Speech Recognition applied in Games

Yue Wang
College of Computer Science and Technology
Harbin University of Science and Technology
Harbin, China
meyia @163.com

***Abstract*—with the development of the Pattern Recognition, the technology of Speech Recognition has played an increasingly significant role especially while it is applied into use. Therefore the theory of the application of the Speech Recognition in games is raised, aimed at making it more convenient for the players to interact with the computers without using the keyboard or mouse. Moreover, compared with other methods it has more advantages in both efficiency and portability.**

Keywords-Pattern Recognition; Speech Recognition; Artificial Intellegence; real-time interactioin.

I. INTRODUCTION

Speech Recognition is a way for the human and machines to exchange information by transforming human's voice into the language which the machines can recognition and execute in order to give the result back to human by a form of information expression. It is one of the five main fields of the Pattern Recognition, which is used to make strategic decisions for the Artificial Intelligence Systems [1]. Speech Recognition technology can be traced back to the 20th century 50s AT & T Bell Laboratories Audry system which is the first ten English numbers recognition of the human voice [2]. Followed by this, the substantial progress was made during the end of the 1960s and the beginning of the 1970s, because of the possibility of both hardware and software condition. Moreover, the raising of the technology of linear predictive coding and dynamic time warping make it more efficient to solve the problems of Speech feature extraction and the matching of unequal length. At the same period, the Speech Recognition is mainly based on the principle of template matching, and raised vector quantization and hidden Markov model. Till the 1980s, the Sphinx system invented by Carnegie Mellon University made a great breakthrough of high performance of non-specific people and large vocabulary continuous Speech Recognition system [3]. University of Cambridge and the Philippine company are also studying English Pu Speech Recognition algorithm applied in other European languages in possibilities and effects [4, 5]. Till now, Speech Recognition has gradually been put into use in many fields. In this essay the Speech Recognition in games will be discussed mainly in three aspects: Characteristics, principle of Microsoft's Speech SDK, and the method for its application in games.

II. CHARACTERISTICS OF SPEECH RECOGNITION

A. Classifications Methods

(1) According to limits of the input speech, Speech Recognition can be classified into three categories.

a) Specific Speech Recognition: To recognize the only specific man's voice.

b) Non-Specific Speech Recognition: To recognize any kinds of man's speech, usually requires a lot of different people's voice recognition system to found the database.

c) Specific-Group identification: To identify everyone's speech in a specific group. It needs to let the specific group of people to practice.

(2) According to the utterance of the speakers, Speech Recognition can be classified as below:

a) Isolated words Speech Recognition: it requires a pause after each word.

b) Linking words Speech Recognition: it asks clear pronunciation of each word, and it allows the appearance of the linking words.

c) Continuous Speech Recognition: it needs the natural frequent and continuous speech input, and the appearance of large amount of continuous words and inflexion are allowed.

(3) According to the amount of the vocabulary, Speech Recognition can be classified as below:

a) Small Vocabulary Speech Recognition: it can recognize small amount of words normally may be dozens of words.

b) Medium Vocabulary Speech Recognition: it can recognize hundreds of thousands words.

c) Large Vocabulary Speech Recognition: it can recognize thousands of tens thousands of words.

As with the fast development of the capacity of both computers and the digital processing power as well as the improvement of accuracy of the Speech systems, Recognition systems are classified according to the size of vocabulary is also constantly changing.

B. Methods of Speech Recognition

Generally speaking, the methods of Speech Recognition which is classified as phonetics and acoustics based, template matching, and neural network. To the first method, there are a finite number of different speech primitives which can be distinguished by speech signal's frequency and time domain property that can be found in the common language. The

Yue Wang: *major research direction in graphics, mechanical configuration, and embedded GUI.*, Pattern Recognition, etc.

978-1-4244-5392-4/10/$26.00 ©2010 IEEE

second method which has been in the practical stage is realized in four steps, that is feature extraction, template training, template categories, and judgment. There're normally three technologies of matching, Dynamic Time Warping (DTW), Hidden Markov Method (HMM), and Vector Quantization (VQ).The third method, to its essence, is a self-adaptive nonlinear dynamic system which imitates the principle of human nerves' activity. Moreover, it has the characteristics of self-adaptability, parallelism, robustness, fault tolerance and learning as well as an extremely strong classifying ability, and also the capacity of mapping the input and output. But it's still under-development as an ideal goal.

The second method, template matching is adopted in the theory of the Speech Recognition applied in games. Because of its speech training and matching steps makes it more efficient and accurate especially to different people's voices. This point will be discussed in the next chapter.

C. Composition of Speech Recognition Systems

The basic principle of the Speech Recognition is to process the input speech, after which compare the speech with the voice template library, therefore give the result. See in Figure1 below.

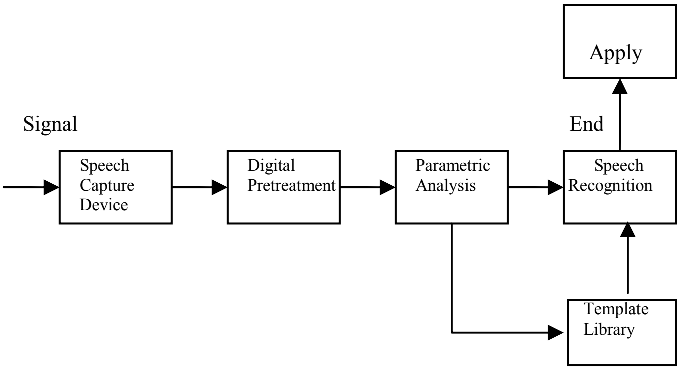

Figure1. Principle of Speech Recognition

III. WORKING PRINCIPLE OF SPEECH SDK

Microsoft's Speech Software Development Kit (SDK), is based on the COM platform, which contains the Server Application Program Interface (SAPI), Microsoft Continuous Speech Recognition (MCSR), Text-To-Speech (TTS), etc. The speech framework is used to provide better cooperation between application programs and speech engine by using SAPI. In addition, the speech framework has a variety of interfaces to realize distinguished functions of speech. In the application the SDK and TTS will be used together, therefore, the instructions of the two engines for the system should be introduced.

A. Speech Recognition Engine Interface

There are four interfaces contained.

- IspRecoContext: Interface for the speech to recognize the context which is the main interface of the program, is used to send and receive the context messages of the Speech Recognition, and to create objects of grammar rules.
- IspReeognizer: Interface of speech recognition engine, is used to create an instance.
- IspRecoGrammar: Interface of the rules of grammar, it gives definition of the concrete content that the recognition engine needs. It creates, loads, and actives grammar rules.
- IspVoice: Interface of the result of recognition, it is used to get all the results of literal, grammar rules, etc.

B. Speech Synthesis Engine Interface.

There are three steps it works in.

- Create the object of SpVoice.
- Set the speech attribute by using the function of object member, if not set this, it will be dealt in default.
- Call the speak function of the object member in order to assign the two parameters it needs. The first is read-need text while assigned, and the second is the pronunciation way while assigned.

C. Working Principle

While in the whole project, the COM component is responsible for a series of initial work of the speech; the speech is recognized by the recognition engine; the speech synthesis is worked out by the synthesis engine; and the programmer can therefore concentrate on the application by calling the SAPI so that the Speech Recognition is realized. The message-driven mechanism is described as below. The flowchart of the Speech Recognition is shown in figure2.

- First, initialize the COM platform.
- Second, giving the definition of each speech interface object (In specific order), making the recognition engine in the working state by setting the recognition grammar and message.
- Third, after some are recognized following the grammar rules, ISpRecoContext object send messages to the applied programs, therefore message recognition reaction mechanism is called. In the function of message matching, get the result by the Interface of ISpPhrase, loop until the stop of the grammar rules.
- Finally, while the application program is quitted, uninstall the COM platform.

IV. APPLICATION IN GAMES

After all the preparation is done for the application, the program written by C# will describe the real-time interaction games between the computer and the user. After the vibration transform from the user's throat to the computer by the medium of microphone, the COM platform receives the input instruction and under the recognition engine's effort, if the speech recognition engine find the speech can be recognized and if it matches the library whose database is founded by the user's speech practice training, then the speech is translated into ASCII code which the computer can execute the instruction easily. Next, the computer returns the executed results to the Synthesis engine if necessary or just shows the result on the screen. Finally the user gets the result he wants.

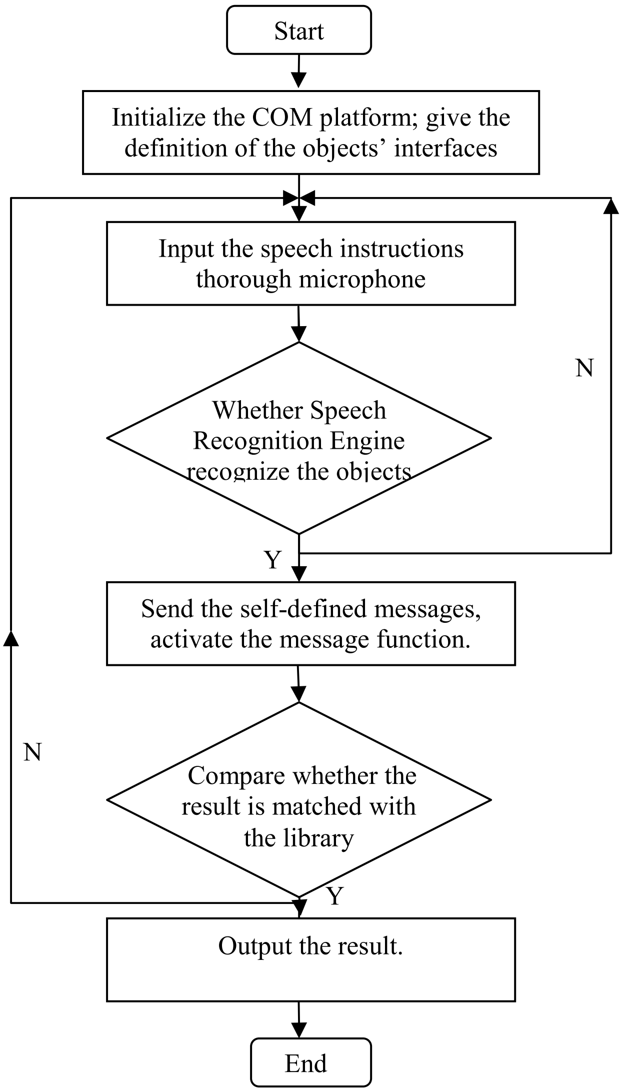
Start
Initialize the COM platform; give the definition of the objects' interfaces
Input the speech instructions thorough microphone
Whether Speech Recognition Engine recognize the objects
N
Y
Send the self-defined messages, activate the message function.
Compare whether the result is matched with the library
N
Y
Output the result.
End

Figure2. The Flowchart of Speech Recognition

A. *Real-Time Interaction*

Under the theory of template matching method, the Microsoft's SDK easily achieved the training for the speakers, and the dynamic database which will be put into the library for the supervisor's call. And with the fast speed of the speech recognition engine, the user can interact with the machine in no time, which means more flexibility and less waiting time. Application in games

One of the modules of the games is the Tetris, and some quotations are put into the project so as to make the program faster and accurately. Some important code is shown in the below.

```
        private void lReC_TextChanged(object sender,
EventArgs e){if (block != null && this.paused == false
&& !this.failed)
            {
             if (lReC.Text.ToString().Equals("Left"))
                { if (block.Move(0))
                    { block.EraseLast();}}
             else if
(lReC.Text.ToString().Equals("Down"))
                {if (!block.Move(2))
                    {  this.FixAndCreate();}
                    else
                    {   block.EraseLast(); }}
            else if
(lReC.Text.ToString().Equals("rotate"))
                { if (block.Rotate())
                    {block.EraseLast();}}
            else if
      ……}
    private void btHelp_Speech(object sender, EventArgs
e) {
           button4_Speech(null, null);
            MessageBox.Show("Instructions: \n\"Left\",
\"Right\", \"Down\", \"Rotate\", \"Bottom\", \"Left
End\", \"Right End\", \"Start\", \"Pause\",
\"Continue\", \"Return\", \"Stop\"", "Help",
MessageBoxButtons.OK);
            button4_Speech(null, null);
        }……
```

Hereby, some quotations are made as below in order to draw the Window and call the system parameters and the SDK's functions.

SpeechCommon package;.

SpeechLib;

System;

System.Core;

System.Data;

System.Data.SetExtensions;

System.Deployment;

System.Drawing;

System.Windows.Forms;

System.Xml;

System.Xml.Ling;

Tetris;

……

Another game used to illustrate the application is the Quick Memory. Some important code is shown in the below.

```
private void lbText_TextChanged(object sender, EventArgs
e)
        {
            if (lbText.Text.ToString().Equals("开始")
&& !gameBegin)
{ spR.CloseRec(); Start();}
            if (index > level)
{lbStatus.Text = "说确定，以检测您的结果";}
            if(lbText.Text.ToString().Equals("帮助"))
{ pause = true; Help();pause = false;}
            if (lbText.Text.ToString().Equals("返回"))
{ btReturn_Click(null, null);}
            if (lbText.Text.ToString().Equals("结束"))
{  btEnd_Click(null, null);}
            if (userBegin)
            {
if (!pause)
              {if (lbText.Text.ToString().Equals("取消"))
                    {
                        index--;
                        GetPb();
                        pb.Image = null;
                    }
                    else if (index <= level)
                    {
                        LoadUserPicture();
                    }
                    if (index > level &&
lbText.Text.ToString().Equals("确定"))
                    {
                        CheckResault();
                        userBegin = false;}}}
            lbText.Text = "";
        }
        private void Start()
        {
            lbStatus.Text = "游戏开始";
            lbLevel.Text = "级数：" + (level -
2).ToString();
            lbScore.Text = "分数" + score.ToString();
            qOrder = "";
            userOrder = "";
            LoadOrderPicture();

            timer1.Interval = 1000 + level * 100;
            timer1.Tick += new EventHandler(ClearU);
            timer1.Enabled = true;
            lbStatus.Text = "请说出您所看到的箭头方向，
说取消以取消上一块";
        }
```

```
private void LoadUserPicture()
{
    string pbName = null;
    GetPb();
    if (lbText.Text.ToString().Equals("上"))
    {
        pbName = "upArrow";
        userOrder += "上";
        index++;
    }
    else if (lbText.Text.ToString().Equals("下"))
    {
        pbName = "downArrow";
        userOrder += "下";
        index++;
    }
    else if (lbText.Text.ToString().Equals("左"))
    {
        pbName = "leftArrow";
        userOrder += "左";
        index++;
    }
 pb.Load("picture\\" + pbName + ".bmp");
   }
```

V. CONCLUSION

In this essay, the classifications and methods of the Speech Recognition are discussed, and the characters of each categories and methods are described. Through the technology of using the Speech Recognition method applied in games, the man-machine conversation is finally achieved. What's more, this technology has a considerable probability in both the computers and the embedded devices with good capability by using the SDK. So with a well-organized game package after the applications are encapsulated, prospects of the future markets must be satisfied.

REFERENCES

[1] Sergio Theodoridis, Konstantinos Koutroumbas, Pattern Recognition Forth Edition[M], ISBN: 978-7-121-10278-3, Beijing: 2010

[2] D. Karis, K. M. Dobroth, Automating services with speech recognition over the public switched telephone network: human-factors considerations, IEEE Journal on Selected Areas in Communications, Vol. 9, No. 4, 1991, 574-585.

[3] X. D. Huang et al. The SPHINX- II speech recognition system: an overview, Computer Speech and Language, No. 2, 1993, 137-148.

[4] S.J. Young A review of large-vocabulary continous-speech recognition, IEEE Signal Processing Magazine, Vol.13, No.5, 1996, 45-57.

[5] S.J. Young et al, Miltilingual large vocbulary speech recognition the European SQALE project, Computer Speech and Langauage. Vol.14, No.11, 1997, 73-89.