AN INTELLIGENT INTERNET-OF-THINGS(IOT) DOOR BELL SYSTEM FOR SMART NOTIFICATION ALERT

Melissa Qian[1], Yu Sun[2] and Fangyan Zhang[3]

[1]Northwood High School, Irvine, CA 92620
[2]Department of Computer Science, California State Polytechnic University, Pomona, CA, 91768
[3]ASML, San Jose, CA, 95131

ABSTRACT

This paper presents an innovative redesign of a doorbell system in order to eliminate unnecessary ringing noise from users' daily life. Employing artificial intelligence for face recognition, the IoT doorbell system define the visitors as complete strangers or someone who is expected. The next step operates based on this result; the doorbell system will either ring or send out notification to the users' phone depending on the familiarity of the visitor and the user.

KEYWORDS

Machine Learning, Deep Learning, Artificial Intelligence, Wireless Network

1. INTRODUCTION

In today's world, doorbell plays an unseeingly crucial role. Although it is just a small device that makes a sound when triggered, it is located next to almost everyone's front door as people rely on its clangorous notifications of coming visitors.

However, despite its convenience, doorbell is also bothersome. When it rings at a wrong time or is activated by an unwanted person, the deafening ringtone can easily interrupt people's normal life rhythm, making them feel extra tired [1]. For example, Sunday mornings are one of the few precious sleeping times for people who are worn out by their overloaded weekday schedules, but the bell rung by a delivery man just to inform them their package has arrived will surely ruin this. Another example, parents of a newborn baby are exhausted after working for eight hours and decide to relax while the baby is sleeping, but the doorbell rung by an uninterested political campaigner can easily ruin this as the baby starts to cry.

As a result, people might unplug their doorbell angrily, but soon plug it back in as they do not want to leave themselves at the risky edge of missing out important visitors, such as a delivery that requires a signature.

Although the overall impact of the doorbell seems to be negligible, people should never overlook the potential butterfly effect [2]. In modern society, living under the pressure from work, family and personal social life, people will inevitably be in some extent of anxiety that keeps building up if they lack proper rest [3].

Natarajan Meghanathan et al. (Eds) : NLP, ARIA, JSE, DMS, ITCS - 2019
pp. 43-50, 2019. © CS & IT-CSCP 2019 DOI: 10.5121/csit.2019.91205

As a solution, the approach presented in this paper includes revolutionizing a traditional doorbell with features such as internet connection and face recognition. By attaching a camera on its device, Smart Doorbell will record the visitor's face when they press the button and use its algorithms to determine the familiar level of the visitation to the house owner. Based on the result, Smart Doorbell will then choose one or both of the notifying methods: rings directly or sends out notifications to the user's phone through the corresponding app, along with the visitor's photo, prediction of his or her identity and the visiting time.

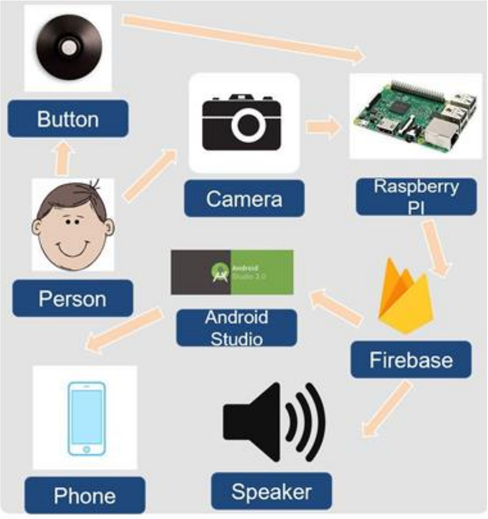


Figure 1: Component of IoT Doorbell System

The rest of the paper is organized as following: Section II details the challenges encountered during the research and building process of the project while Section III demonstrates the solutions; Section IV displays the empirical results on the accuracy of the face recognition; lastly, Section V, Section VI, and Section VII include related work, conclusion and future work, and references respectively.

2. CHALLENGES

2.1. Setting up connection between Raspberry Pi and users' phone

When outlining my project, I had some difficulties in trying to connect the two devices. At first, I was thinking to connect them directly through Bluetooth but then realized the limitation; they would have to be within a range of distance to stay connected [4]. After some research, I learned that connecting through cloud would be a better idea because now everyone has Wi-Fi in their house and it would be effective in reaching to the users since there was no limitation.

2.2. Setting up the connection between Firebase and the mobile app

It was not hard to just set up Firebase; however, it was challenging when I tried to have interacted connection between Firebase and the mobile app. To solve this problem, I did numerous researches on methods to link Android Studio and Firebase and learned about the usage of token during the process [5].

3. SOLUTIONS

As shown, the IoT Doorbell System consists of the following main aspects: Raspberry Pi, speaker and phone app installed on users' phone. In order to boost the whole system and connect the dots, every component should be able to interact with each other once the system is triggered (when someone rings the bell).

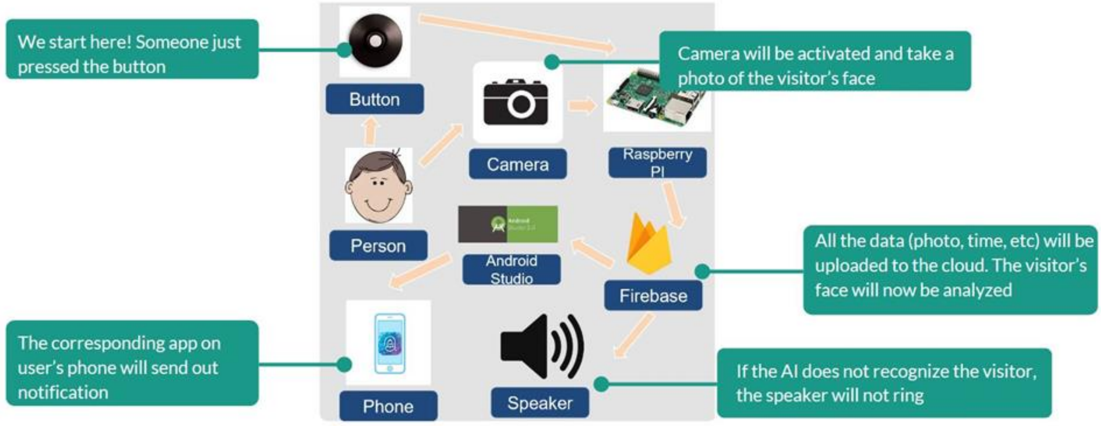
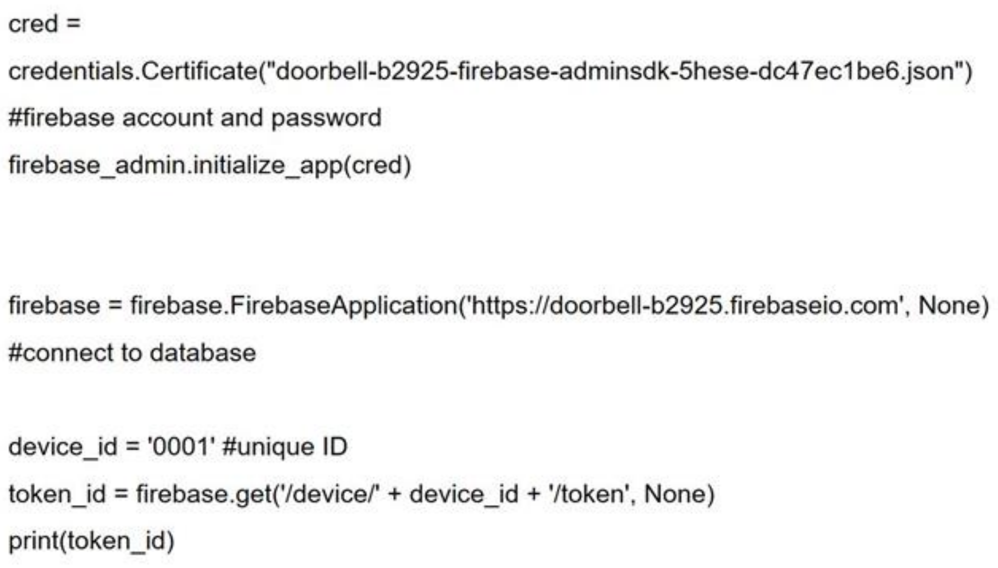

We start here! Someone just pressed the button
Button
Camera
Raspberry PI
Camera will be activated and take a photo of the visitor's face
Person
Android Studio
Firebase
All the data (photo, time, etc) will be uploaded to the cloud. The visitor's face will now be analyzed
The corresponding app on user's phone will send out notification
Phone
Speaker
If the AI does not recognize the visitor, the speaker will not ring

Figure 2: Component of IoT Doorbell System

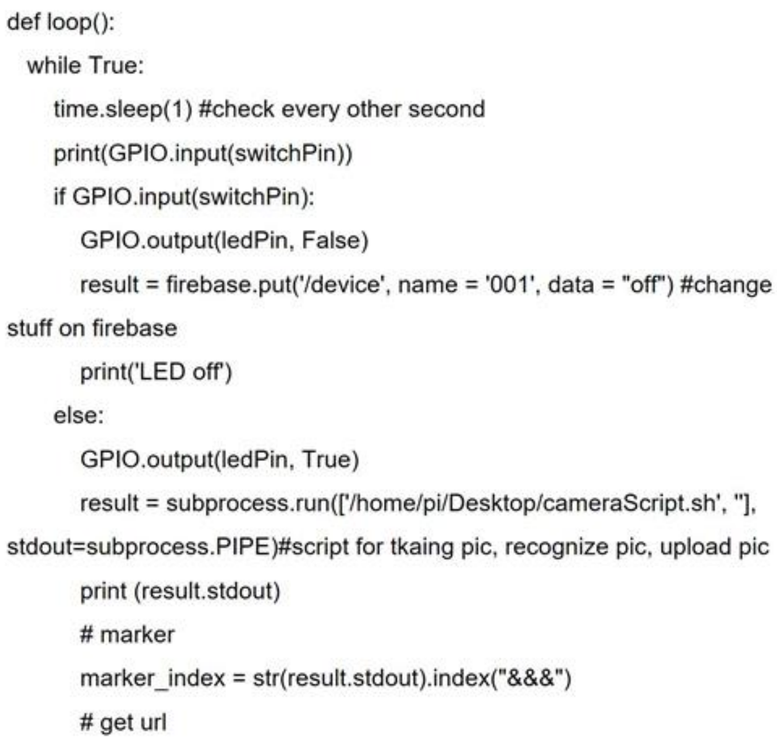
3.1. Connection Between Firebase and Raspberry Pie Through Wireless Local Area Networking

In order for the raspberry pi to be able to connect to users' phone in long distance, there should be a cloud in between. Thus, I set up a cloud on Firebase as well as its connection with the raspberry pi and users' phone.

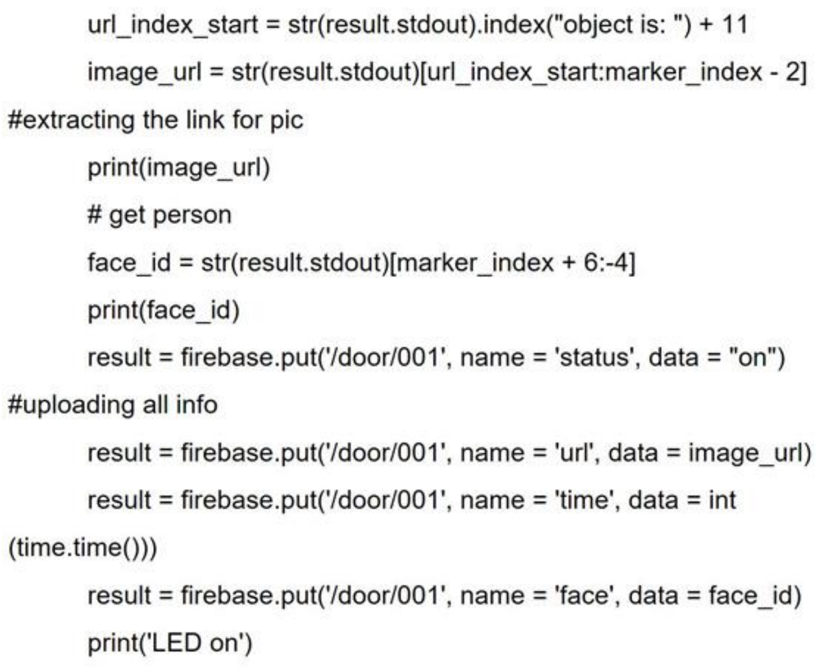

```
cred =
credentials.Certificate("doorbell-b2925-firebase-adminsdk-5hese-dc47ec1be6.json")
#firebase account and password
firebase_admin.initialize_app(cred)


firebase = firebase.FirebaseApplication('https://doorbell-b2925.firebaseio.com', None)
#connect to database

device_id = '0001' #unique ID
token_id = firebase.get('/device/' + device_id + '/token', None)
print(token_id)
```

Above is the code implemented to give the Raspberry Pi access for updating and receiving data on Firebase.

Following is the code for uploading visitors' pictures once the button is pressed:



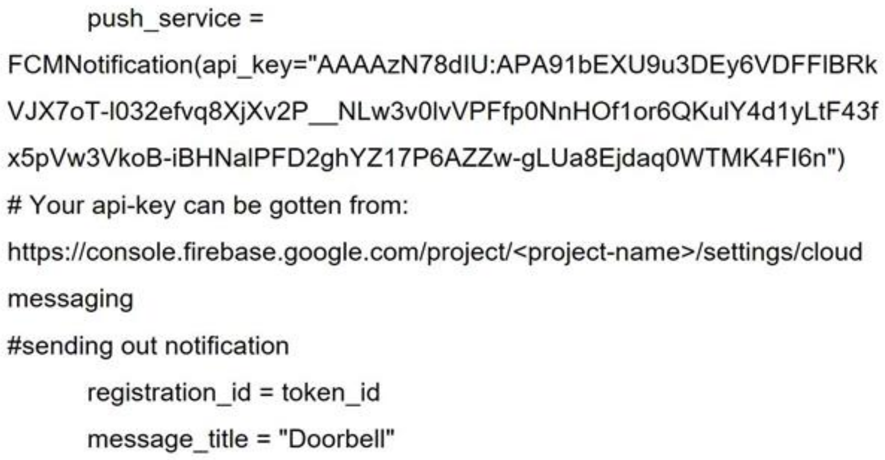
```
def loop():
  while True:
    time.sleep(1) #check every other second
    print(GPIO.input(switchPin))
    if GPIO.input(switchPin):
      GPIO.output(ledPin, False)
      result = firebase.put('/device', name = '001', data = "off") #change
stuff on firebase
      print('LED off')
    else:
      GPIO.output(ledPin, True)
      result = subprocess.run(['/home/pi/Desktop/cameraScript.sh', ''],
stdout=subprocess.PIPE)#script for tkaing pic, recognize pic, upload pic
      print (result.stdout)
      # marker
      marker_index = str(result.stdout).index("&&&")
      # get url
```



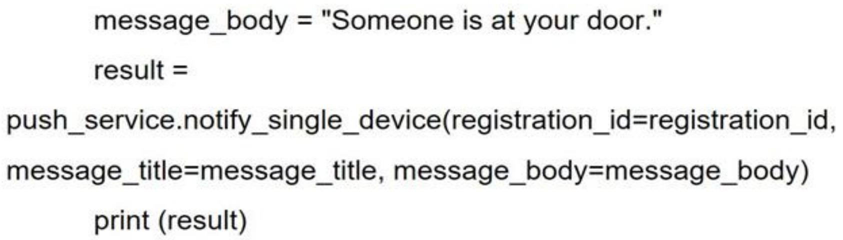
```
          url_index_start = str(result.stdout).index("object is: ") + 11
          image_url = str(result.stdout)[url_index_start:marker_index - 2]
#extracting the link for pic
          print(image_url)
          # get person
          face_id = str(result.stdout)[marker_index + 6:-4]
          print(face_id)
          result = firebase.put('/door/001', name = 'status', data = "on")
#uploading all info
          result = firebase.put('/door/001', name = 'url', data = image_url)
          result = firebase.put('/door/001', name = 'time', data = int
(time.time()))
          result = firebase.put('/door/001', name = 'face', data = face_id)
          print('LED on')
```

3.2. Connection between firebase and user's phone

To fully set up an IoT Doorbell System, users' phone should be connected to Firebase so that it can interact with the doorbell. As the following code is executed, both the visitor's photo and the prediction of his or her identity will be sent to users' phone, along with the visiting time and the message "Someone is at your door."



```
        push_service =
FCMNotification(api_key="AAAAzN78dIU:APA91bEXU9u3DEy6VDFFIBRk
VJX7oT-l032efvq8XjXv2P__NLw3v0lvVPFfp0NnHOf1or6QKulY4d1yLtF43f
x5pVw3VkoB-iBHNalPFD2ghYZ17P6AZZw-gLUa8Ejdaq0WTMK4FI6n")
# Your api-key can be gotten from:
https://console.firebase.google.com/project/<project-name>/settings/cloud
messaging
#sending out notification
        registration_id = token_id
        message_title = "Doorbell"
```



```
        message_body = "Someone is at your door."
        result =
push_service.notify_single_device(registration_id=registration_id,
message_title=message_title, message_body=message_body)
        print (result)
```

4. EMPIRICAL RESULTS

To evaluate the accuracy of facial recognition, I uploaded up to 24 different faces to the cloud and chose two time periods—8am and 8pm—for testing. In order to be able to compare the data, I conducted experiments to verify two aspects: the accuracy of facial recognition under the influence of number of stored faces on the cloud and the time of the day. The results are as following:

4.1. Accuracy of the Facial Recognition based on the number of faces stored on cloud

To test out the accuracy, I designed an experiment consisting of 10 trials for each face. Out of the 10 trials, the number of correct matching results were divided by 10 to calculate the accuracy.

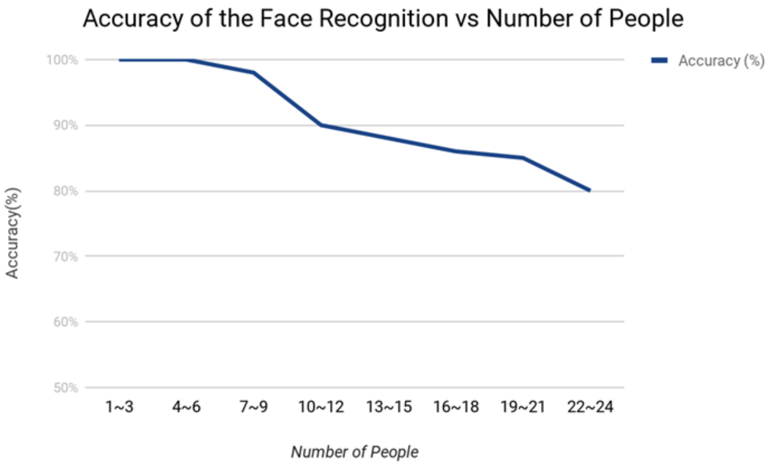


Figure 3: Facial recognition Result

Figure 3 illustrates the decreasing trend of the accuracy of facial recognition as the number of faces that are stored on the cloud. As can be seen, when there are fewer than 6 faces, the accuracy is 100%. As visitors' faces are added up to 9, the accuracy decreased by about 2%, still nearly perfect. However, when it was added to 12, a huge decrease in accuracy appeared. But this sharp decreasing trend seemed to grow slighter, as the accuracy reduced by about 4% from 10 to 21 people. When added 3 more people, the decreasing trend went steep again; the accuracy declined from 86% to 80%.

4.2. Accuracy of Facial Recognition based on time

The assessment of accuracy based on time was conducted as following: test out the 10 trials of each face at 8am and 8pm. Record the number of correct matching and then divide that by 10 to calculate the accuracy for each time period.

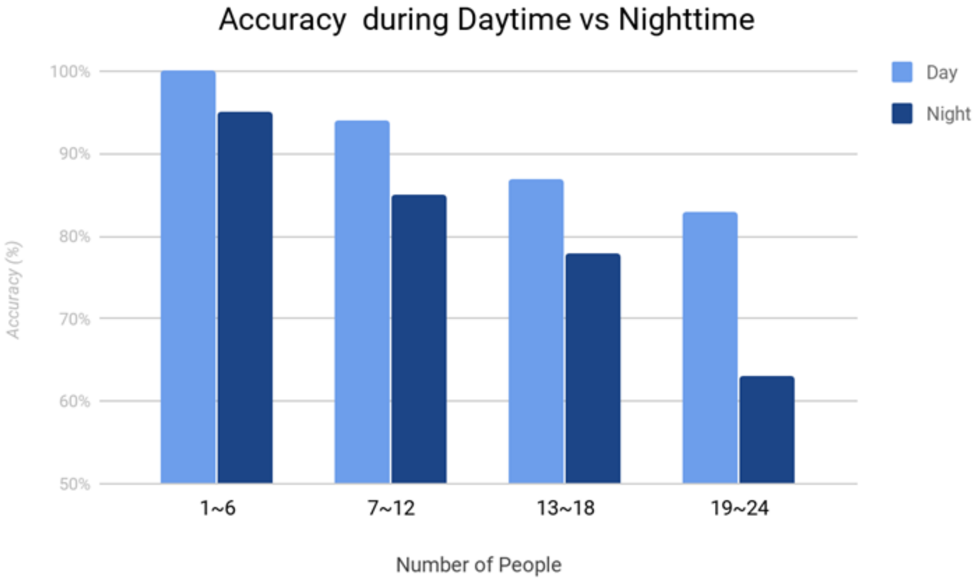


Figure 4: Accuracy During Daytime and Nighttime

Overall, figure illustrates a decreasing trend for both daytime and nighttime, while the nighttime accuracy has a steeper decline. As can be seen, when the number of faces on cloud is limited to 6, daytime can be 100% accurate, while nighttime accuracy already decreased about 5%. When number of faces is added up to 12, daytime accuracy started to reduce, with an accuracy of around 94%. Nighttime accuracy, however, decreased to about 85%. As added to 18 faces on the cloud, both daytime and nighttime experienced a steady decline, with an accuracy of 88% and 79% respectively. When added to 24 faces, daytime accuracy is about 82% and still kept a steady decreasing trend, while nighttime experienced a sharp decline to 64%.

5. RELATED WORK

Being a new emerging yet popular topic, deep learning in face recognition has acquired plenty studies. In PARKHI et al, the authors presented a deep learning approach in the application of face recognition using a convolutional neural network [6]. They also managed to maintain the accuracy in face recognition even with big data. Additionally, Turk et al employed a working, near-real-time face recognition, similar to the operation of IoT Doorbell System which relied on real time face recognition and notification sending [7]. In Robust Face Recognition via Sparse Representation, the authors pointed out the feature extraction and robustness to occlusion in a functioning deep learning-based face recognition [8]. In the project of IoT Doorbell System, I fulfilled both anticipation as the face recognition retained a high accuracy when the number of faces on the cloud remain in a small number.

6. CONCLUSION AND FUTURE WORK

In conclusion, this paper proposed an intelligent approach to the problem of anxiety brought by traditional doorbells utilizing aspects of machine learning. In addition, a mobile app has been developed for uses to receive notification from the IoT Doorbell. Currently, with the employment of deep learning in facial recognition and internet connection in long-distance notification to users, IoT Doorbell System effectively prevents unnecessary doorbell ringing from interrupting users' everyday life as an alternate for a traditional doorbell. For future work, I would focus on two aspects: optimization of the phone app and improvement on facial recognition accuracy. Though being able to send out notifications, the phone app contains only basic functions such as logging in and checking visitors' photos. Hence, I envision further enhancement on the variety of functions. For face recognition, I would boost its accuracy by trying other developing softwares such as Machine Box, etc.

REFERENCES

[1] Chung, Tae. "POTS: A Little-Known Cause of Extreme Fatigue." *John Hopkins Medicine*, 2019, www.hopkinsmedicine.org/health/conditions-and-diseases/pots-a-little-known-cause-of-extreme-fatigue. Accessed 13 July 2019.

[2] Rowan Medicine. "The 'Butterfly' Effect on Health." *Courier-Post*, The Courier-Post, 30 Apr. 2017, www.courierpostonline.com/story/sponsor-story/rowan-medicine/2017/04/30/butterfly-effect-health/100989648/. Accessed 13 July 2019.

[3] Williams, Alex. "How Anxiety Became Society's Prevailing Condition." *The Independent*, 17 June 2017, www.independent.co.uk/news/long_reads/anxiety-prozac-nation-depression-mental-health-disorder-america-panic-usa-memoirs-self-help-book-a7785351.html. Accessed 13 July 2019.

[4] Shorey, R., and B.A. Miller. "The Bluetooth Technology: Merits and Limitations." *2000 IEEE International Conference on Personal Wireless Communications. Conference Proceedings (Cat. No.00TH8488)*, 2000, 10.1109/icpwc.2000.905777.

[5] “Read and Write Data on Android | Firebase Realtime Database | Firebase.” *Firebase*, 2019, firebase.google.com/docs/database/android/read-and-write. Accessed 13 July 2019.

[6] Parkhi, Omkar, et al. *Deep Face Recognition*. 2015.

[7] Turk, Matthew, and Alex Pentland. *Face Recognition Using Eigenfaces*. 1991.

[8] Zhang, Haichao, et al. “Pose-Robust Face Recognition via Sparse Representation.” *Pattern Recognition*, vol. 46, no. 5, May 2013, pp. 1511–1521, 10.1016/j.patcog.2012.10.025. Accessed 13 July 2019.

[9] Lee, Kwang, and Jae Choi. “Remote-Controlled Home Automation System via Bluetooth Home Network - IEEE Conference Publication.” *Ieee.Org*, 2019, ieeexplore.ieee.org/abstract/document/1323826. Accessed 14 July 2019.

[10] Sun, Yi, et al. “DeepID3: Face Recognition with Very Deep Neural Networks.” *ArXiv.Org*, 2015, arxiv.org/abs/1502.00873. Accessed 14 July 2019.

[11] LeCun, Yann, et al. “Deep Learning.” *Nature*, vol. 521, no. 7553, May 2015, pp. 436–444, www.nature.com/articles/nature14539, 10.1038/nature14539.

[12] Schmidhuber, Jürgen. “Deep Learning in Neural Networks: An Overview.” *Neural Networks*, vol. 61, Jan. 2015, pp. 85–117, www.sciencedirect.com/science/article/pii/S0893608014002135, 10.1016/j.neunet.2014.09.003.

[13] Deng, Li. “Deep Learning: Methods and Applications.” *Foundations and Trends® in Signal Processing*, vol. 7, no. 3–4, 2014, pp. 197–387, www.nowpublishers.com/article/Details/SIG-039, 10.1561/2000000039. Accessed 14 July 2019.

[14] Lin, Shang-Hung. “Face Recognition/Detection by Probabilistic Decision-Based Neural Network - IEEE Journals & Magazine.” *Ieee.Org*, 2019, ieeexplore.ieee.org/abstract/document/554196. Accessed 14 July 2019.

[15] Lu, Huimin, et al. “Brain Intelligence: Go beyond Artificial Intelligence.” *Mobile Networks and Applications*, vol. 23, no. 2, 21 Sept. 2017, pp. 368–375, 10.1007/s11036-017-0932-8. Accessed 14 July 2019.