Voice-controlled Face Recognition Based on Smart Glasses

Zijun Yang^{a,*}, Seiichi Serikawa^a

^aKyushu Institute of Technology, Kitakyushu, 8048550, Japan

*Corresponding Author: zijun.yang529@mail.kyutech.jp

Abstract

This article focuses on projects that analyze human behavior and memory through smart glasses. Face recording is performed through smart glasses, relevant data is retrieved through voice control, and face recognition is performed. Based on the previous research, this paper will add a voice control system to optimize the use of the program and make the entire program easier to operate. Later stage will be further optimized, integrated packaging into a complete application to facilitate user operations.

keywords: Voice input, Face recognition, Baidu voice

1 Introduction

We want to realize more functions related to human behavior memory through smart glasses, so as to help people in need. We have already completed the face recognition part before.⁽¹⁾ This part is mainly divided into the main parts of face capture, image data training, face database generation, face detection, etc. After the detection is completed, we can browse the face through the picture browser and learn the related names at the same time. As shown in Fig. 1. The interface display uses the pygame library.

In order to make the operation easier, we have further optimized the program and added a voice operation section, which controls the operation program by voice. The entire voice operation panel is divided into the following parts: collect recording information, convert voice signals into character strings, perform corresponding operations on the obtained character strings, and output related data. We mainly use the package of Baidu Intelligent Voice to assist us in completing the entire voice control module.^{(2),(3)} Below we will introduce the ideas and content of the entire program in detail.

Fig. 1: Results display

Fig. 2: Waiting input interface

2 Program Function Introduction

When we run the program, the first thing we see is the waiting input interface. As shown in Fig. 2.

In this interface, we set a button. We monitor the operation of the mouse. When the click collision point of the mouse is within the rectangular frame corresponding to the button, the program enters the recording state. After clicking, in order to distinguish it from the standby state, the button will change color, as shown in Fig. 3.

After the program enters the recording state from the standby state, the recorder will be turned on to record the sound. There are two ways for the program to judge the end of the recording. One is to calculate the duration of the locale-free environment, and the other is to

Fig. 3: Button clicked interface

Fig. 4: Speech to text process diagram

determine whether the button is triggered again. After the recording is completed, the program will convert the entered audio file into text mode, and match the text with the keywords set in the program in advance. If the match is successful, perform the corresponding operation, for example, the voice text is “What’s your name”, the program will perform face matching; the voice text is the chat content, then the program will execute the simple chat content. The process as shown in Fig. 4.

When the program is ready to perform face matching, the camera is first turned on, the face in front of the camera is read, and it is photographed, and the captured face is compared with the face in the established database. During this process, the interface will remind you “Program is running, please wait” and other information. As shown in Fig. 5.

After the program completes the comparison, it will get the name of the recognized face, and then create a picture browser in the interface to display the relevant information of the face. As shown in Fig. 6.

The program sets different buttons, which can respectively operate the browsing of pictures and the opera-

Fig. 5: Loading interface

Fig. 6: Results display

tion of saving or not. When the identified face is “Unknown”, the user can decide whether to save the face into the database by using the “save” button and the “del” button. The above is the operation idea of the entire program, and the specific process of each part will be introduced in detail below.

3 Program Function Introduction

3.1 Basic introduction of voice input

We decided to add the voice control module to our existing program part. First we need to implement voice input. For Python, it has many methods for processing audio files, such as the built-in library *wave*, scientific computing library *scipy*, and the easy to use voice processing library *librosa*, and *Pyaudio*. Among them: *wave* module provides a convenient interface for processing WAV sound formats. It does not support compression/decompression, but supports mono/stereo;⁽⁴⁾ *librosa* is a python toolkit for audio, music analysis, and processing. Some common time-frequency processing, feature extraction, drawing sound graphics and other functions are available and powerful. It is a Python module that is usually used to analyze audio signals, but is more inclined to music;⁽⁵⁾ *scipy* tends to perform matrix processing on audio. Considering that we may add emotional speech later, we also include this library as a can-

Fig. 7: Framing model

didate;⁽⁶⁾ *Pyaudio* provides bindings for cross-platform *PortAudio* (a library that handles audio input and output), enabling users to easily record and play audio. The audio input library used by the program in this article is *Pyaudio*. After the basic voice input function is completed, we will convert the input voice signal into text to realize the operation of our program.⁽⁷⁾

3.2 Basic Principles of Speech Conversion

To realize the conversion of speech into text, we must first realize speech recognition. We know that speech recognition is different from other string recognition and picture recognition, because one of the characteristics of speech recognition is that the speech to be recognized is irregular in length, that is, before recognition, we cannot know the phoneme duration of the current speech, so When we construct a statistical model to input speech features, we cannot judge how long it takes to build the model for recognition. At the same time, most common models are not convenient to handle input features with uncertain dimensions, so we take another method is to split the input speech frame by frame and use the frame as the smallest unit we need. Of course, our split is not simply cutting the sound into small pieces, but multiplying the uncompressed pure waveform file with a “window function”. So the frames we get actually overlap. As shown in Fig. 7.

After framing, the speech becomes a lot of small segments. But because the waveform has little ability to describe in the time domain, the waveform must be processed. A common processing method is to extract speech MFCC features based on the physiological characteristics of the human ear. Mel-scale Frequency Cepstral Coefficients (MFCC) are cepstrum parameters extracted in the frequency domain of the Mel scale. The Mel scale describes the non-linear characteristics of the frequency of the human ear. Its relationship with frequency can be

Fig. 8: Frequency and mel-frequency curve

Fig. 9: The speech feature parameter MFCC extraction process

used as follows. Equation approximates:

$$Mel(f) = 2595 \times \lg \left(1 + \frac{f}{700} \right) \quad (1)$$

Where f is the frequency and the unit is Hz. The following figure shows the relationship between Mel frequency and linear frequency as shown in Fig. 8.

The speech feature parameter MFCC extraction process as shown in Fig. 9.

Now, the sound becomes a matrix of 20 rows (assuming the acoustic characteristics are 20 dimensions) and N columns, which is called the observation sequence, where N is the total number of frames. Because the pronunciation of each word is composed of several phonemes, each phoneme can be divided into several small states (usually we divide a factor into three states). So for speech to text we need the following three main steps: the first step is to recognize frames as states; the second step is to combine states into phonemes; the third step is to combine phonemes into words.

As shown in Fig. 10, each small vertical bar represents a frame, and several frames of speech correspond to one state, and every three states are combined into a phoneme, and several phonemes are combined into a word. In other words, as long as you know which state each frame of speech corresponds to, the result of speech recognition will come out. The step of identifying frames as states is the key and difficult point. What we can think of is that to see which state corresponds to a certain frame has the highest probability, then which state

Fig. 10: Frame and State Relationship

Fig. 11: Determination of status

the frame belongs to. For example, as shown in Fig. 11, this frame has the highest probability of corresponding to the S3 state, so let this frame belong to the S3 state.

And the acquisition of these probabilities depends on the hidden Markov model. We first use HMM to model the states contained in each phoneme, and then use Gaussian mixture model GMM to fit the corresponding observation frames for each state. The observation frames are combined into observation sequences in time series. Each model can generate observation sequences of different lengths, that is, one-to-many mapping. Then divide the samples into specific models by phoneme, and then learn the parameters of the HMM's transition matrix, GMM's weight, and mean variance in each model. After some learning and training, we can call the model to implement our speech recognition.⁽⁸⁾ At this point, we have completed the part of converting speech into text.

3.3 Basic procedures for voice control

We control the program and turn on the recording device. At this time, we use the button to input the first keyword by voice, such as “Who am I?”/“Who is this person?” The program judges whether to stop inputting the voice information according to the duration of the no-language environment and whether the button is triggered again. If so, the recording function is turned off to realize the next function. In the next function, the program will convert speech text to text text through Baidu speech library. As shown in Fig. 12.

Fig. 12: Voice conversion to text output

When we successfully input the speech, the speech is converted into text, which provides the program with matching keywords. At this time, the content of the keywords is to identify the face in front of the camera and the basic dialogue. When the program executes the “Face Recognition” module, we will have two situations. One is to output the name of the person who matches the face in the library successfully, and the other is to output “Unkonw”, at the same time, we will also output the corresponding voice content.

3.4 Basic Introduction to Baidu Speech Library

Below we briefly introduce the basic functions of Baidu Voice Library. Baidu Voice adopts the leading international integrated streaming end-to-end speech and language modeling method, and integrates Baidu natural language processing technology. The near-field Chinese Mandarin recognition accuracy is 98%, and it supports Mandarin and slightly accented Chinese recognition. It supports Cantonese, Sichuan dialect recognition; supports English recognition. Use a large-scale data set to train a language model, and intelligently match appropriate punctuation marks (including, ...?) According to the content understanding and pause of the speech, so that the expression of the recognition results fits the expression and is more understandable. According to the understanding of the speech content, the digital sequence, decimal, time, fraction, and basic operators can be correctly converted into a digital format, so that the recognized digital result is more in line with the usage habits, intuitive and natural. Supports self-help training models on the voice self-training platform, uploading vocabulary text can complete training with zero code, accurately improving the vocabulary recognition rate in the business field by 5-25%, and can be used exclusively.^{(9),(10)} We can get different language results through different parameters, the specific parts are as follows:

4 Face Recognition

The face recognition part is a part that has been completed before. It is mainly divided into the following steps: capturing the face to form a training set, training the training set to generate a recognizer, and face recog-

Table 1: The dev pid parameters of Baidu voice

dev pid	Language	Is there a punctuation
1536	Chinese English	no punctuation
1537	Chinese	Punctuation
1737	English	Punctuation

Fig. 13: Voice conversion to text output

nition. The first is the capture of human faces, which are collected as a basis for training. Name a person's face and use the camera for a period of time. The length of the shooting time determines the size of the training set. Take a screenshot of the camera with a certain frequency, convert the obtained picture into a grayscale image reduction operation, and use the `get_frontal_face_detector` algorithm in the `dlib` library for face detection on the grayscale image. The second is to train the data in the data set to generate a recognizer. The training of the data uses the LBPHF face recognition algorithm under OpenCV. The basic principle of LBPH (Local Binary Patterns Histograms) is: use each pixel as the center to determine the relationship between the center point and the surrounding 3×3 pixel gray value. It performs a binarization process and builds a list clockwise in order to store feature vectors of different faces. As shown in Fig. 13. In addition to this algorithm, there are many other known face recognition algorithms, such as Fisher-Faces and EigenFaces under the updated OpenCV, and libraries of known algorithms for extracting 68 and 128 face symbol values under Dlib. The main reason for choosing LBPHF is that it has the advantage of being unaffected by factors such as light, zoom rotation and translation.

Finally, face recognition. Similarly, the face captured by the camera is input to the trained recognizer and the matching is performed in the recognizer to recognize the face and match its name. At this point, the face recognition module is also completed.

5 Conclusions

We tried to record human behavior with smart glasses, and then control the program through image recognition

and voice control to better help us identify the relevant data recorded by smart glasses. At present, there are still many immaturities in the city promotion, such as the accuracy of face recognition, which has great defects in detecting face. For example, running time, we need a lot of time to run the program. Later, we will improve these two aspects, and by the way add other functions of the program, so that the program can have more functions to help people.

References

- (1) N. Kaushik and T. Sasaki and Y. Takahashi and T. Nakazawa and K. Hane : " Smart Glass Type Retinal Imaging System Using MEMS Scanner " , 2018 International Conference on Optical MEMS and Nanophotonics (OMN), pp. 1-5, 2018
- (2) L. Kharb : " A Perspective View on Commercialization of Cognitive Computing " ,2018 8th International Conference on Cloud Computing, Data Science Engineering (Confluence), pp. 829-832, 2018
- (3) Y. Han and M. Zhang and Q. Li and S. Liu : " Chinese speech recognition and task analysis of aldebaran Nao robot " , 2018 Chinese Control And Decision Conference (CCDC), pp. 4138-4142, 2018
- (4) A.Vaisnys and E. Y. Chen : " Digital audio applications to short wave broadcasting " , ISIE '97 Proceeding of the IEEE International Symposium on Industrial Electronics, pp. SS33-SS36 vol.1, 1997
- (5) P. Raguraman and M. R and M. Vijayan : " LibROSA Based Assessment Tool for Music Information Retrieval Systems " ,2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR), pp. 109-114, 2019
- (6) T. Viarbitskaya and A. Dobrucki : " Audio processing with using Python language science libraries " , 2018 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA), pp. 350-354, 2018
- (7) R. Wijaya and M. V. G. Aziz and A. S. Prihatmanto and W. Siswantoko : " A Semantic Text Similarity Model for Double Short Chinese Sequences " , 2018 International Conference on Intelligent Transportation, Big Data Smart City (ICITBS), pp. 736-739, 2018
- (8) L. R. Rabiner : " A tutorial on hidden Markov models and selected applications in speech recognition " , Proceedings of the IEEE, 257-286, 1989

- (9) <https://ai.baidu.com/tech/speech/asr>
- (10) T. Shancheng and B. Yunyue and M. Fuyu :
“FAREC — CNN based efficient face recognition
technique using Dlib ”, 2016 International Con-
ference on Advanced Communication Control and
Computing Technologies (ICACCCT), pp. 192-195,
2016