DOOR LOCK SYSTEM FOR ZENSE RECRUITMENT

Name: Krutik Patel

Email: krutik.patel@iiitb.ac.in

1. What is the idea?

This project takes a users face details and detects it using deep learning. Also there is a fun pygame app that takes in your username and detects your face, if the face detected is of username entered, then it unlocks the door. Since you are the owner of your home, you will use the app as your home, with your face being the key.

2. How to start?

Below is the brief view on how to start...

2. 1 Libraries you will need

1. Tensorflow
2. Keras (although it is included with tensorflow 2 onwards)
3. OpenCV
4. pickle
5. pygame
6. numpy
7. pillow

Make sure that you install all the libraries prior to running the program

2.2 Taking user's images for learning

Run the capture.py file, enter your username in the terminal and let the window open. The window will detect a face and a coloured box will appear around your face. It will take 100 pictures of your face in grayscale and save them in png format. All the photos will be stored in the images directory in appropriate username's folder. If you wish to close the window, press 'q' key on the keyboard. If the webcam is not starting, anti-virus might be blocking your cam.

2.3 Training images

1. Once more than enough people have been captured, you can go on with training the model.
2. Run the train.py python file.

3. It may take some time (the time depends on the number of people who have been captured).
4. Check with the accuracy, if you feel the accuray is less, you may run the file again. But beware, not to train the model too many times, or the model may become overfitting.

Sample images is already included in the images folder. If you dont want them, delete all the folders. But make sure that the train.py is run on more than 2 users, else you might face some issues while detecting your face, talked in the following section.

2.4 Checking the detection

1. Once you are satisfied with the training, you can move on with trial on the detect.py app.
2. Run the detect.py file
3. A window similar to capture.py will appear.
4. Try detecting, your face, your username will be displayed above the blue box

2.5 DOOR LOCK SYSTEM

1. Since you are the owner of your home, you will use the app as your home, with your face being the key.
2. If your face is detected, properly, you can start with the door.py app.
3. When you run the file, a pygame window will appear. Do as it says. That is, you will have to input your username (IMP: your username must be similar to the one used in images section. If the images folder name is "faulty_nut" put faulty_nut as username), else you might face problems while unlocking the door.
4. After you successfully enter your name, press Space to open the detection app. When the app detects, it finds the name of that person. If that is the username entered, then Viola! THE DOOR UNLOCKS.

To exit the app, press the close button. To exit the detection app, press the 'q' key in the keyboard.

3. How to use (after the setup)?

Suppose you are the owner of the house, you will have your username.

1) Run the door.py app
2) A pygame window will appear with a closed door, and a text prompting you to enter username. Enter your username(the same name that your corresponding images folder has)
3) When you enter your name, you will be asked to press Space to detect your face.
4) Once your face is detected, and if it matches with the username, the door unlocks.

4. What can it used for...

The complexity of the program can be increased even further with you keeping a hand gesture as a password, and the door only opens if your faces is authorized and the password is correct. It can be implemented in real life scenario, using the detection app, you can either start your tv, ac or even lights, once the app detects that the owner of the house is within range.

5. What problems did I face while making this...

Well, there were many problems that I faced, while making this project.

1. My anti virus blocked my webcam from starting. Had to figure out what was causing the issue.
2. Dataset got corrupted, while implementing the deep learning model in jupyter notebook. I had to start all over again.
3. When you import the images to the python file in train.py, they are imported as numpy array. But while passing the image to deep learning model, errors orrcured, saying it wanted dim_n = 4, but only 3 were passed. Turns out, even in grayscale image, the individual pixel has to be passed as an array of 1.