

Two-Factor Authentication for ATM System

The Facial Recognition System that is implemented in this project will integrate a facial recognition module to the existing ATM cash withdrawal process thus adding a 2-step verification alongside the pin authentication module. The customers will need to go through a facial recognition phase as they simultaneously go through the pin authentication process. In this facial recognition module, we have proposed Inception-ResNet V1 as the machine learning CNN model to identify faces of the bank's customers. The model has been initially trained on sample faces to capture the characteristics of a human face and subsequently trained on the LFW dataset with more than two hundred classes thereby providing better accuracy.

Clone the GitHub repository

git clone https://github.com/Big-Data-And-Data-Analytics/Face_PIN_2step_Authen.git

Setup the repository

Step 1:

Install all the necessary packages using [requirements.txt](#)

Step 1.a:

Follow the steps present in [/Documents/command to install requirements.txt](#) to install all the necessary python packages using requirements.txt

Step 2

Start the application by running the python file [gui.py](#)

About the repository

1. The training dataset contains over 20000 images, belonging to 200+ classes. To avoid unbiased results, each class contains 100 images.
[Face PIN 2step Authen/aligned image data/](#)
2. The testing dataset contains over 4000 images, belonging to 200+ classes. To avoid unbiased results, each class contains 20 images.
[Face PIN 2step Authen/test images/](#)
3. The username and PIN are stored in a JSON file, this file acts a dummy database.
[Face PIN 2step Authen/File/userCredentials.json](#)

4. The model can be evaluated using [evaluation.py](#)
We have achieved a model accuracy of **89.60%**

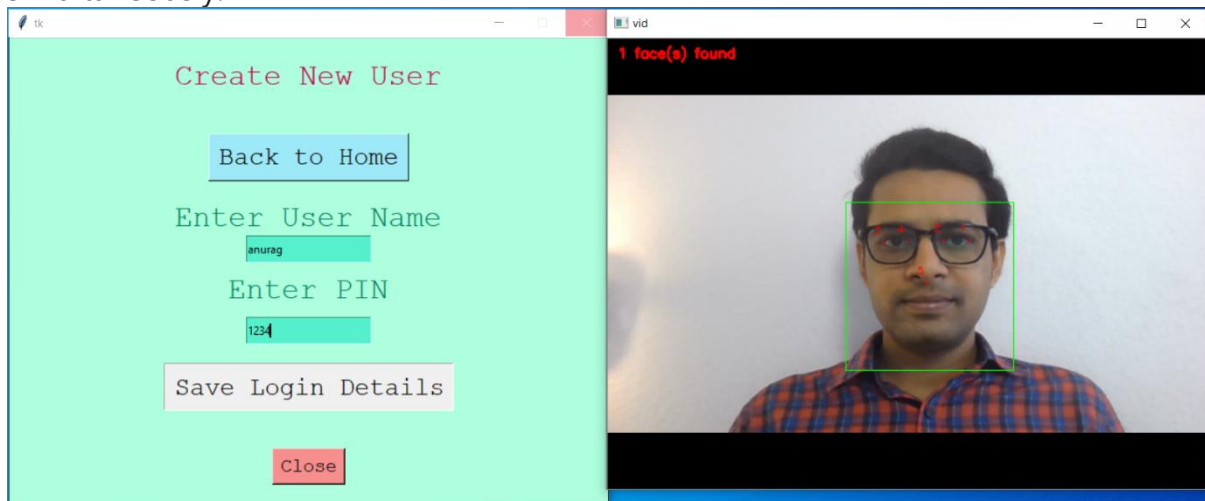
```
Run: evaluation x
recognized image: Zinedine_Zidane
**Image 4100 of 4100**
folder name: Zinedine_Zidane
**Loading Encodings...**
**Recognising faces...**
recognized image: Zinedine_Zidane
overall accuracy: 0.8960975609756098

Process finished with exit code 0
```

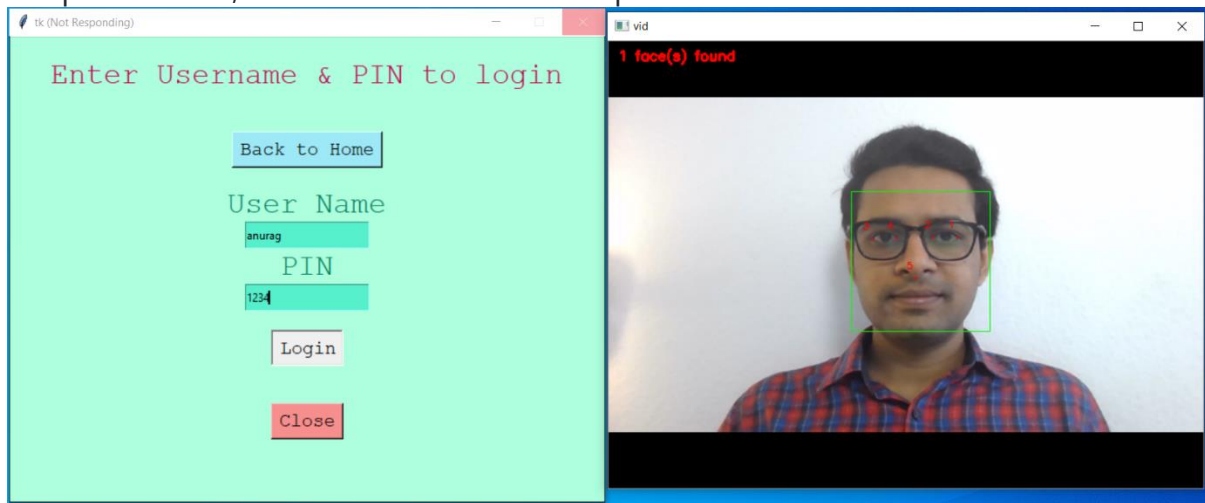
About User Interface

The application provides two options

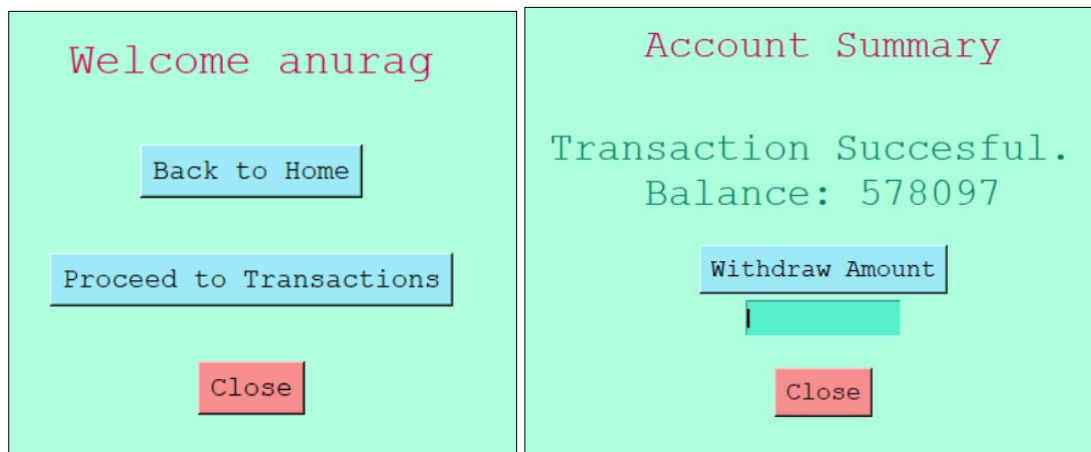
1. Create New User
 2. Existing User
1. To create a new user, username and PIN is required. When the user clicks on "save login details", two threads are started. One thread saves the login credentials into the database and second thread capture the user image. Login credentials and image, both are stored simultaneously.



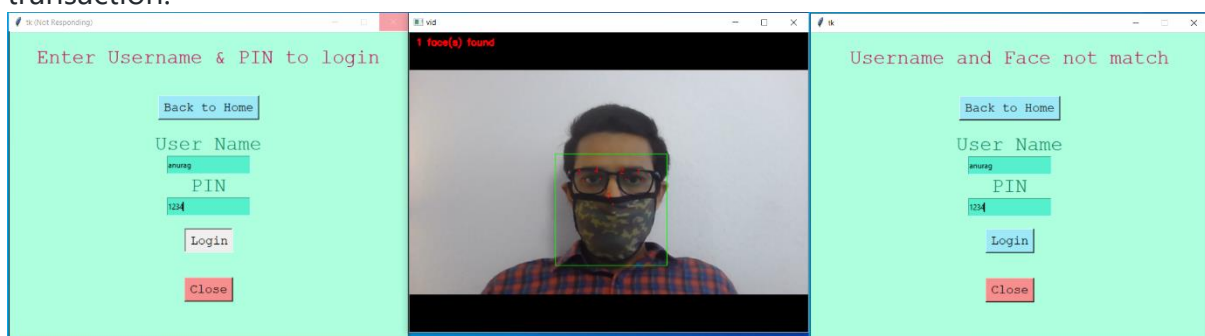
- While accessing as an existing user, the user must enter his username and PIN. When the user clicks on "Login" button, two threads are started. One thread verifies the username and PIN, another thread verifies the image. Both the processes happen simultaneously. If anyone the process fails, the user is not allowed to perform the transaction.



When the user credentials and face are of same user, then only the user can perform the transaction.



- If the username & PIN does not match with the face, the user is not allowed to perform transaction.



Architecture Used

