# 

**Speech Controlled Robotic Arm**

ME735-Computer Graphics & Product Modelling

*Guide: Prof. S.S.Pande*

**Akash Kishore 130010010**

**Vivek Pandit 130010009**

**Prajval Somani 130040003**

**Jivraj Karwa 13D100017**

****

**Department of Mechanical Engineering**

**INDIAN INSTITUTE OF TECHNOLOGY, BOMBAY**

**November, 2016**

# **Table of contents**

**Introduction 3**

[**Objective**](#_1gf52qwglol5) **4**

[**Algorithm**](#_xup008ge37hh) **5**

[**Results**](#_h1vs5d2wywen) **7**

[**Future Work**](#_6grfnxk4ja3e) **10**

[**References**](#_nmifiwtv7fb6) **11**

# 

# 

# 

# **Introduction**

Speech recognition (SR) is the inter-disciplinary sub-field of computational linguistics which incorporates knowledge and research in the linguistics, computer science, and electrical engineering fields to develop methodologies and technologies that enables the recognition and translation of spoken language into text by computers and computerized devices such as those categorized as smart technologies and robotics. It is also known as "automatic speech recognition" (ASR), "computer speech recognition", or just "speech to text" (STT).

Some SR systems use "training" (also called "enrollment") where an individual speaker reads text or isolated vocabulary into the system. The system analyzes the person's specific voice and uses it to fine-tune the recognition of that person's speech, resulting in increased accuracy. Systems that do not use training are called "speaker independent" systems. Systems that use training are called "speaker dependent".

# Robotic-arm-2400px.png

# 

**Objective**

To make an application on Windows which takes user voice as input and converts it into a text file which is mapped to its appropriate commands which further acts as input for the robotic arm.

In this project we tried to operate the robotic arm present in lab using speech. For this purpose we have used inbuilt APIs of windows. The codes were written using C-Sharp in Visual Studio 2015.



# **Algorithm**

The aim of the project is to control a robotic arm through speech. For this we followed the following algorithm.

Untitled Diagram.png

### **Code Structure**

**Form1.cs** - Contains the outermost layer of the GUI- contains 1 image-button and four normal popup buttons.

**Form2.cs** - When the image-button is clicked, form 2 is launched. This is the main window application which does the speech recognition. It contain an audiobar and a textbox which prints all the words the system hears, after it has been filtered appropriately by the code.

**Form3.cs** - This c# indow application file enables the user to modify the vocabulary which the code understands.

**Word.cs** - defines the list of word ( or vocabulary) which the application understands.

**Program.cs** - defines the order in which the application will run. Form1 i launched when the application is started.

Detailed explanation of the code :

Important Libraries used - System.speech

System.IO

System.Diagnostics

System.Net.Mail

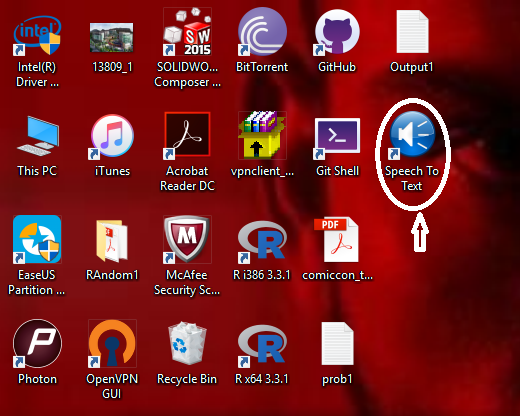
Important functions defined :

* **loadGrammarAndCommands()** : this function defines the grammar(vocabulary) of the application. This function takes as input example.txt file which contains the list of commands to recognise and corresponding actions to be taken. The output file thus generated is thus recorded in output.txt
* **engine\_AudioLevelUpdated()** : this function updates the progress bar level as per the audio level recorded by the microphone
* **engine\_SpeechRecognized()** : whenever speech is correctly recognised, this function is called.The recognised text is written into the textbox.
* **createSpeechEngine()** : this function recognises which is preffered language ( here US- English )
* **Form2\_FormClosing()** : when form2 is closed speechRecognitionEngine is disposed.
* **myButtonObject : UserControl** : makes a round image-button during runtime
* **btnEdit\_Click()** : new Form2 is created when ‘Edit Grammar’ is clicked
* **button2\_Click()** : Instruction.txt is launched when ‘Instructions button’ is clicked

# **Results**

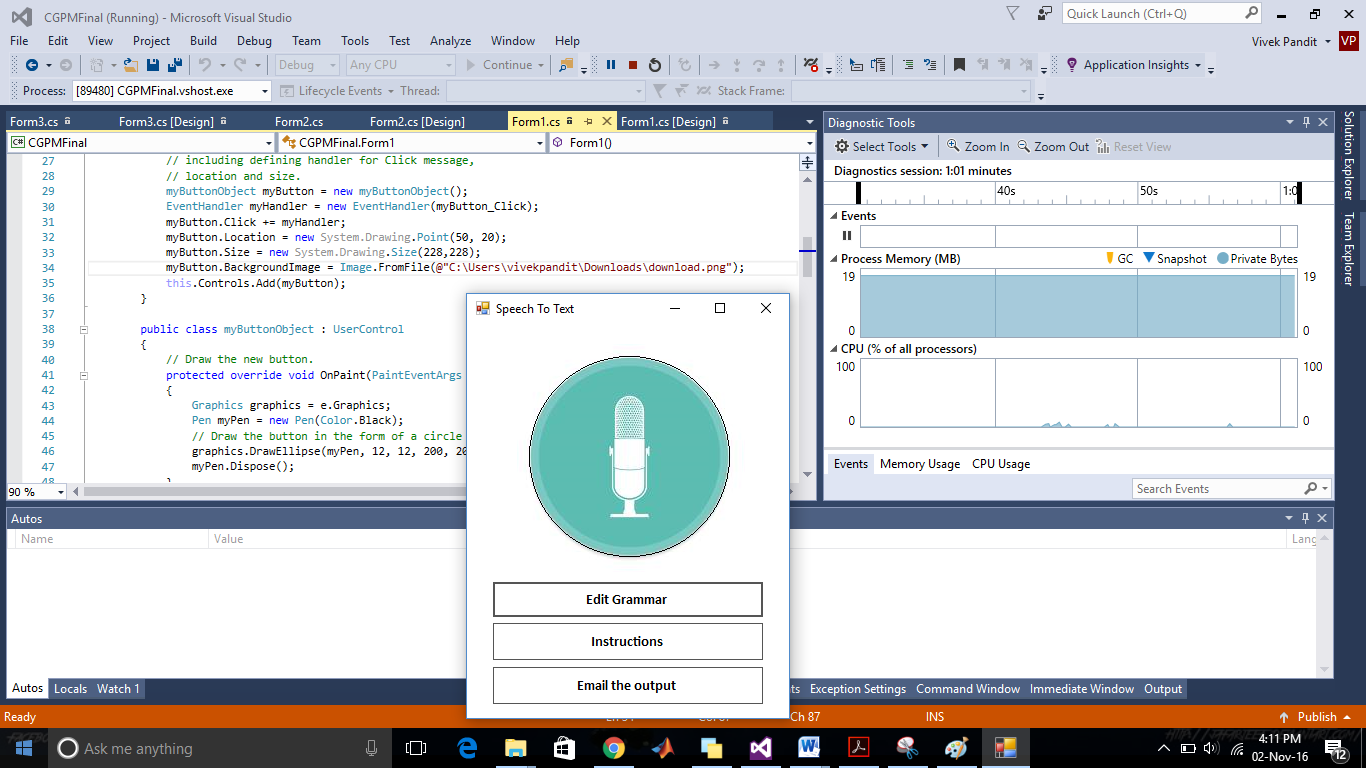
The Robotic Arm takes in commands ,according to a specified syntax, from a text file. Our application takes in user input I.e user's voice and converts it to the required syntax before writing it to the text file.

The following results are obtained.

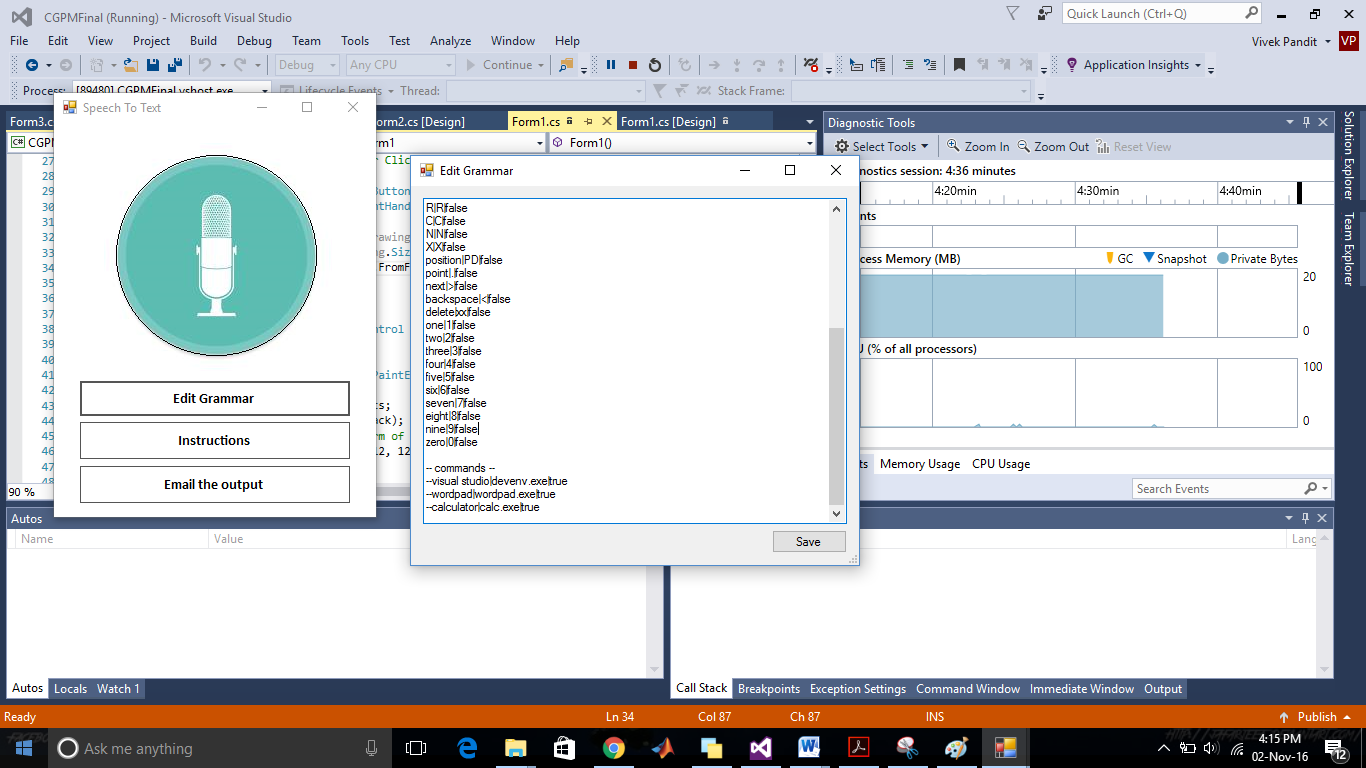


**Fig.1.1 Desktop App**

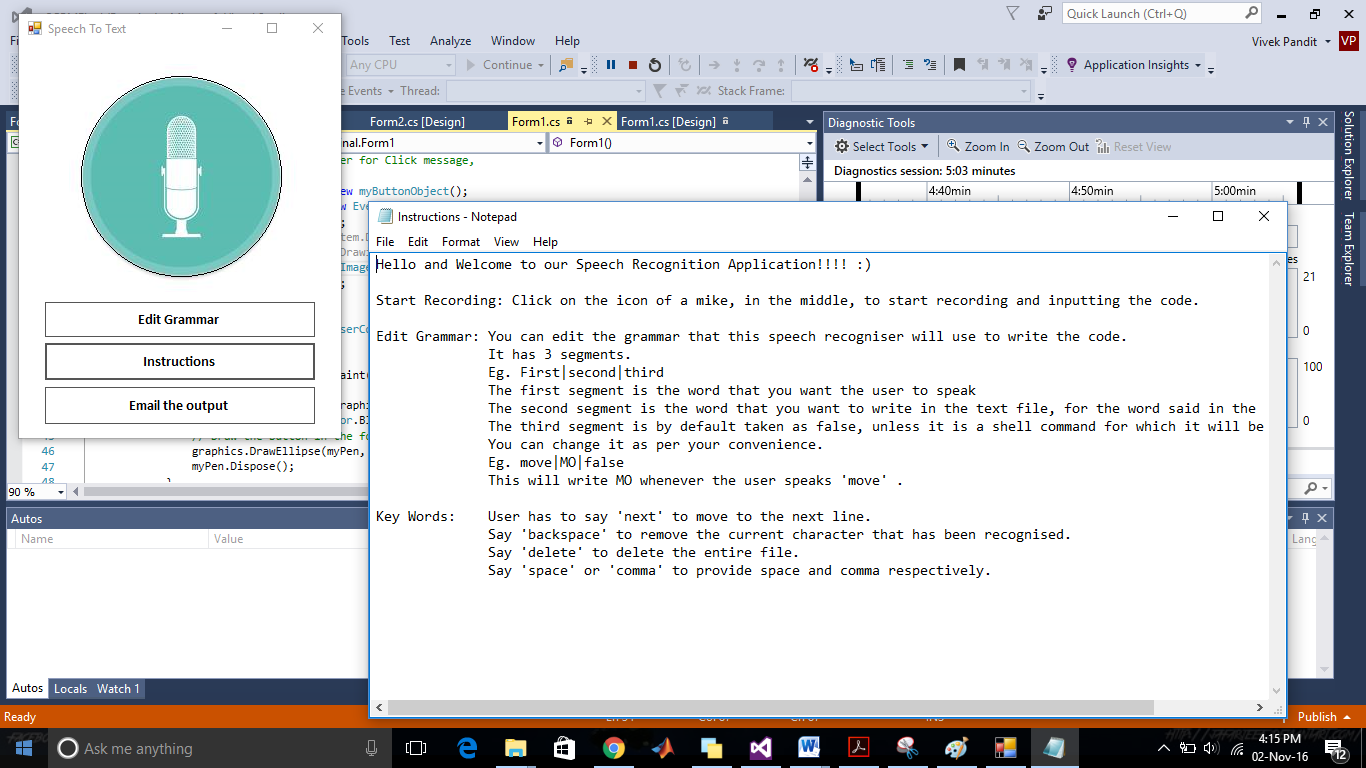
Desktop shortcut for the application - click this to launch the application



**Fig.1.2 - Initial Window**



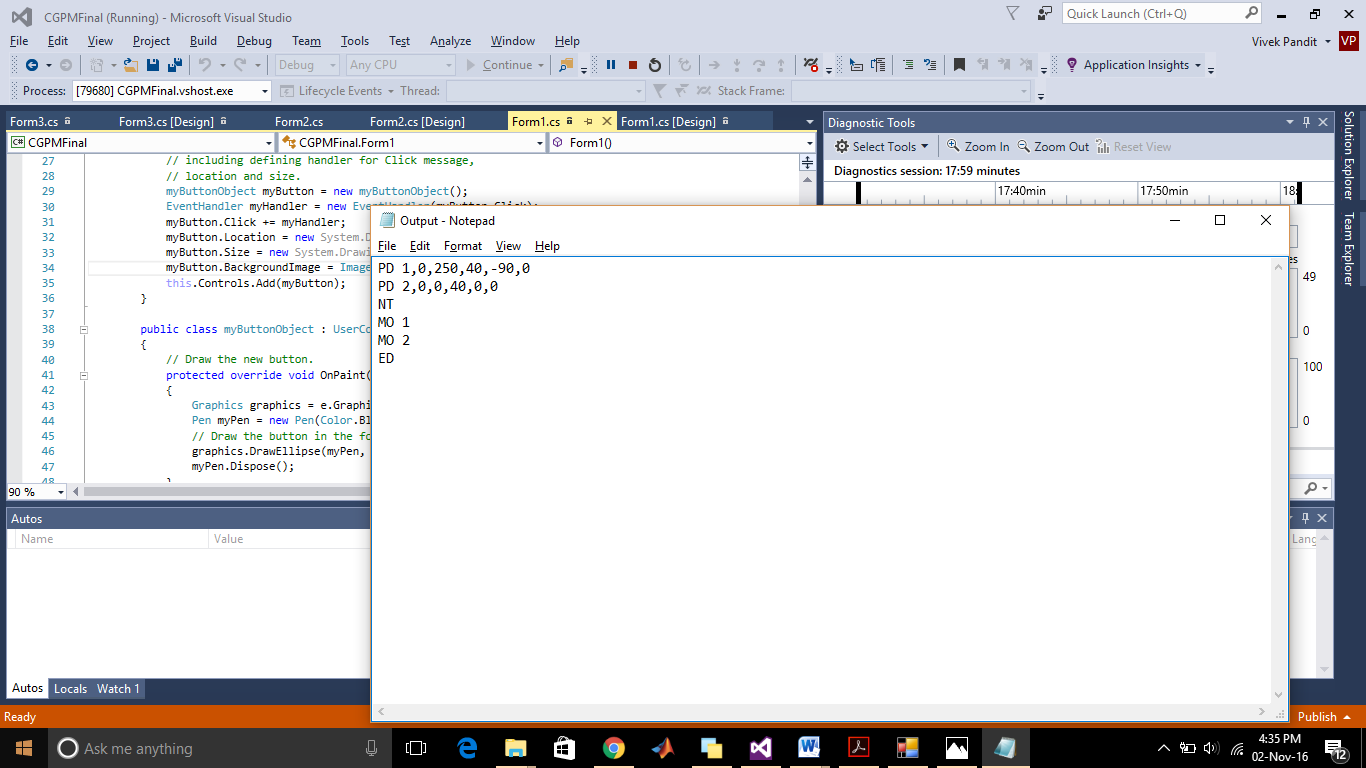
**Fig.1.3 - Edit Grammar**



**Fig. 1.4-Instructions.txt**

# 

# **Fig. 1.5 Recording**



**Fig. 1.6 Output text file**

# **Future Work**

# Currently the user has to speak an entire code before inputting it to the arm. It would be better and more intuitive if the arm could support line by line transfer of code I.e able to interpret input streams of code line by line and move accordingly. Alongwith this the integration of the speech recognition software with the arm will make the interface much easier.

# In case of numerical values, currently our application supports speaking the digits individually. Eg. For writing 254, the user has to speak two,five and four. It can be improved such that user has to speak two fifty four to write 254 in the text file. This will be much more user friendly and will save time.

Also we can make writing the code specific to the arm, i.e appropriately put in comma and space according to the syntax so that the user doesn’t have to speak trivial words like comma, space, etc. everytime.

# 

# **References**

1. **Visual Studio 2015 Documentation**

<https://msdn.microsoft.com/query/dev14.query?appId=Dev14IDEF1&l=en-US&k=k(MSDNSTART)&rd=true>

**2) CAM Lab Robotic Arm Manual**

Contains syntax of the code for the robotic arm