

文章编号:1671-4229(2021)03-0009-11

基于 Python 的微信远程桌面控制程序的设计及实现

徐大伟^{1,2}, 高家奇¹, 祝烈煌^{2*}, 吴阜东¹

(1. 长春大学 网络安全学院, 吉林 长春 130022; 2. 北京理工大学 网络空间安全学院, 北京 100081)

摘要: 随着移动互联网和远程办公时代的到来,手机 APP 的多样化不仅丰富了手机的功能,也方便了人们的工作和生活。同时,手机的便携性是对电脑的补充,尤其是在消除操作不便方面。为加强手机与电脑的连接,解决因距离远而带来的不便,文章设计并实现了一个基于 Python 的微信远程桌面控制程序。该方案结合了 Itchat、图灵机器人等关键技术,采用软件工程思想设计了微信远程桌面控制软件的整体功能模块和流程。总体设计的基础按照以下步骤进行,包括 CMD 命令执行模块、远程截屏模块、远程实时拍照模块、消息助手控制模块、文件助手模块、一键发送模块和智能聊天模块。分步细分设计方法。结合现有已完成的设计工作,对上述功能模块进行编程实现。最后,通过对所有功能模块分别进行测试,验证了微信远程桌面控制软件程序的有效性。

关键词: 远程桌面控制; 微信; Python; 安全性

中图分类号: TP 311.1 文献标志码: A

Design and implementation of WeChat remote desktop control program based on Python

XU Da-wei^{1,2}, GAO Jia-qi¹, ZHU Lie-huang^{2*}, WU Fu-dong¹

(1. College of Cybersecurity, Changchun University, Changchun 130022, China;

2. School of Cyberspace Science and Technology, Beijing Institute of Technology School, Beijing 100081, China)

Abstract: With the advent of the era of mobile Internet and telecommuting, the diversity of mobile phone APPs not only enrich the functions of mobile phones, but facilitate people's work and lives. At the same time, the portability of mobile phones is a supplement to the computers, especially in eliminating operation inconvenience. To strengthen the connection between mobile phones and computers and solve the inconvenience caused by long distance, this paper designs and implements a remote desktop control program based on Python. The program combines Itchat, Turing Robot and other key technologies, which designs the overall function module and process of WeChat remote desktop control software by adopting a software engineering idea. The basis of the overall design follows the steps, including the CMD command execution module, remote screen capture module, remote real-time photography module, message assistant control module, file assistant module, one-button sending module and intelligent chat module, designed by the step-by-step subdivision design method. Combined with the existing completed design work, the above functional modules are programmed to achieve. Finally, the effectiveness of WeChat remote desktop control software program is verified by testing all function modules respectively.

Key words: remote desktop control; WeChat; Python; security

作者简介: 徐大伟(1977—),男,副教授,博士研究生. E-mail: xudw@ccu.edu.cn

*通信作者. E-mail: liehuangz@bit.edu.cn

引文格式: 徐大伟,高家奇,祝烈煌,等. 基于 Python 的微信远程桌面控制程序的设计及实现[J]. 广州大学学报(自然科学版), 2021,20(3):9-19.

现今国内外市场上大部分用户电脑安装的是微软的 Windows 系统。手机系统的市场份额大致被 IOS 和 Android 平分天下,导致了手机和电脑端显现出一种生态不统一,移动端和 PC 端割裂开来的现象。

目前带有控制功能的软件,如通过手机控制电脑的程序很少,实现类似功能的软件有向日葵^[1]、HiPC^[2]和 TeamViewer 等^[3]。这些软件的共性都是通过下载手机 APP,同时在电脑端下载对应软件,实现控制效果。但是相对于通过微信控制手段,上面的 APP 软件需要一直在系统后台运行^[4-5],这种方式会增加手机负载。此时微信控制电脑的优势便显现出来,通过微信 APP 登录,可以和微信共用一个进程,减少手机系统负载,同时实现手机控制电脑的效果。

1 研究意义及方法

1.1 研究的意义

移动互联网时代,人们工作生活中,移动端和 PC 端联系越来越密切。绝大多数人现在外出更多是携带手机而不是笨重的电脑,但是又离不开电脑强大的计算、处理能力和各种手机不可代替的应用程序。手机操作系统多数是 Android 和 IOS,PC 操作系统多是 Windows、Macos 和 Linux 等,系统之间的差异使得手机和电脑之间总是存在间隔。控制软件设计的意义是加强手机与电脑之间联系。打开微信向自己的 PC 端微信远程发送命令,来控制 PC 机,不仅能够远程对自己的电脑乃至他人的电脑进行监控,还可以远程使用电脑的计算能力帮助完成复杂事务,方便了人们日常的生活。

1.2 研究方法

控制软件是基于 Python 的一个应用程序,通过在 PC 机上编写 Python 程序并生成二维码,用手机微信扫描二维码,使得二者关联起来。一方面实现用户利用手机端微信 APP 远程登录到远端 PC 机,通过微信远程桌面控制程序控制 PC 端;另一方面也是对微信功能的拓展。

系统设计使用的集成开发环境是 Pycharm^[6-7],编写程序的环境是 Python3.x 以上版本和 Windows10,在运行程序之前需要安装好 OpenCV-Python、Itchat。可以通过命令 `pip install + 安装模块`实现相关模块的安装。

程序的技术路线是引用 Itchat^[8]库函数的功

能,通过使用 Itchat 库向网页端微信发送 request 请求,利用扫码登录到微信网页端,然后通过 Itchat 来发送消息和接收消息。通过 OpenCV^[9]内置功能强大的图像处理函数,来调用电脑的摄像头,并可以自动拍下电脑前的画面^[10]。通过各种 request 库来访问特定的网页,进而获取大量的自定义信息。同时也可以帮助用户完善微信社交、群发和自动回复等功能。也可根据用户的需求进行自定义功能^[11]。

2 系统关键技术分析

2.1 开发语言 Python

Python 是 20 世纪 90 年代诞生的,创始人是芬兰的 Guido van Rossum^[12]。它是一款功能强大,集开放性、扩展性和简易性于一体的语言。起初 2004 年发布 Python2.7,随后 2008 年又发布了 Python3,如今已经成为大数据、人工智能必不可少的语言^[13]。

Python 之所以被用作开发语言,主要是因为其内置了丰富的函数库,代码短小,容易理解^[14]。主要用于 3 大领域:

(1) Web 开发,Python 中有很多开发框架,例如 Django、Pyramid 和 Flask。这些框架有着开发周期短和方便调整等优势^[15];

(2) Python 在人工智能和大数据领域有着得天独厚的优势。而如今正是大数据时代,相信 Python 语言也会在这个过程中不断发展壮大^[16];

(3) Python 在嵌入式领域也有着广泛的应用,随着万物互联的时代到来,设备、平台网络和应用的开发都离不开 Python。相信 5G 时代的到来,Python 在物联网领域也会得到更好的发展^[17]。

2.2 开发工具 Pycharm

Pycharm 是一种 Python IDE,可以使得用户在编写 Python 语言时提高效率,完备的功能如代码跳转、自动完成、单元测试、脚本控制、调试、语法高亮和 Project 管理功能等,方便用户使用。此外,Pycharm 在连接 xshell 和 virtual box 方面特别方便,在使用到 Django 时,Pycharm 也有自带的 server 方便使用。

2.3 关键的接口

2.3.1 Itchat 基本介绍

Itchat 是一个开源的用于微信的多功能接口,使用它可以简化 Python 调用微信,通过简单的代码就可以完成需要处理的全部信息^[18]。

Itchat 还有很多功能,例如利用 API 可以轻松获取你的微信好友网名、性别、年龄、所在地分布、接受并处理你的好友请求和群消息收发等。如今微信已经成为大部分人的社交方式。利用 Itchat 能够扩展个人的微信功能,给用户使用带来方便。

Itchat 是一个 Python 第三方库,可以采用 pip install Itchat 安装这个库,其中提供了登录微信网

页端的 auto_login() 方法,当执行此方法时 Itchat 会对“https://login.weixin.qq.com”发送 request 请求,会自动生成一个二维码,然后进行扫码登录微信网页端。如果扫描成功,程序会提示已经扫描成功等待确认。当在微信中点击确认登录微信网页端时,程序会显示用户(网名)已经成功登录。具体流程见图 1^[19]。

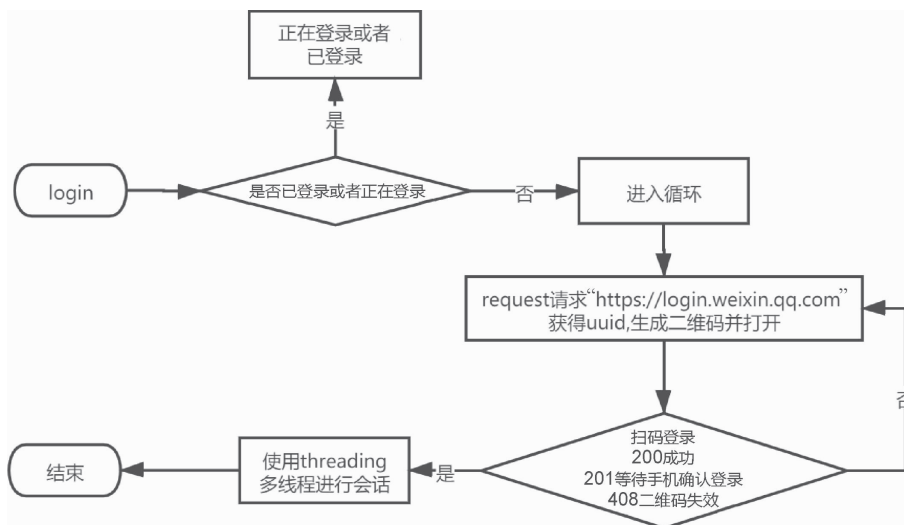


图 1 Itchat 原理图

Fig. 1 The principal diagram of Itchat

2.3.2 图灵机器人介绍

图灵机器人是隶属于光年无限科技的一家人工智能公司。它致力于研究一种分析语言文字并且可以和人对话的机器人,可以为自然语言对话、深度问答等人工智能领域服务。并且随着使用次数越来越多,机器人可以每天 0.8% 的增速进行学习^[20]。

此程序中的智能机器人核心来自于图灵机器人,通过获取 API,并且设定机器人属性,来达到切合用户身份的聊天。同时此机器人也是控制软件中智能聊天模块的核心^[21]。

其核心原理是通过用户输入文字进行语义识别,包括:实体识别、用户感情识别、指代消解和省略恢复、恢复确认和拒识判断。然后根据识别内容进行信息的匹配,匹配方式包括 Web、社区问答、搜索引擎和知识库。最后根据检索到的数据进行过滤并输出。

2.3.3 自定义消息介绍

控制程序中的天气信息来自于聚合数据,包括获取天气预报、新闻头条、菜谱和号码归属地查询等诸多数据信息。通过 request 请求访问“http://

apis.juhe.cn/simpleWeather/query”并且返回 json 格式的的参数,包含天气状况、湿度、风力等信息。输入当前城市请求参数如下:

请求地址: http://apis.juhe.cn/simpleWeather/query;

请求参数: city = %E4%B8%B9%E4%B8%9C&key = key;

请求方式: GET# GET 传进的参数是当前城市的 utf8 URL encode 编码^[22]。

本程序通过 get 请求访问“http://open.iciba.com/dsapi”,程序返回一个 json 格式推送内容,之后利用 Python 的信息筛选功能,根据你想要的模块进行筛选。同时本程序不仅限于此,还可以根据用户需求自定义数据。

3 需求分析与模块设计

3.1 需求背景

移动互联网的发展,人们对于手机的使用远远高于电脑。可能会遇到诸如需要用手机发送一个文件给同事,但是由于文件存储在电脑中导致

无法用手机发送的情况。为了满足大多数人对于远程控制的需求,方便人们的生活,因此,开发一款小巧并且功能强大的远程控制桌面的应用程序十分必要。

3.2 功能需求分析

根据用户的基本需求并且结合软件工程思想,设计远程桌面控制程序可以从以下 3 个方面来考虑:①远程控制方面,需要实现基本电脑启动关闭操作、可以远程运行电脑中的程序和软件等功能;②安全性方面,需要实现远程监控自己电脑是否被使用、查看当前电脑桌面状态等功能;③办公自动化方面,需要实现远程读取并下载文件、忙碌语音助手、每日打卡固定消息等功能。基于以上 3 个方面,特设计以下 8 种功能:

(1) 远程调用 CMD 命令实现远程关机、延迟关机、重启、注销当前用户等功能;

(2) 远程实时拍照:通过调用摄像头进行拍照获取当前电脑前的实时状况;

(3) 远程实时屏幕截图;

(4) 获取本地外网 IP 地址;

(5) 查看本地电脑的文件,并且通过微信将文件发送到手机端;

(6) 将指定目录下的文件下载到手机端;

(7) 给特定的人一键发送特定的消息;

(8) 开启自动回复聊天机器人功能。

3.3 系统总体设计

通过以上所描述的功能进行模块的设计,总共可以分为 8 大基本模块:①CMD 模块;②screen 远程屏幕截图模块;③camera 远程实时拍照模块;④消息开启关闭模块;⑤获取 IP 地址模块;⑥文件助手模块;⑦一键发送模块;⑧智能聊天模块。系统整体设计如图 2 所示。

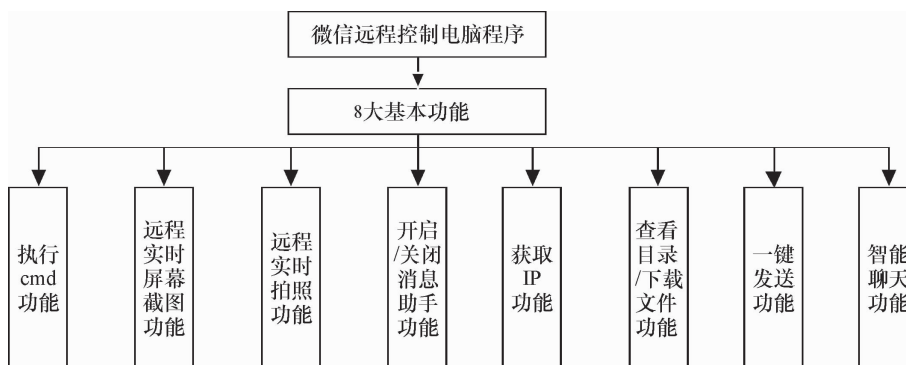


图 2 系统总体设计模块

Fig. 2 System overall design module

系统模块通过扫描二维码登录微信网页端进而启动文件助手,通过文件助手发送使用说明,等待用户输入指定的字符串,匹配关键字进行相应功能的调用。

扫码的主要功能是连接手机和电脑。此二维码看似是运行在程序端二维码,其实是用来进行

登录微信网页端的二维码,使网页端微信可以获得个人微信的全部内容,并且网页端和手机端可以互相传递消息^[23]。因此,在电脑端运行程序,主要功能是获取微信网页端的内容,为相应的功能提供关键数据。三者关系见图 3。



图 3 电脑端和手机端交互过程

Fig. 3 Interaction between computer and mobile phone

3.4 系统流程设计

系统流程描述如下:①执行该程序自动生成二维码,使用微信扫一扫程序进行扫描登录;②电

脑端会给微信的文件助手发送一个使用说明(默认消息助手是打开的),输入对应的字符串,程序会根据用户的输入匹配并运行相应的功能。输入

astc 可以关闭消息助手,此时关闭整体模块功能。如果想再次唤醒该功能需要输入 ast. 系统总体流

程设计见图 4。

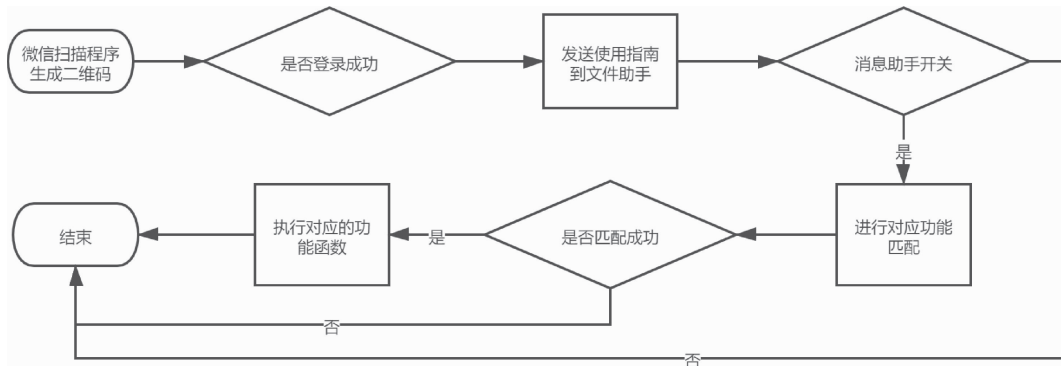


图4 系统总体流程图

Fig. 4 System general flow chart

(1)首先程序运行时会等待用户输入,当用户根据说明文档,想要对电脑进行控制的时候,用户只需要输入 CMD + 想要执行的命令即可,例如 CMD shutdown -s -t 0。此时程序会自动匹配该关键字对应的功能,然后通过 strip() 函数过滤掉之前的关键字,并且调用 Python 系统库 OS 中的 os.system() 将剩余字符当作 CMD 命令行来执行,如果命令格式不符或者输入错误则不执行。

(2)判断用户输入的关键字是否是 screen,当字符串匹配成功后执行 window_capture(), 此函数功能是获取屏幕截图并将图片保存到本地,但是由于屏幕截图通常像素点较多,图片较大,所以为了减少发送开销并且加快传输速度,调用 produce_image() 函数,将图片同比例缩小,并保存到本地,将保存好的图片发送到微信文件助手。

(3)判断用户输入合法之后,执行 camera() 函数,这里使用的是 cv2 图像处理库中的 VideoCapture() 函数,启用电脑的前置摄像头进行拍照,将照片保存到本地并将照片发送到微信文件助手上,调用 release() 函数^[24],结束进程调用,完成一整套流程。

(4)在程序运行大框架之外有一个 flag 标志位,每次运行都会检查此标志位,也就是消息助手开关,当标志位(flag)为 1 的时候,程序正常检测用户输入的内容是否与关键字匹配,当标志位(flag)为 0 的时候程序不检测除了开启关键字之外的关键字,此时用户可以输入任意文字。

(5)安全设计考虑,同时为了方便定位个人电脑位置,程序提供了获取计算机外网 IP 的功能,当用户输入关键字 IP 时,程序执行 IP_getAdr(),

此时该程序会访问 <http://ifconfig.me/ip>. 获取本机对外 IP 地址,并且通过聚合数据自动获取所在的城市。

(6)文件助手设计目的是为了更方便用户日常使用手机发送文件,同时用电脑接收和编辑文件。对于电脑端上保存和编辑的文档,可以远程通过手机端控制,方便进行电脑文件远程传送。控制程序在查找到文件的绝对路径之后,通过 download + 文件路径\文件名,下载该文件,文件会通过微信文件助手,发送到用户手机端。

当需要下载文件(如上述描述的功能),但是记不住电脑中文件具体位置的时候,输入 dir + 路径的方式,查看文件目录,程序执行 CMD 命令功能并将结果写入一个记事本当中,并通过文件助手功能将该目录发送至用户的手机,方便查找电脑中文件的位置。

(7)一键发送是为了满足用户没有时间频繁地发送大量重复的文字和枯燥的数据而设计的功能。

用户只需要输入一个固定的关键字(girl),便可以实现给固定的用户或者群,发送复杂消息的功能。此程序用给朋友发送消息为例进行说明,当输入 girl 关键字时,程序会执行相应 get_weather_focast(), 此时程序会自动获取当前用户所在位置,并且通过聚合数据查找该地的天气预报,聚合数据会返回一个 json 格式,获取该天气预报关键内容发送给朋友,同时附加每日一句英文和对应翻译。从而实现了一键发送复杂消息功能,大大减少了用户平时打字带来的时间和精力上的成本,同时还可以自定义发送消息内容,来实现各种场景如:每次发布群推送消息,每日打卡、每日发

送今日新闻等等。

(8)智能聊天功能的设计是为了改善用户回复消息慢、不及时等问题,以及作为商业用途需要,有大量的日常消息需要回复的场景,同时对于个人使用也可以礼貌性的回复各种简单问题^[20]。通过图灵机器人这个关键人工智能技术,实现聊天机器人功能,如果用户暂时处于忙碌状态不能及时回复消息,输入 robot 关键字启动聊天机器人。当用户接收到其他人发送的消息时,图灵机器人会根据当前的人物设定以及收到的消息进行自动回复。同时可以设定机器人属性,来实现简单的问答,解决基本的问题。

4 系统功能实现

4.1 整体功能函数

Python 作为脚本语言的一种,和 C 语言 C++ 等有 main() 函数作为程序的入口的编程语言有所不同。它的运行过程中,不会进行编译生成二进制文件再运行,而是从上到下运行代码,所以为了开发方便,并且提高程序的可读性,Python 会模拟一个程序入口,定义为 if __name__ == '__main__',相当于 C 语言中的 main() 函数^[25]。本程序整体模块函数是在此函数下面运行的,具体核心代码如下:

```
if __name__ == '__main__':
    Itchat. auto_login (hotReload = True)
    Itchat. send (usageMsg, "filehelper")
    Itchat. run ()
```

其中,程序执行 Itchat. auto_login (hotReload = True) 时会自动调用库函数生成一个用于登录微信网页端的二维码。登录成功后会执行 Itchat. send (usageMsg, "filehelper"), 向微信文件助手发送指定说明文档。此时手机微信就会收到该程序功能说明,从而简化操作。

4.2 各函数功能

控制功能是本程序的基本功能,是为了方便用户可以远程控制电脑,其中包括关机、注销、重启等各种操作。该功能主要是通过 Python 强大的系统控制模块 OS 中的 system 方法,可以通过将用户输入的字符串过滤,并执行 CMD 程序,在程序启动时发送说明文档,使用户更加简单地实现控制效果。具体的核心代码如下:

```
if message[0:3] == "CMD":
```

```
os. system (message. strip (message[0:4]))
```

```
Itchat. send ("已成功执行% s" % message,
"filehelper")
```

(1) 远程屏幕截图功能

屏幕截取的功能主要是通过 win32gui 中相关函数,通过窗口句柄获取窗口的设备上上下文,并且设置宽度与高度进行截取,并把图片保存到本地。关键伪代码如下:

```
def window_capture (filename):
```

```
...
```

```
获取桌面参数:分辨率,桌面大小等
```

```
...
```

```
# 创建 bigmap 准备保存图片
```

```
saveBitMap = win32ui. CreateBitmap ()
```

```
# 获取监控器信息
```

```
MonitorDev = win32api. EnumDisplayMonitors (None, None)
```

```
w = MonitorDev[0][2][2]
```

```
h = MonitorDev[0][2][3]
```

```
# 为 bitmap 开辟空间 saveBitMap. CreateCompatibleBitmap (mfcDC, w, h)
```

```
# 高度 saveDC, 将截图保存到 saveBitmap 中
saveDC. SelectObject (saveBitMap)
```

```
# 截取从左上角(0,0)长宽为(w,h)的图片
saveDC. BitBlt ((0, 0), (w, h), mfcDC,
(0, 0), win32con. SRCCOPY)
```

```
saveBitMap. SaveBitmapFile (saveDC,
filename)
```

(2) 远程实时拍照功能

安全功能主要包括调用电脑前置摄像头对用户的拍摄,这样可以实现对于电脑的使用者以及使用者正在进行的操作进行初步的掌握。如果这台电脑不仅仅是自己使用,那么会极大程度上保证电脑的安全性,调用摄像头并拍照的核心代码如下:

```
def camera():
```

```
cap = cv2. VideoCapture (0) #获取摄像头
```

```
ret, img = cap. read () #将图片保存到 img
```

```
cv2. imwrite ("weixinTemp. jpg", img) #将
图片保存到本地
```

```
Itchat. send ('@ img @ % s' % u'weixin-
Temp. jpg', 'filehelper') #将图片发送
```

```
cap. release () #释放进程
```

该函数功能是用户调用摄像头进行拍照。

(3) 获取 IP 地址功能

首先此程序运行需要联网状态,通过访问 `http://ifconfig.me/ip` 这样一个可以获取外网 IP 地址的 URL,来获取本机 IP 地址,具体的代码如下:

```
def IP_getAdr():
    localIP = requests.get('http://ifconfig.me/ip',
        timeout=1).text.strip() # 获取外网 IP
    url =
    'http://apis.juhe.cn/ip/ipNew? ip =' + localIP
    + '&key = 2e9607ccb4629860786a13c8c62d185d'
    local = requests.get(url).json()
    Country = local['result']['Country']
    Province = local['result']['Province']
    City = local['result']['City']
    Itchat.send("您的电脑的当前 IP 是:%s,
        所在地区为:%s%s%s"%(localIP, Country, Province, City), "filehelper").
```

上述部分代码会获取到 PC 机对外 IP 地址,以及相应 IP 地址的地理位置,将结果保存到变量 `local` 中,然后通过 `Itchat` 的消息发送功能将 IP 等相关信息发送至微信文件助手上。

(4) 文件助手功能

文件助手的主要功能是查看电脑的文件目录,并将想要的文件通过输入文件所在的绝对路径的方式发送至微信文件助手,主要分为 2 个主要函数,即 `dir()` 函数和 `download()` 函数。该功能主要通过对文件的操作实现。

首先 `dir()` 功能核心是调用控制模块中的 `CMD` 命令,将运行的结果保存到本地一个临时的记事本文件,之后通过 `Itchat` 文件发送,将此文件发送到微信中。核心代码如下:

```
os.system(message + r' > D:\pycharm\item\directory.txt')
Itchat.send("@fil@ %s" % r'D:\pycharm\item\directory.txt', 'filehelper')
```

通过查看文件目录找到用户所要下载的文件,然后根据目录绝对路径,调用 `download()` 函数进行下载。`Download` 核心代码如下:

```
status = Itchat.send_file(msg.strip(msg[0:9]), 'filehelper')
if status:
    Itchat.send_msg("您已经成功将该文件下载到手机:\n", "filehelper")
else:
```

```
Itchat.send_msg("您所下载的文件不存在。 \n", "filehelper")
```

`Itchat.sendfile()` 函数成功找到文件返回 `true`, 否则返回 `false`, 通过返回值判断是否存在该文件,并发送相应的状态信息。

(5) 一键发送功能

一键发送功能的核心是数据收集以及对于微信用户信息的获取,该功能首先是根据用户想发送的信息进行定制化信息的收集,Python 的 `requests` 提供访问 url 并获取信息的能力,通过访问聚合数据这个多样化的信息数据收集网站,来获取诸如本地天气、当日新闻、快递单号和航班查询等各种信息,核心代码如下:

```
weather_forecast = requests.get(url).json() # 获取 api 结果
result = weather_forecast['result']
today_weather = weather_forecast['result']
['realtime']
city = result['city'] # 获取城市
temperature = today_weather['temperature']
# 获取温度
weather = today_weather['info'] # 获取天气
wind = today_weather['direct'] # 获取风力
```

本程序还通过获取金山词霸每日推送,以方便快捷地一键发送至指定的用户核心代码,代码如下:

```
content = sentence['content'] # 抓取英文句子
translation = sentence['note'] # 抓取中文句子
```

(6) 智能聊天功能

智能聊天功能的核心是人工智能图灵机器人。该功能首先通过 `Itchat` 这个第三方库函数,获得手机微信消息,并记录下该用户,此后将该用户的消息通过 `requests` 请求,将其发送到图灵机器人服务器端。具体的代码如下:

```
apikey = '80e19a0534f34c839b9843f3ec286d31'
// 图灵机器人的 api 密钥
api_url = 'http://www.tuling123.com/openapi/api' // 图灵机器人接口
data = {'key': apikey, 'info': '你好'}
req = requests.post(api_url, data = data).text
replies = json.loads(req)['text']
print(replies)
```

图灵机器人收到消息后会根据此机器人的属性,进行智能回复,并通过 `Itchat` 将回复信息返回

到对应用户的微信上。程序整体框架代码如下:

```
@ bot. register(msg_types = FRIENDS)
def auto_accept_friends(msg):
    // 自动接受好友请求
    wx_reply. auto_accept_friends(msg)
```

5 系统测试

5.1 测试环境要求

此程序测试环境要求 win7 及以上操作系统,需要安装 Python 环境,由于系统的远程控制命令是基于 windows 命令提示符所编写的,所以无法在 windows 以外的环境运行。同时需要安装 Python 的第三方库 requests 库和 Itchat 库。如果想使用 camera 功能,电脑端还必须带有摄像头。

此程序的微信要求必须是可以登录微信网页端的微信,如果安全性不高或者部分微信号不能登录微信网页端,则不能实现此功能。同时要求微信安全性必须达标。

5.2 运行主界面

程序运行后会生成一个用户登录微信网页端

的二维码。

扫描系统生成的二维码,此时手机会提示是否登录微信网页端,点击允许登录,此时手机会在上面显示已经登录微信网页端,同时运行在本机程序会显示该用户已经成功登录。

程序会向手机微信文件助手发送一条使用说明。并且每当重新运行程序的时候都会发送此说明文档,根据这个通俗易懂的说明文档进行对应的功能测试,如图 5 所示。同时给出常用的控制命令,这样对于那些不熟悉 CMD 命令的用户也可以方便的使用。需要注意的是如果使用了关机命令,PC 机被远程关机从而不能进行之后的控制。

5.3 功能及性能测试

主要针对执行 CMD 功能、远程屏幕截图功能、远程实时拍照功能、开启关闭消息助手功能、获取 IP 地址功能、文件助手功能、一键发送功能和智能聊天功能 8 个功能进行测试。

根据操作说明,输入 shutdown-s-t 100 其中 100 参数,可以指定延迟关机的时间,在输入框输入之后运行结果如图 5 所示。此时在 PC 端显示关机倒计时。

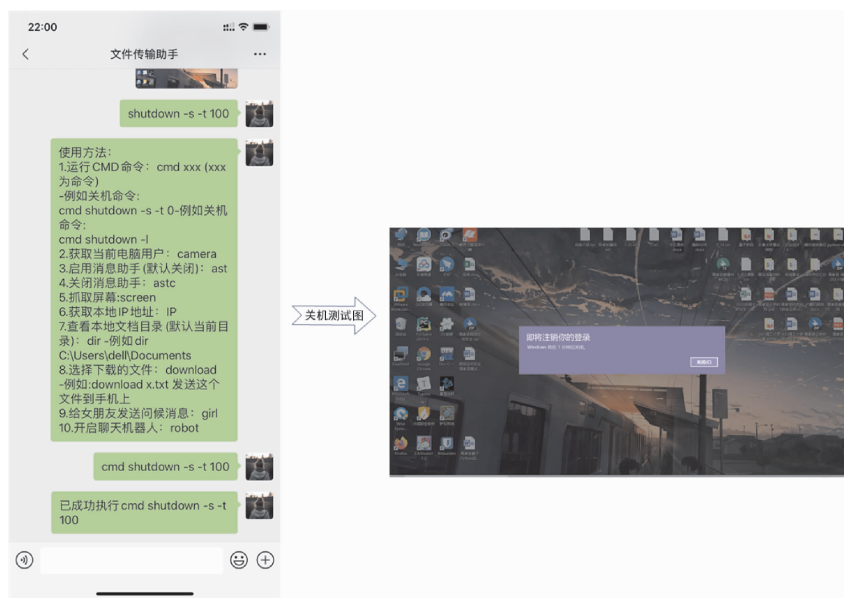


图 5 远程关机

Fig. 5 Remote shutdown

用户远程查看本机电脑屏幕截图,根据说明输入 screen,此时程序会匹配关键字进行屏幕截图功能实现,并将当前电脑屏幕截图发送到微信文件助手。功能实现如图 6 所示。

根据说明,用户远程使用手机进行功能实现,

如果要获取电脑的用户即调用摄像头拍取当前用户,可以在输入框中向文件助手发送 camera,此时系统会将当前用户以照片的形式发送到文件助手方便用户查看,功能测试图如图 7 所示。

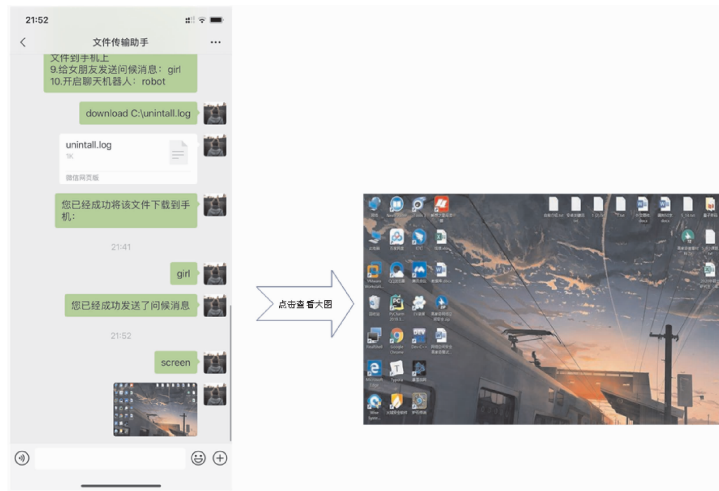


图 6 获取屏幕截图

Fig. 6 Get a screen capture



图 7 多功能运行图

Fig. 7 Multi-functional operation diagram

文件助手是本程序为了方便用户操作所添加的功能,具体的测试方法:通过输入特定的开启关闭命令,观察程序是否可以正确运行。当输入 astc 时,关闭消息助手,此时用户即使输入功能关键字也不会执行对应操作,当用户输入 ast 启动消息助手,可以重新使用对应的功能。功能测试如图 7 所示。

用户可以远程查看本机的 IP 地址,并根据其 IP 地址获取相应所在地理位置,方便查找电脑所在地,提高了安全性,在输入框输入 IP 即可获取。功能实现如图 7 所示。

用户可以远程查看电脑的文件目录,通过使用 dir + 文件路径的方式查看文件目录(默认查看当前运行文件目录),例如:dir 功能会将运行的结果保存到.txt 件中,并发送到微信文件助手,用户点击此文本文档下载文档。

用户可以通过控制程序将文件目录中的文件下载到手机中,解决出门在外却发现想要的文件

不在手机上的尴尬情况,用户在输入框中输入 download + 文件的绝对路径,进行下载,例如:download C:/uninstall.log。

如果下载成功可以将此文件发送到手机文件助手,如果该文件不存在或者大小为空,会返回文件下载失败,具体功能演示如图 8 所示。

用户可以根据想要发送的内容进行定制化一键发送,首先通过聚合数据等数据收集网站,获得需要发送的数据信息,此程序特定指发送每日天气预报、每日英语给指定的人,例如在微信输入 girl,程序获取系统设定的反馈信息,并将信息发送给指定手机端。功能测试如图 8 所示。

用户在不方便回复信息时,通过智能聊天功能进行自动回复,根据使用方法,在输入框输入 robots 开启图灵机器人,此时如果收到好友发来的消息,就会调用人工智能小助手自动进行回复。具体演示如图 9 所示。



图 8 一键发送消息

Fig. 8 Send a message with one click

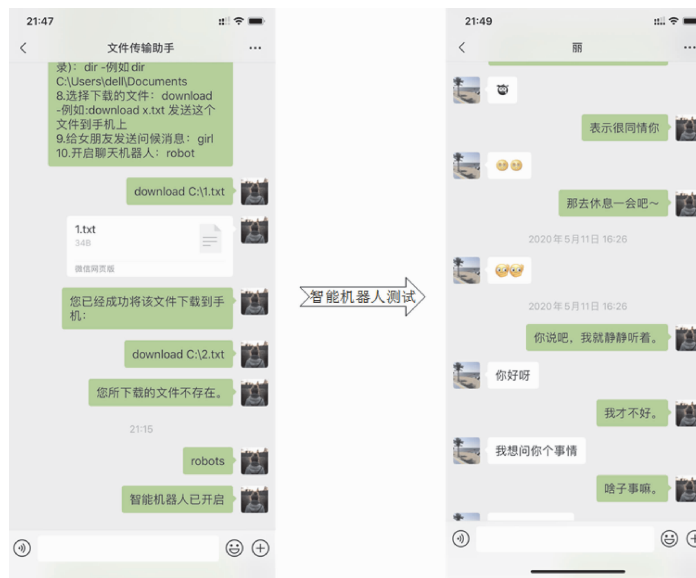


图 9 微信自动回复

Fig. 9 WeChat automatically replies

5.4 测试结果分析

经过此次程序测试结果分析,发现了该程序虽然在功能上基本实现,但是还会有很多缺陷,比如由于网速的不稳定,发送文件、图片大小不同导致手机接收图片的时间上不同。特别是当电脑屏幕分辨率不同时,会出现部分屏幕截图在发送过程中丢失、图片传输时间过长、两次执行结果而只运行一次等问题。后期通过调用图片压缩函数将其解决。同时对于计算机基本 CMD 命令不是很熟悉的用户,添加帮助文档十分必要。另外在程

序结构优化以及算法时间复杂度上还可以进一步的改进,并且程序界面还需要进一步的美化。

5.5 设计优势

根据上述设计流程,此程序的优势主要体现在以下 3 个方面:

(1) 用户不需要下载安装应用程序就可以使用远程控制,只需要扫一扫登录微信,就可以实现各种功能,方便了用户。简约的风格使得用户有不错的用户体验;

(2) 传统的 APP 需要注册登记审核等一系列

流程,这种设计使得开发成本降低,同时节省大量的时间和费用;

(3)本程序基于微信 APP 平台开发,对于普通的微信用户都可以方便地接触、使用此程序。

6 结束语

随着移动互联网的飞速发展,可以看到智能

移动设备已经在人们生活中起到了至关重要的作用,但是电脑终端以其强大的性能,方便快捷的操作在短时间内不可能被手机代替,本设计实现了一款基于 Python 的微信远程桌面控制程序,程序实现了手机和电脑之间的连接,加深二者之间的联系,不仅可以方便用户,更可以减轻用户对于电脑的依赖,通过网络进行远程控制,从而达到人机分离,不受距离的限制,方便用户实时控制。

参考文献:

- [1] 谢志俊. 远程桌面最快的软件《向日葵远程控制》[J]. 计算机与网络,2017,43(8):38.
- [2] 陈世军. 基于 SOCKET 技术的计算机远程控制实现[J]. 计算机光盘软件与应用,2012,3(5):40-45.
- [3] 李明. 浅析计算机远程控制的技术及实现方法[J]. 计算机产品与流通,2019,9(1):156.
- [4] 张卓鸿. 局域网内如何实现远程桌面控制[J]. 计算机与网络,2017,43(8):45.
- [5] Jiang W, Tian Z, Zhang H, et al. A stochastic game theoretic approach to attack prediction and optimal active defense strategy decision[C]//2008 IEEE international conference on networking, sensing and control. Piscataway: IEEE, 2008:648-653.
- [6] Matthes E. Python 编程从入门到实践[M]. 袁国忠,译. 北京:人民邮电出版社,2016.
- [7] Wang Y, Tian Z, Zhang H, et al. A privacy preserving scheme for nearest neighbor query[J]. Sensors, 2018,doi:10.3390/s/8082440.
- [8] Littlecoder, Terxpdban, Chyraoc. Itchat[EB/OL]. (2019-03-27)[2021-06-20]. <https://itchat.readthedocs.io/zh/latest/>.
- [9] Kaehler A, Bradski G. Learning openCV 3: Computer vision in C++ with the openCV library[M]. 阿丘科技等,译. 北京:清华大学出版社,2018.
- [10] Minichino H J. Learning open CV 4 computer vision with python 3: Get to grips with tools, techniques, and algorithms for computer vision and machine learning[M]. Birmingham: Packt Publishing Limited, 2020.
- [11] Montag C, Becker B, Gan C. The multipurpose application WeChat: A review on recent research[J]. Frontiers in Psychology,2018,2(2):47-52.
- [12] Yarlagaadda R T. Python engineering automation to advance artificial intelligence and machine learning systems[J]. SSRN Electronic Journal, 2018,5(6):87-97.
- [13] 崔文化. 基于 Python 的地理信息数据属性检查软件设计与实现[J]. 地理空间信息,2021,19(5):55-58.
- [14] Shafiq M, Tian Z, Bashir A K, et al. IoT malicious traffic identification using wrapper-based feature selection mechanisms[J]. Computers & Security, 2020,doi:10.1016/j.cose.2020.101863.
- [15] Meena B S. Indoor localization-based office automation system using IOT devices[J]. Intelligent Computing in Engineering, 2020,1(1):199-212.
- [16] 王萍. 计算机远程控制软件及应用实践[J]. 科技经济市场,2019(7):9-10.
- [17] Singha H S, Bhuvana J. Abrasion of current weather of a city using variant python libraries and weather application programming interface API[J]. Journal of Trend in Scientific Research and Development,2020,5(1):593-596.
- [18] 何舒. API 接口自动化测试系统设计与实现[D]. 成都:电子科技大学,2018.
- [19] 张林. 虚拟环境下的智能语音聊天机器人设计[J]. 计算机与数字工程,2019,47(10):2617-2621,2644.
- [20] 李建坤. 面向学生的智能聊天机器人[J]. 电子制作,2019,3(4):30-48.
- [21] 陈琳. 计算机通信与网络远程控制技术应用研究[J]. 无线互联科技,2019,16(4):7-8.
- [22] 安章顺,迟明路,刘荣升,等. 基于微信小程序的智能台灯系统设计[J]. 南方农机,2021,52(10):154-157.
- [23] 黄丽燕,王嘉樱. 基于 Python 和 Coh-Metrix 高考英语“一年两考”阅读文本复杂度分析[J]. 外语测试与教学,2020(3):1-11,28.
- [24] Ding Z, Yang C, Tian Z, et al. sEMG-based gesture recognition with convolution neural networks[J]. Sustainability, 2018, 10(6):1865.
- [25] 高鸿斌,申肖阳. Python 数据分析技术综述[J]. 邯郸职业技术学院学报,2018,31(4):49-51.

【责任编辑:孙向荣】