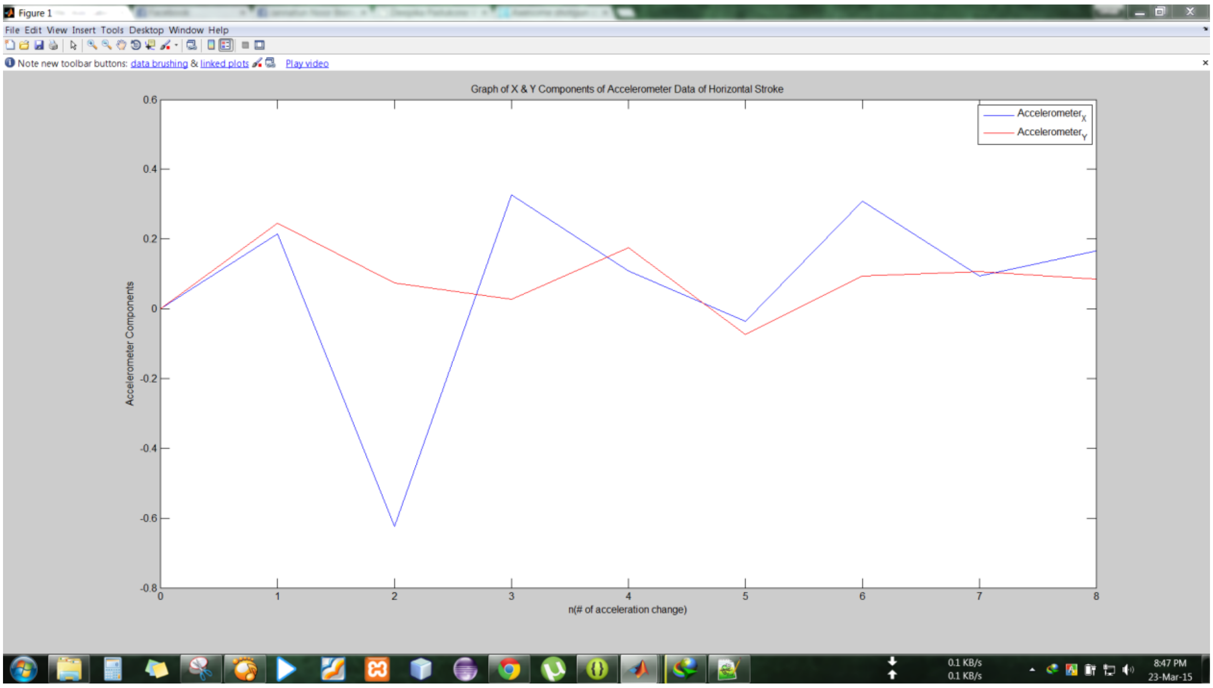

Figure 4: Individual Plotting of Accelerometer Data of Horizontal Stroke

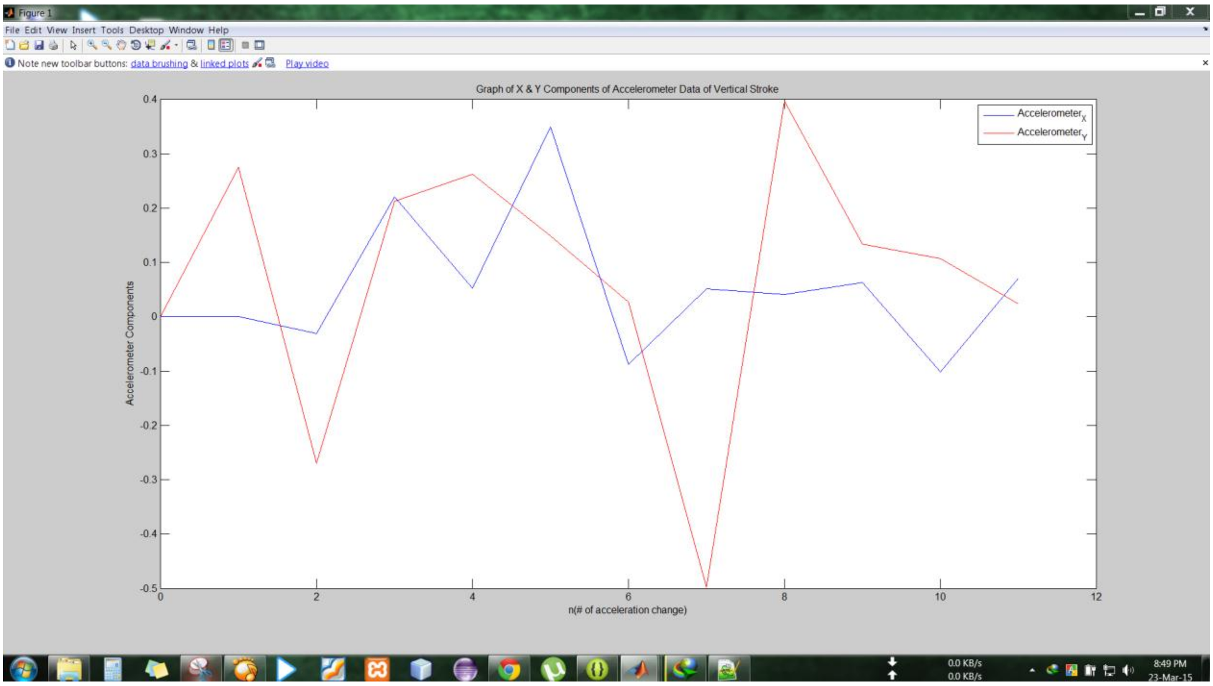

Figure 5: Individual Plotting of Accelerometer Data of Vertical Stroke

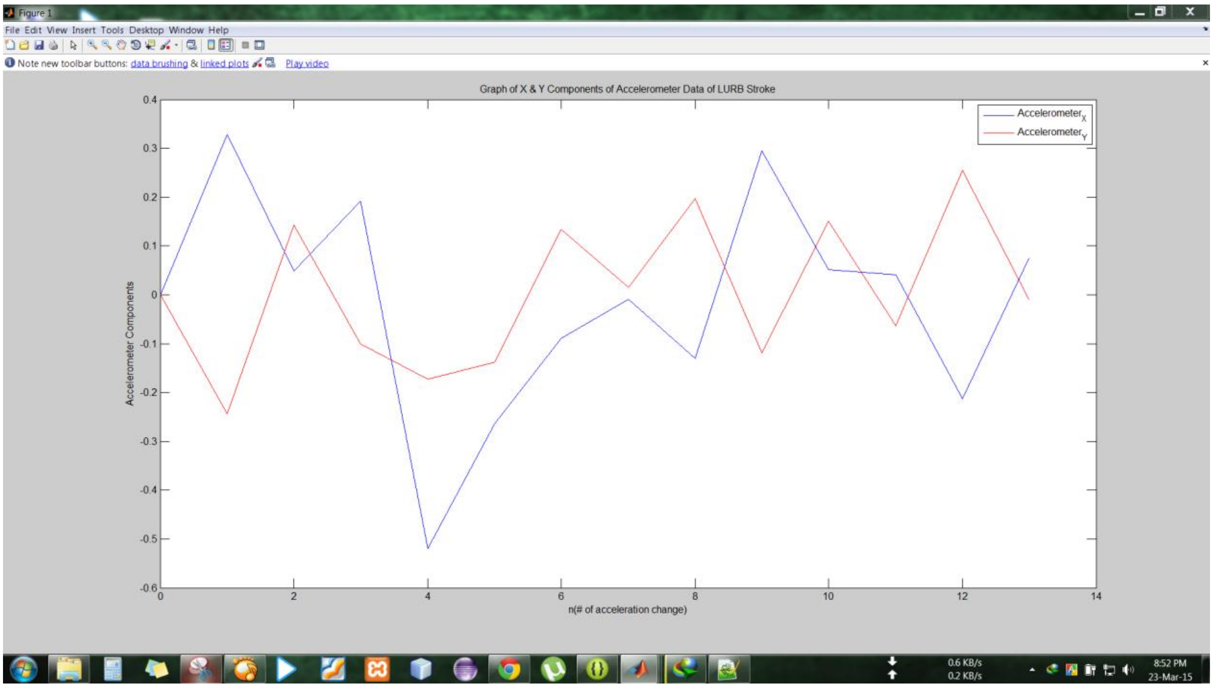

Figure 6: Individual Plotting of Accelerometer Data of LURB Tilted Stroke

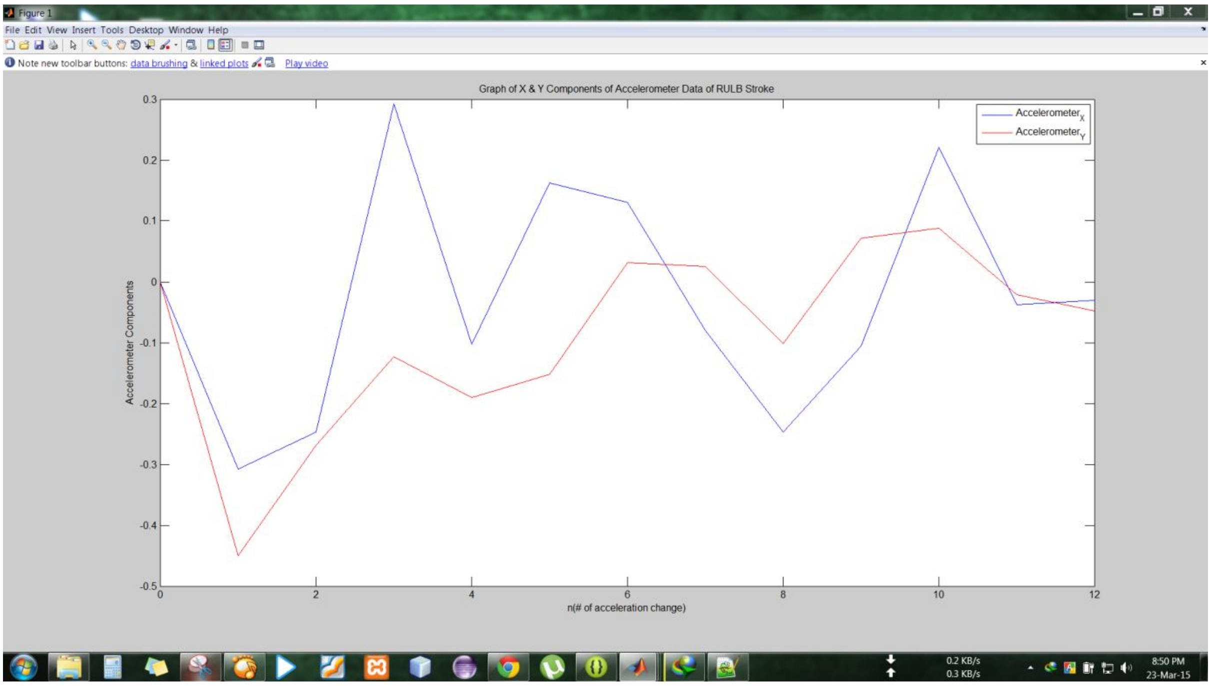

Figure 7: Individual Plotting of Accelerometer Data of RULB Tilted Stroke

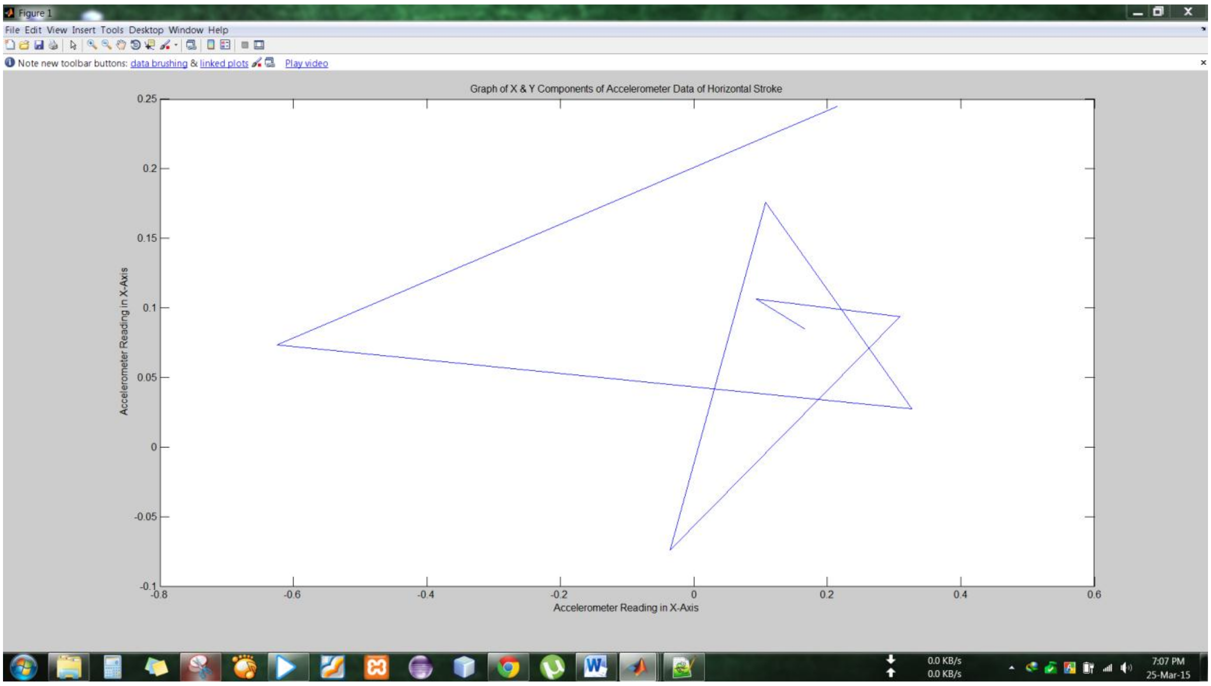

Figure 8: Joint Plotting of Accelerometer Data of Horizontal Stroke

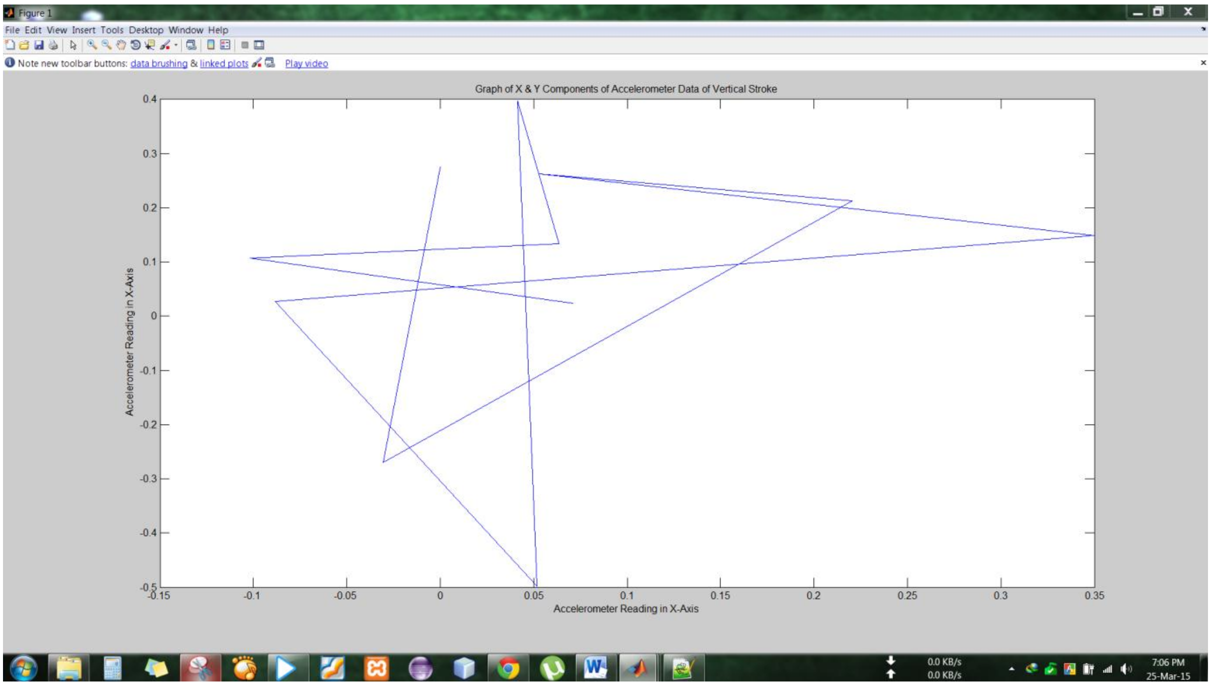

Figure 9: Joint Plotting of Accelerometer Data of Vertical Stroke

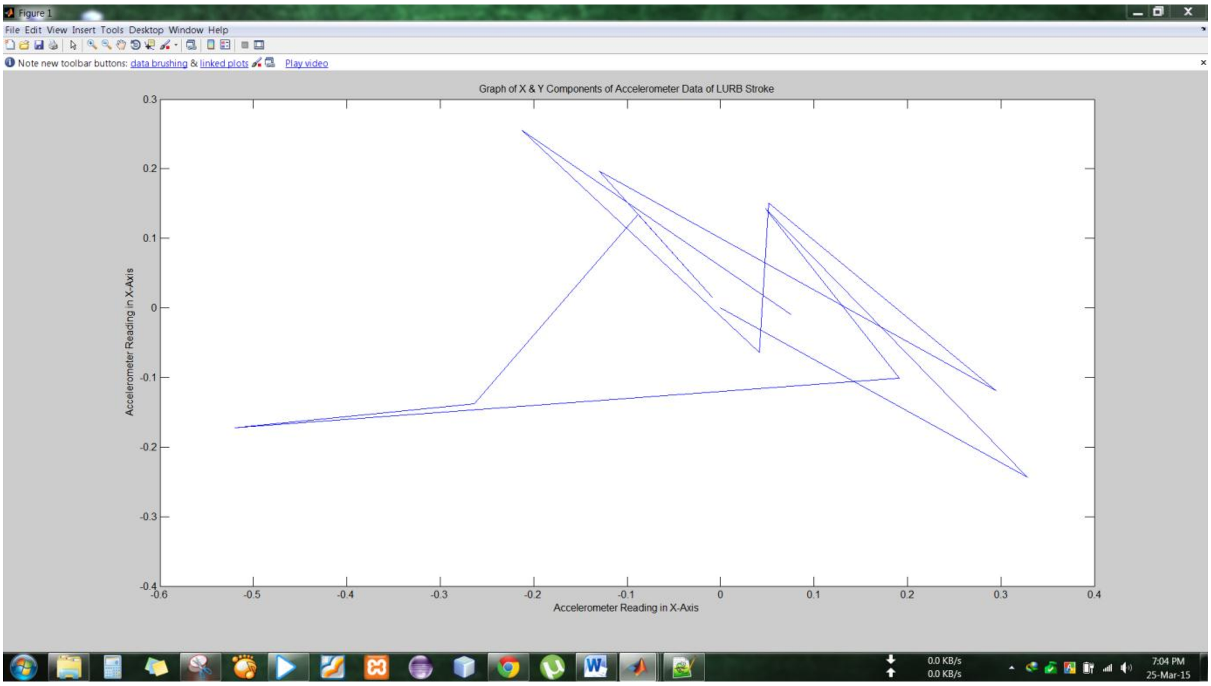

Figure 10: Joint Plotting of Accelerometer Data of LURB Tilted Stroke

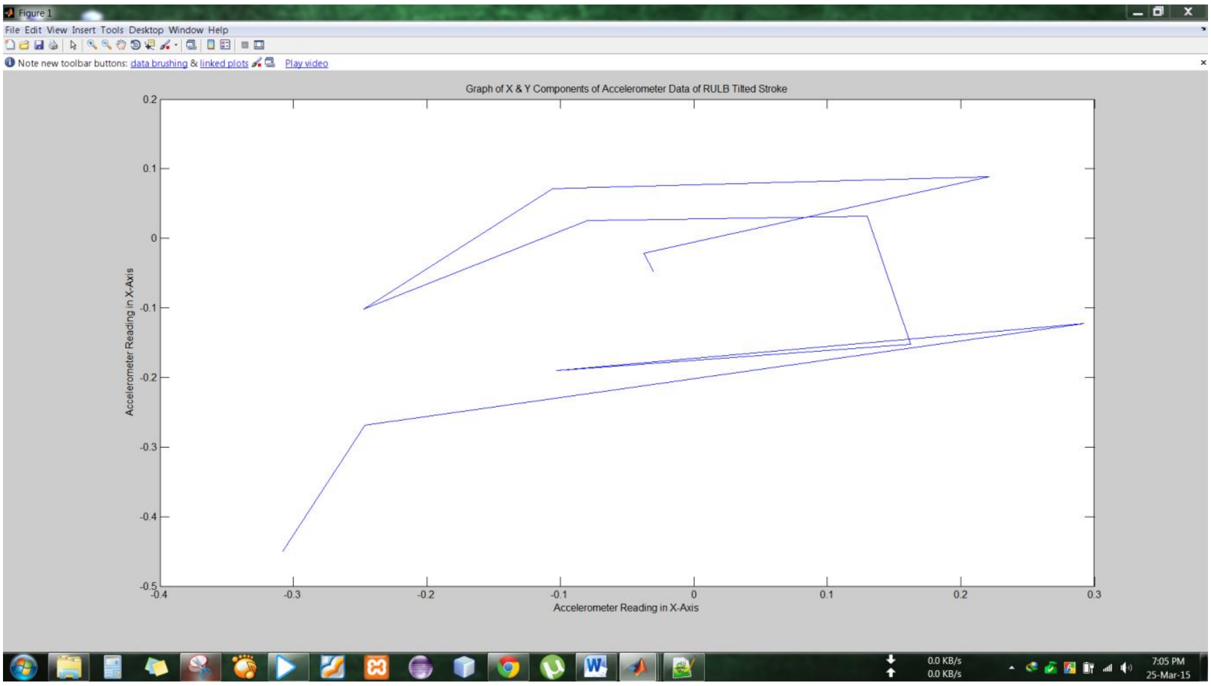

Figure 11: Joint Plotting of Accelerometer Data of RULB Tilted Stroke

3.3.4 Analysis and summary

The output data of accelerometer corresponding to X and Y axis of various strokes are very confusing. We do not find any distinct pattern of the strokes from the data. So, as a result we cannot derive any result of recognizing strokes only from supervising these data.

CHAPTER 4

Implementation

We faced a few hurdles during the implementation of the decisions found during our experiments. Among them some we have been able to solve and some have not. With the solutions of the hurdles we have implementation a system and developed an android app to recognize the Bangla Segmented Digits.

4.1 Implementation of Horizontal and Vertical Stroke Recognition and Recognition of Characters Consists of only Horizontal and Vertical Strokes

4.1.2 Hurdles:

We faced following hurdles during the implementation:

- Reducing the noise and gravitational effect from the raw accelerometer data.
- Detecting individual strokes.
- Deriving any pattern from the analysis from the corrected Accelerometer data.
- Recognizing the points lifting hand during air writing characters.
- Handshakes during the writing affect the accelerometer reading.

4.1.3 Methodology:

To overcome the hurdles we adopted some methodologies. These are:

- We used some mathematical calculations to reduce noise and gravitational effect from accelerometer readings. We have discussed about this method in section 3.2.3.1.
- As we could not derive any pattern from the accelerometer data we adopted an out of the box approach to recognize vertical and horizontal strokes. For practical reason during the writing in air or in 2-D plane our handshakes. For this reason unwanted noise is produced. This cannot be reduced. During writing a horizontal stroke the accelerometer sensor readings in X-axis are supposed to be zero and readings in Y-axis are supposed to be either positive or negative. But in practical field the readings in both axes are very far from theoretical expectations. This also happens for the other stokes. The methodology we have adopted to solve this problem is based on the distance moved to write a stroke. If we try to draw a horizontal stroke then there should be a counter shake for each shake in the Y-axis either upward or downward. In practical this does not happen for every shake but for most of the shakes. Counter shakes neutralizes the distance moved in Y-axis to significant amount. In X-axis the summation of distance moved is not that much affected by shakes.
- The same method we have used to recognize the vertical strokes.
- To calculate the distance moved we have used an equation derived from Newton's laws of motion. As we know during writing a single stroke the initial and final velocity is zero. We have the accelerations in each point of acceleration change. We calculated the distance moved in both axes and compared them. The more distance moved in X-axis means the written stoke is horizontal and vice-versa. We used the following equation to calculate distance.

$$v^2 = u^2 - 2 * a * s$$

Where v = final velocity, u = initial velocity, a = acceleration, s = distance moved

This equation works for the system where there is no external force. But we are pushing our device constantly towards a direction which creates external force. But this force is neutralized by the frictional force. So it affects less in the result. We also work the acceleration in small segments. Though the result we get is not correct but reliable enough to work with.

- Recognizing a single stroke is not enough to detect a character. We need to distinguish each stroke of a character during continuous writing. To do so we need to recognize every starting and end of each stroke. To solve this hurdle we have calculated the resultant acceleration. The ''tends to zero'' value for resultant acceleration means halt and the opposite means the movement. By calculating resultant acceleration we now can distinguish every stroke of a character. We have used the following equation to calculate resultant acceleration.

$$\text{resultant\_acceleration} = \sqrt{(\text{acceX} * \text{acceX} + \text{acceY} * \text{acceY} + \text{acceZ} * \text{acceZ})}$$

4.1.4 Calibration:

This system needs some time to calibrate the sensors at startup. Calibration needs some random writing after the sensors are registered.

4.1.5 Range:

As we still cannot detect Left up-Right bottom tilted and Right up-Left bottom tilted stroke, the app now can only detect Bangla segmented digits having only horizontal and vertical stroke. We are still not able to solve the hand lifting problem. This also cannot detect the digits those require to lift hand during writing. With this system we can detect the distinguish strokes of ০, ১, ৭, ৮. But we distinguish these four digits because of their structures being ambiguous.

4.1.6 Efficiency:

After the calibration process the system does pretty fine job. It produces 70% efficiency within its range.

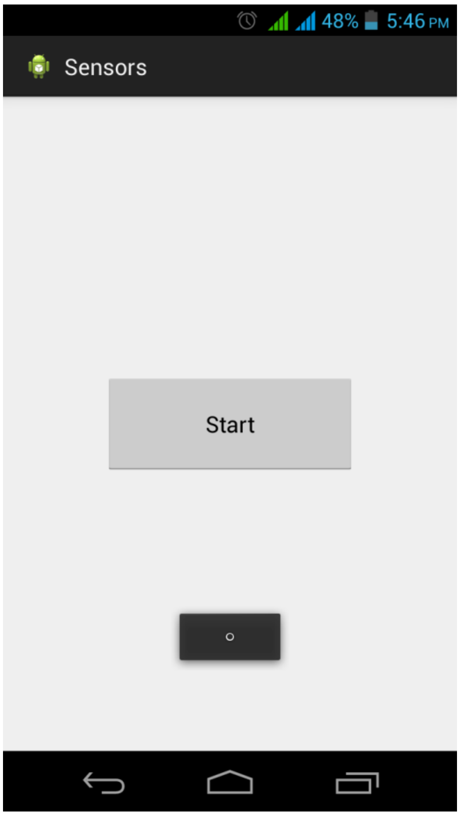

Figure 12: Implementation of Characters Consists of only Horizontal and Vertical Stroke

CHAPTER 5

Proposals

This chapter we will describe our future work plan and details about our proposals.

5.1 Proposal 1: Detecting LURB and RULB Tilted Strokes

We have discussed about LURB and RULB tilted strokes earlier. If we can detect these two strokes then we can detect all the Bangla Segmented Characters. Detecting these two strokes open the whole way to detecting all the segmented Bangla digits.

What we will do: We will use the rational distance comparison of a stroke to detect these two strokes. We will also use data from gyroscope sensor readings for fine tuning during detecting these strokes.

5.2 Proposal 2: Resolving Ambiguity Issue of Characters

Ambiguous characters are not possible to be detected by this system. To develop a full-fledged system we need to resolve the issue.

What we will do: We will use the rational distance comparison of the strokes of two characters to distinguish between them. We will also use data from gyroscope sensor readings for fine tuning during detecting these strokes.

5.3 Proposal 3: Detecting Hand Lifting During Character Writing

We have encountered a problem of hand lifting during the implementation of detecting the strokes. We need to resolve this issue to build a full functional system of Bangla Air Writing. Without this we will not be able to implement the things we are trying to do.

What we will do: We will use data from gyroscope sensor readings for resolving the issue. As we know gyroscope can measure angular acceleration so we can resolve the issue easily using gyroscope data.

5.4 Proposal 4: Implementing Air Writing For Segmented Bangla Alphabets

We have encountered couple of issues implementing Segmented Bangla digits. All the issues are also valid for the Bangla alphabets. So after implementing full functional system of detecting Segmented Bangla digits will also apply our knowledge to detect Segmented Bangla alphabets.

What we will do: We will use the knowledge we have gained and will gain to implement the detection of Segmented Bangla alphabets.

5.5 Proposal 5: Implementing Bangla Air Writing in 3-D plane

Our current system only works in 2-D plane. We will make it work for 3-D plane. If we can make the system work for 3-D plane then the system can be used in delivering class lectures.
The will just need connect this android system to PC and the pc will show the characters written by lectures in the air.

What we will do: We will make use of gyroscope readings to make this happen. Integrating the gyroscope data will make the system work in 3-D plane.

References

[1] Md. Abul Kalam Azad, Rezwana Sharmeen, Shabbir Ahmad and S. M. Kamruzzaman, “A Unique 10 Segment Display for Bengali Numerals” available from: arxiv.org/pdf/1009.4590

[2] Salahuddin Mohammad Masum, Swapon Chandra Dash, Sarwar Morshedul Haque, Kazi Faisal Kabir, “SEGMENTED DISPLAY SYSTEM FOR BENGALI ALPHABET” available from: research.daffodilvarsity.edu.bd/

[3] Sandip Agrawal, Ionut Constandache, Shravan Gaonkar, Romit Roy Choudhury, Kevin Caves, Frank DeRuyter, “Using Mobile Phones to Write in Air” available from: www.ee.duke.edu/.../P3.pdf

[4] N. Pranathi, Syed Musthak Ahmed, “Tri-Axis Motion Detection using MEMS for Unwired Mouse Navigation System in the Future Generation Machines” international Journal of Advanced Research in Computer and Communication Engineering Vol. 2, Issue 9, September 2013

[5] Ray Lockton, “Hand Gesture Recognition Using Computer Vision” available from: research.microsoft.com/.../project

[6] Jue Wang, Deepak Vasisht, and Dina Katabi, “RF-IDraw: Virtual Touch Screen in the Air Using RF Signals” available from: people.csail.mit.edu/.../RF-IDra...

[7] Christoph Amma, “Air Writing Recognition Using Wearable Motion Sensors” Karlsruhe, den 19. August 2009, available from: csl.anthropomatik.kit.edu/1627….

[8] Hyuk-Jeen Shu, “Air Writing Motion Sensing Input for Portable Devices” available from: google.com/patents/US7667686

[9] Christoph Amma, Marcus Georgi, Tanja Schultz, “Air-Writing: Hands-free Mobile Text Input by Spotting and Continuous Recognition of 3-D Space Handwriting with Inertial Sensors” available from: dl.acm.org/citation.cfm?id=1785465

[10] Rene Mayrhofer, Alexander Sommer, Sinan Saral. "Air- Writing: A Platform for Scalable, Privacy-Preserving, Spatial Group Messaging" available from: dl.acm.org/citation.cfm?id=1967517

[11] Zhichao Ye, Xin Zhang, Lianwen Jin, Ziyong Feng, Shaojie Xu, "Finger-Writing in the Air System Using Kinect Sensor" Multimedia, IEEE, 2013, available from: vc.cs.nthu.edu.tw/…./[PDF]A%2…

[12] Mehdi, S.A., Khan, Y.N., "Sign Language Recognition Using Sensor Gloves" available from: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=1201884&isnumber=27060&url=http%3A%2F%2Fieeexplore.ieee.org%2Fstamp%2Fstamp.jsp%3Ftp%3D%26arnumber%3D1201884%26isnumber%3D27060

[13] Thomas Deselaers, Daniel Keysers, Jan Hosang, Henry A. Rowley, "GyroPen: Gyroscopes for Pen-Input with Mobile Phones" available from: research.google.com/…/43127.p…

[14] Alexander Schick, Daniel Morlock, Christoph Amma, Tanja Schultz, Rainer Stiefelhagen, "Vision-Based Handwriting Recognition for Unrestricted Text Input in Mid-Air" available from: isob.fraunhofer.de/servlet/is/6635/sp202-schik.pdf

Web Links

[1] Accelerometer – Wikipedia, the free encyclopedia
http://en.wikipedia.org/wiki/Accelerometer

[2] Gyroscope – Wikipedia, the free encyclopedia
http://en.wikipedia.org/wiki/Gyroscope

[3] Motion Sensors | Android Developers
http://developer.android.com/guide/topics/sensors/sensors_motion.html