

用 Delphi 实现基于 SAPI 的电脑语音控制应用程序

Using Delphi Implemented an Voice Controlled Application Based on SAPI

(湖北师范学院) 童 强

Tong, Qiang

摘要: 介绍了语音功能的基本概念, 讨论了开发具有语音控制功能应用程序的条件和步骤。并用 Delphi 实现了一个基于微软 SAPI 的语音控制应用程序, 此程序可以用方便的扩展以实现更多语音控制功能。

关键字: 语音识别; SAPI; XML; 自动化; 对象

中图分类号: TP391.42 文献标识码: A

Abstract: This paper introduces the base concept about speech function, discusses the condition and the process of designing a voice Controlled application, and Using Delphi Implemented a voice controlled application based on MS SAPI, This application can implemented more voice controlled functions by easy program extension.

Key word: Speech Recognition (SR); SAPI ; XML; Automation; Object

技
术
创
新

1 前言

实现人类和电脑的语音交流一直是人们追求的目标, 这要求开发具有语音功能的应用程序。电脑语音功能主要包括两大部分: 语音合成 (Text-To-Speech Synthesis, TTS) 和语音识别 (Speech Recognition- SR)。前者让电脑能说; 后者让电脑能够获取并理解使用者的声音, 从而使用户能象使用鼠标和键盘一样用语音来控制电脑工作。

语音识别通常又可以分为两种工作方式, 一种就是命令控制 (command and control) 方式, 这个方式下语音识别引擎可以识别简短的语音命令, 以便执行相应的程序; 另外一种连续听写方式, 这个模式下语音识别引擎要识别连续的语音, 这种功能实现起来比语音控制更复杂, 因为语音听写过程中需要对上下文以及相同相似发音的词语进行分析、作出判断, 而在命令控制语音方式中不需要作上下文分析。

语音识别和语音合成的基础技术是非常复杂的, 世界各地的一些研究所和高校的实验室已经对语音合成和语音识别技术进行了多年的研究, 微软等大公司每年也在这方面投入大量的研发资金。目前一些成熟的产品正在逐步走向实用, 比如微软已经在 OFFICE 套件中加入了语音功能。

2 开发语音功能程序的条件

虽然语音技术理论研究涉及到人机交互、用户界面、自然语言、人工智能等多个科学领域, 应用程序开发人员难以从底层进行研究开发。令人欣慰的是一些

公司、研究所研究的成果已经走向实际应用, 并且为广大开发人员提供了实用的开发工具和平台, 其中影响较大的有微软的 Speech 语音开发包系列以及 IBM 的 ViaVoice 语音开发包, 利用这样的开发平台可以比较容易的开发具有语音功能的应用程序。本文采用的是微软的 speech SDK 5.1, 这是一个免费的开发包, 并且可以用它开发具有中文语音功能的软件。

2.1 SAPI 5.1 简介

SPEECH SDK5.1 是微软公司免费提供的语音应用开发工具包, 这个 SDK 中包含了语音应用设计接口 (SAPI)、微软的连续语音识别引擎 (MCSR) 以及微软的语音合成 (TTS) 引擎等等。目前的 5.1 版本一共可以支持 3 种语言的识别 (英语, 汉语和日语) 以及 2 种语言的合成 (英语和汉语)。SAPI 中还包括对于低层控制和高度适应性的直接语音管理、训练向导、事件、语法编译、资源、语音识别 (SR) 管理以及 TTS 管理等强大的设计接口。其结构如图 (1):

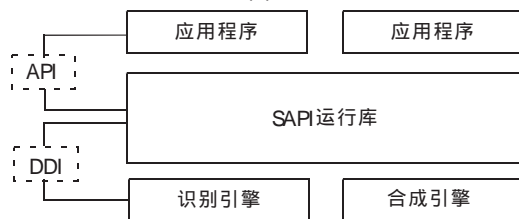


图 1

语音引擎则通过 DDI 层 (设备驱动接口) 和 SAPI (SpeechAPI) 进行交互, 应用程序通过 API 层和 SAPI 通信。通过使用这些 API, 用户可以快速开发在语音识别或语音合成方面应用程序。

到微软 Speech 站点下载安装 SAPI5.1 后就可以进行开发了。SAPI 5.1 的 SAPI 调用有两种方式: 一种

童强: 讲师

湖北师范学院 03 院研究基金启动项目

是低级的 COM 接口,一种是高级的自动化接口。可以用 VC, VB, c# 等多种语音开发 SAPI 的语音功能。本文采用 Bland 公司的深受广大开发人员喜爱的 Delphi 作为集成开发工具, Delphi 应用程序中增加语音功能并不很复杂。虽然 Delphi 7.0 不能实现 COM 接口转换,但是可以通过自动化接口调用 SAPI 的语音功能,并且调用方法很简捷。

2.2 在 Delphi 应用程序中应用 SAPI 的准备

SAPI 5.1 在系统注册类库中包含了重要的接口、类型、常量等,开发者可以通过前期或后期自动化绑定技术调用 SAPI 5.1 提供的函数功能。本文采用的是前期自动化绑定方法,这需要在 Delphi 中先导入 SAPI 5.1 类库(这里使用的是 Delphi 7.0)。其方法如下:

首先选择 Project 的 Import Type Library 菜单,在弹出的类库列表中选择名为 Microsoft Speech Object Library (Version 5.1)的类库,同时勾选 Generate Component Wrapper 复选框,然后点击 Install 按钮导入类库,这样可以把每个导入的自动化对象以控件形式默认添加 Delphi 的 Active 控件选项卡上,也可以单独建立一个 SAPI 控件选项卡。添加成功后,在控件选项卡上会增加 19 个 SAPI 相关控件。利用这些控件就可以方便地在程序中加入语音对象来开发具有语音功能的软件了。

相对基于复杂的 COM 接口的调用开发,自动化接口的使用要方便很多。例如在应用程序中加入 SpVoice 自动化对象调用它的 Speak 方法就可以轻松实现朗读。下面这段代码实现了点击一个按钮就朗读一个文本框中的文本内容的功能。

```
procedure TForm1.Button1Click(Sender: TObject);
begin
  SpVoice1.Speak(Memo1.Text,SVSFDefault) //朗读文本
end;
```

3 语音控制应用程序开发步骤

语音控制程序就是用语音控制代替键盘或者鼠标实现对软件或者硬件的控制。主要思路是利用语音识别的命令控制(command and control)方式识别语音命令,再做相应控制。

要实现应用程序的语音控制功能,就要将语音识别的基本工作融合到电脑应用程序的运行机制中去,具体到 windows 操作系统,就是要把语音识别引擎的工作信息及事件处理加入到 windows 的消息机制中,从而使应用程序中能够接受和响应语音识别引擎识别工作中产生的语音识别消息,对应特定的消息就可以执行对应的控制程序。具体应该分这么几步:

(a) 在应用程序中加入语音识别自动化对象,以便程序能通过 SAPI 调用语音识别引擎;

(b) 编写辞书文法文件(XML 文件),定义要识别

的语音命令及其对应属性;

(c) 然后在应用程序初始化时载入语音识别功能对象(启用语音识别引擎)以及辞书文法文件(使语音识别引擎识别程序感兴趣的语音命令);

(d) 在语音识别对象的语音消息响应函数中查对识别信息,根据辞书文法文件中定义的语音命令,执行对应的控制程序。

根据实际需要,对(b),(d)两步做修改就可以在应用程序中实现多种语音控制功能。

4 语音控制应用程序的实现

下面用一个简单的语音控制应用程序来说明语音控制应用程序的实现过程,该程序可以实现语音控制界面颜色的变化,以及语音控制关闭程序。

(1) 添加语音识别自动化对象

如前所述 Delphi 中导入 SAPI 5.1 类库后就有 19 个自动化对象可以选用,要调用语音识别引擎添加 SpSharedRecoContext 对象即可,该对象有 18 个事件响应函数可以使用。

(2) 建立辞书文法文件

辞书语法文件是一种 XML 格式的文件,文件中主要规定了要识别的语言种类(是中文还是英文或者其他语种),以及定义了识别的语音命令对应的规则。在识别引擎命令控制状态下语音识别引擎不必对所有语音输入都准确识别,只需要按照辞书文法文件的定义的规则准确识别有限的命令即可。下面的语法文件定义了使用中文语音识别几种颜色的语法规则。

```
<GRAMMAR LANGID="804">
  <DEFINE>
    <ID NAME="RID_start" VAL="1"/>
    <ID NAME="PID_colour" VAL="2"/>
    <ID NAME="PID_colourvalue" VAL="3"/>
  </DEFINE>
  <RULE NAME="start" ID="RID_start" TOPLEVEL="ACTIVE">
    <RULEREFF NAME="colour" PROPNAME="colour" PROPID="PID_colour" />
  </RULE>
  <RULE NAME="colour">
    <L PROPNAME="colourvalue" PROPID="PID_colourvalue">
      <P VAL="1">红色</P>
      <P VAL="2">蓝色</P>
      <P VAL="3">绿色</P>
      <p VAL="5">退出</p>
    </L>
  </RULE>
</GRAMMAR>
```

这个语法辞书文件根据点定义了语音识别的语

种是中文(LANGID 为 804, 英式英语是 809, 美式英语是 409)。规则“colour”定义 3 种颜色, 其对应的属性值分别为 1、2、3, 另外定义了一个“退出”命令, 其属性值是 5。在应用程序中可以根据识别后返回信息中的属性值就可以执行对应的控制程序。

(3) 初始化程序

辞书语法 XML 文件要在应用程序初始化的时候在 OnCreate 事件响应函数中装载并激活该文件的语法规则, 代码如下所示:

```
procedure TfrmCommandAndControl.FormCreate
(Sender: TObject);
begin
    SpSharedRecoContext.EventInterests := SREAI-
    IEvents;
    SRGrammar := SpSharedRecoContext.CreateGram-
    mar(0);
    SRGrammar.CmdLoadFromFile (' Color.xml' ,
    SLODynamic);
    SRGrammar.CmdSetRuleIdState(0, SGDSActive)
end;
```

以上采用了两个不同的 ISpeechRecoGrammar 方法初始化语法规则, 用 CmdLoadFromFile 方法装载编写好的语法 XML 文件, 用 CmdSetRuleIdState 方法激活语法文件重点规则, 当第一个参数为 0 的时候所有规则都被激活, 第一个参数也可以是单个规则的语法 ID, 单个激活规则。

(4) 响应语音识别事件

当程序初始化成功后, 语音处理模块就被调用, 并在后台运行, 应用程序就可以接受语音输入了。例如有语音输入被监测到, SAPI 就会有 onSOUND-START 事件发生, 语音结束被监测到就有 onsoundend 事件发生, 而有词语成功被识别就会有 onRecognition 事件发生。而应用程序只需要感兴趣的某个事件响应函数中加入相应的程序即可。

本文的程序用到的 SpSharedRecoContext 对象有 18 个事件可以响应, 这里主要利用 onRecognition 事件, 当 onRecognition 事件发生意味有语音被成功识别, 首先要做的是判断识别的词语是不是本程序所需要的, 即是否为 XML 文件中预先定义了的语音命令, 并且查明具体是哪个命令。方法是在 onRecognition 事件响应函数中对语音识别返回信息对象中的识别属性值进行提取, 如果有 XML 语法文件中定义的属性值, 表明定义的语音命令被识别, 就可以根据属性值执行对应的控制程序。onRecognition 识别响应函数如下:

```
procedure Tcolor1.SpSharedRecoContextRecognition(
ASender: TObject; StreamNumber: Integer;
StreamPosition: OleVariant;
```

```
RecognitionType: TOleEnum; const Result: IS-
peechRecoResult);
begin
    with Result.PhraseInfo do
        begin
            case GetPropValue (Result, [' colour' , '
            colourvalue' ]) of //获取识别信息属性值
                1: Color := clRed; //属性值是 1 则设置设置
                界面为红色
                2: Color := clBlue; //属性值是 2 则设置界
                面为蓝色
                3: Color := clGreen; //属性值是 3 则设置界
                面为绿色
                5: color1.Close(); //属性值是 5 则程序退出
            end
        end
    end;
```

当运行程序后, 对麦克风说“红色”, 界面就变成红色, 说“蓝色”, 界面就成为蓝色。说“退出”, 程序就自动退出。

该程序可以很方便的扩充和应用到其他应用程序中, 只要对辞书文法文件进行扩充, 加入感兴趣的命令和对应属性值, 并在消息响应函数增加响应控制, 利用系统 API 以及程序模拟键盘、模拟鼠标等技术就可以实现对电脑的多种语音控制。例如语音控制菜单、开启收发电子邮件程序, 控制开启播放器等软件控制; 通过外接接口还可以实现语音控制家电等控制功能。

5 结束语

可以预见, 随着语音识别等技术的发展, 计算机就会变得越来越人性化。甚至一些专家认为, 尽管键盘仍将是 PC 机的重要外设, 但在不久的将来, 语音操控计算机会成为主流。开发具有语音控制功能的应用软件将具有实际的意义。

本文作者创新点: 本文讨论的用 Delphi 自动化接口方法开发的基于 SAPI 的语音控制应用程序, 可以实现语音控制功能; 其过程简单快捷; 并且可以很容易通过修改辞书文法文件和语音识别响应函数扩展程序功能; 也能方便地移植到其他应用程序使其具有语音控制功能。对开发实用语音控制功能的程序具有参考价值。

参考文献

- [1]Microsoft Corporation. Microsoft Speech SDK SAPI 5.1 Help. Microsoft Corporation, 2001
- [2]Agus Kurniawan. Simple Program for Text to Speech Using SAPI [EB/OL]. <http://www.codeproject.com/audio/speech.asp>
- [3]宋阳等. 利用 SAPI5 完成中文语音音素分解. [J] 微计算机信息. 2005.3: 230-231.

(见 38 页)

能够找到最佳的操作方式、方法,进一步提高劳动技能,保证产品质量的持续提高。

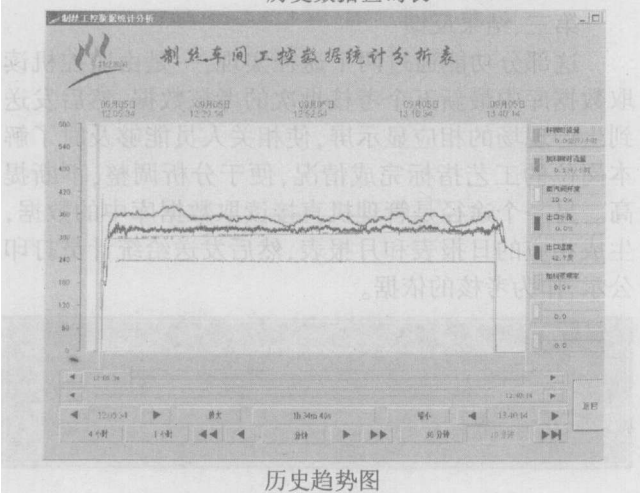
其次,把过程控制指标与设备的各项性能参数放在一起对比分析,可以及时发现生产设备的性能缺陷,以便及时维修或改进,从而提高设备性能,保证产品质量。

第三,对于出现的问题,可通过查询历史数据追溯到每一个生产环节,使解决问题更加方便、快捷、科学。

为了便于对这些历史数据进行科学的查询、分析,我们编写了历史数据查询程序。整套程序分为两部分:数据表格查询和趋势图查询。对于前一部分,我们利用 VB 编写,完成从数据库提取数据报表的功能;对于后一部分,我们利用 Intouch 的历史数据来显示查询批次的历史趋势图。通过它们之间的参数传递及相互调用,实现数据表格与历史趋势图的相互对应。



历史数据查询表



历史趋势图

6 结束语

数据质量控制则是一种与经验质量控制所不同的更科学、更精细的质量控制方法。它是把生产流程的各个细节作为研究对象,以计算机数据采集为依据,建立一套数字描述体系,即用数字描绘生产流程中关键环节的关键质量情况。通过控制关键环节的质量稳定性达到最终产品质量的稳定性和优质性。同时

通过对历史质量数据进行统计、分析和挖掘,必然会从这些数据中发现问题,并找到问题的答案,从而达到“节约、降耗、提高产品质量”的目的。

创新点:用数字描绘生产流程中关键环节的关键质量情况。通过控制关键环节的质量稳定性达到最终产品质量的稳定性和优质性。

参考文献:

- [1]黄宣康;邱龙英.《实行标准偏差和平均值考核管理方法提高滤棒、烟支加工质量》.烟草科技,2003,1002-0861, No.2, p.5-7
- [2]康军;戴冠中.《基于 Internet 的工业以太网远程监控系统设计》第 15 届中国过程控制会议论文集,2004, p.287-289
- [3]张桂珍,浅谈数据挖掘的重要性,第十四届全国管理信息系统学术年会,1999, P.367-371
- [4]张向东,烟草企业工序质量信息系统及控制方法研究,2003
- [5]罗智佳等.基于以太网的分布式数据采集监控系统的应用[J].微计算机信息.2006.1-1:22-23

作者简介:孟庆华,男,1969年8月出生,汉族,工程师,研究生学历,毕业于北京交通大学,计算机工程专业,获工学硕士学位。现就职于将军集团济南卷烟厂,主要从事生产及技术管理工作

Author brief introduction: Mengqinghua, Male, Born in August 1969, Chinese, Engineer, Post Graduate Degree, Graduated from Beijing Communication University, Majored in Computer Engineering, Acquired Master's Degree of Engineer. Works in Jinan Cigarette Factory, General Group, Mainly engages in production and technology management. E-mail: mengqinghua@21cn.com (250100 济南将军集团济南卷烟厂) 孟庆华 段三青 尹旭梅 (投稿日期:2005.8.26) (修稿日期:2005.9.6)

(接 23 页)

作者简介:童强(1968生),男,湖北黄石人,讲师,工程师,硕士,主要研究方向:多媒体应用、数据库应用等; E-mail:ttt_99@sina.com.cn

(435002 湖北黄石湖北师范学院计算机科学系) 童强

(Computer Science Department of HUBEI Normal University, HuBei HuangShi 435000 China) Tong, Qiang

(投稿日期:2005.8.26) (修稿日期:2005.9.6)

(接 35 页)

Author brief introduction: Jiang Shouxu (1958-), Man, Han Nationality, Professor, Harbin industrial of Technology, The Master graduate student teacher of Computer software and theory discipline.

(157100 牡丹江大学机电系) 李静

(157100 mechanical and electrical compartment, Mudanjiang University) Li, Jing

(投稿日期:2005.8.26) (修稿日期:2005.9.6)