



face.ai

face.ai

User Guide Manual

Rocket Systems

https://www.youtube.com/channel/UC9-b_6KfZ2NDLBF5A8ZECmA

mailrocketsystems@gmail.com

Contents

1. Introduction
2. Registration & Licensing
3. Gallery
4. Enroll
5. Recognize
6. Server
7. Feedback
8. Settings

1. Introduction

Face.ai is a UI application which provides complete solution to face recognition services. This application is designed in such a way that it is very easy to use and can also be integrated with any other software application. This consist of creating infinite number of galleries, adding infinite number of persons and then recognizing them. Recognition or enrollment can be done using photos or live stream from inbuilt camera or external USB camera. Thus, it provides end to end solution for face recognition.

This software also has inbuilt API server which provides a list of face APIs. These APIs can also be integrated with any other application. Thus, making it compatible with all other software and applications. To get the list of APIs, refer to the page Server.

This document aims to be a step by step guide on how to use the face.ai application. Although the application is very easy to use but it is recommended to go through this document before using the application. In further sections, this guide will provide explanation on:

How to create gallery

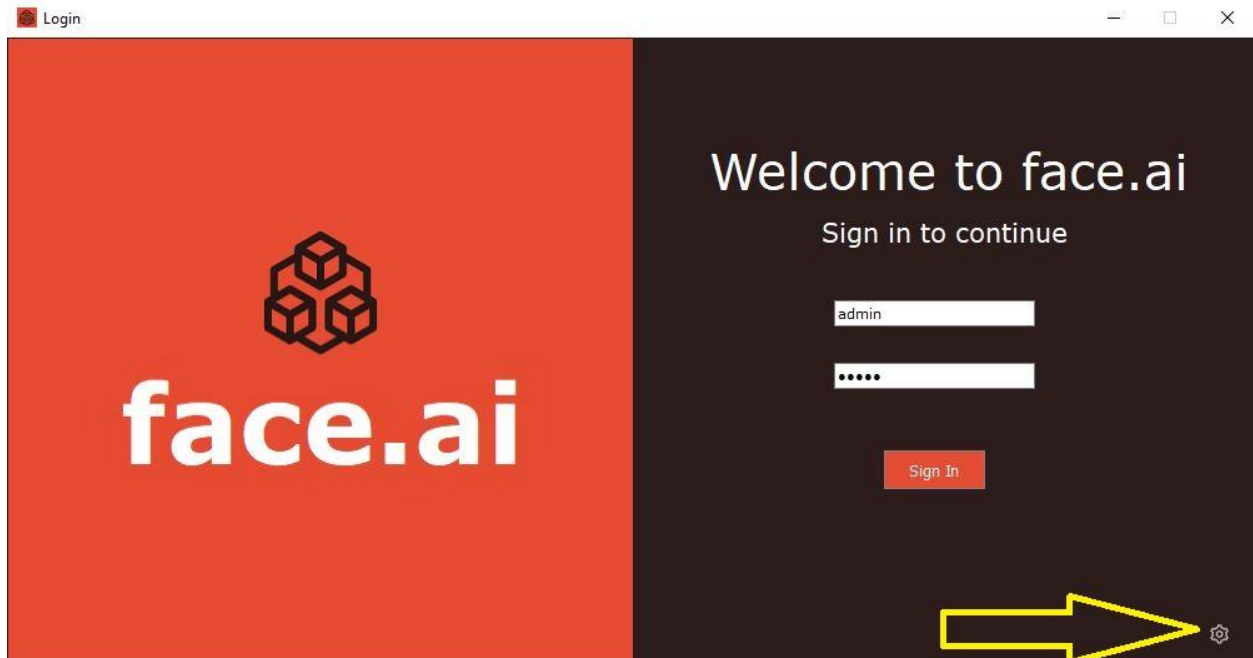
How to enroll person

How to recognize person

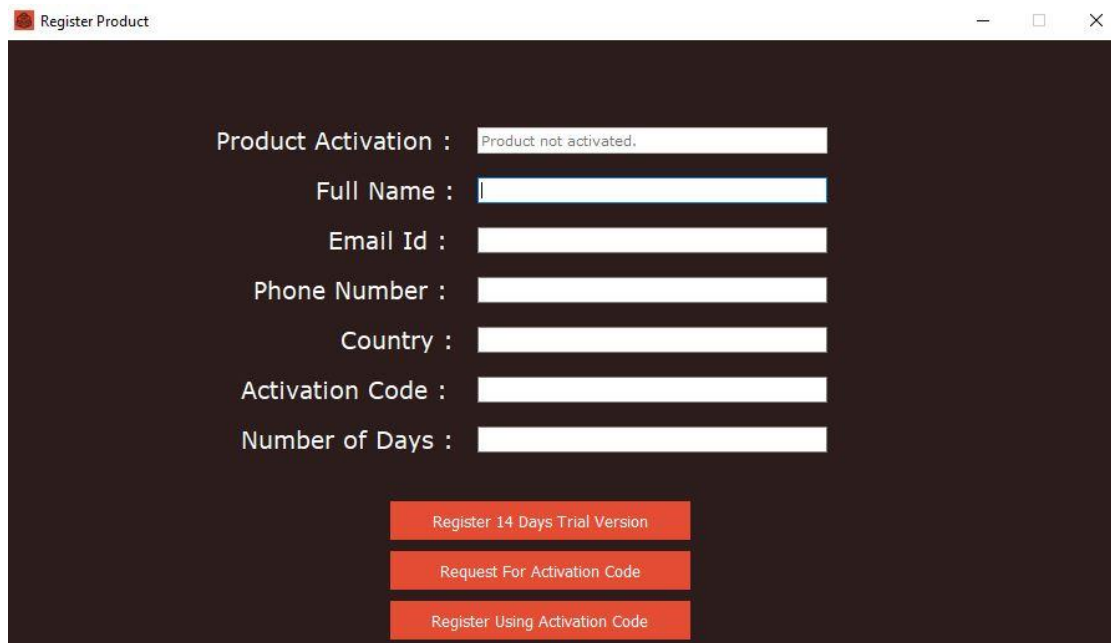
NOTE: Always run the application in administrative mode

2. Registration & Licensing

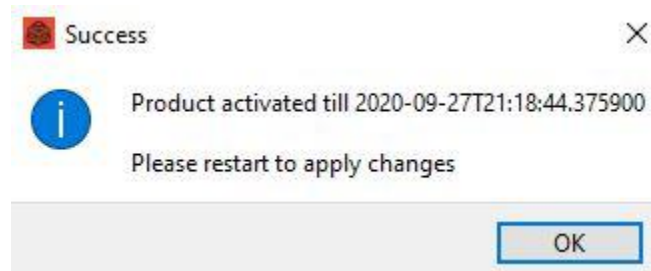
Application needs to be registered first before user can use it. Until the registration has been done, user won't be able to login. Click below button to open the registration window.



Once user click on the button, it will show below window where we can see that it says product not activated.



User need to activate the product first by clicking on button **Register 14 Days Trial Version** and filling the required details. This will activate the product for 14 days trial version.

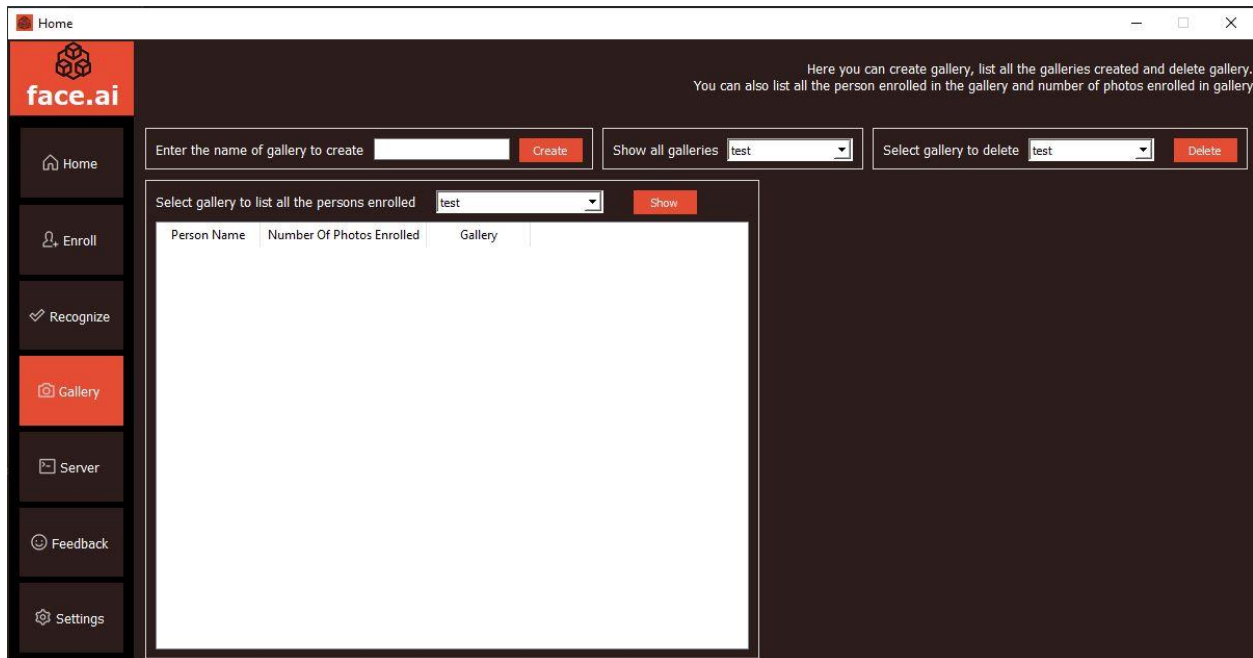


After 14days, user will have to request for activation code by clicking on **Request Activation Code** depending upon how many days user needs the product to be activated. Activation code will be shared over email and user can enter the activation code and click on **Register Using Activation Code** to apply the changes.

Default username and password which comes with the application is admin & admin. Once user has registered the product, use default username and password to login into home window. Username and password can later be changed by going into settings.

3. Gallery

User need to create gallery before enrollment and recognition. All the photos of the persons which are enrolled, are saved in a gallery and thus gallery is required to be created first. User can create a gallery by click on gallery button page first which will show below page.



The screenshot displays the 'face.ai' web application interface. On the left is a dark sidebar with navigation buttons: Home, Enroll, Recognize, Gallery (highlighted in red), Server, Feedback, and Settings. The main content area has a dark background. At the top, it says 'Here you can create gallery, list all the galleries created and delete gallery. You can also list all the person enrolled in the gallery and number of photos enrolled in gallery'. Below this are three controls: 'Enter the name of gallery to create' with a text input and a red 'Create' button; 'Show all galleries' with a dropdown menu showing 'test' and a red 'Show' button; and 'Select gallery to delete' with a dropdown menu showing 'test' and a red 'Delete' button. In the center, there is a table titled 'Select gallery to list all the persons enrolled' with a dropdown menu showing 'test' and a red 'Show' button. The table has three columns: 'Person Name', 'Number Of Photos Enrolled', and 'Gallery'. The table body is currently empty.

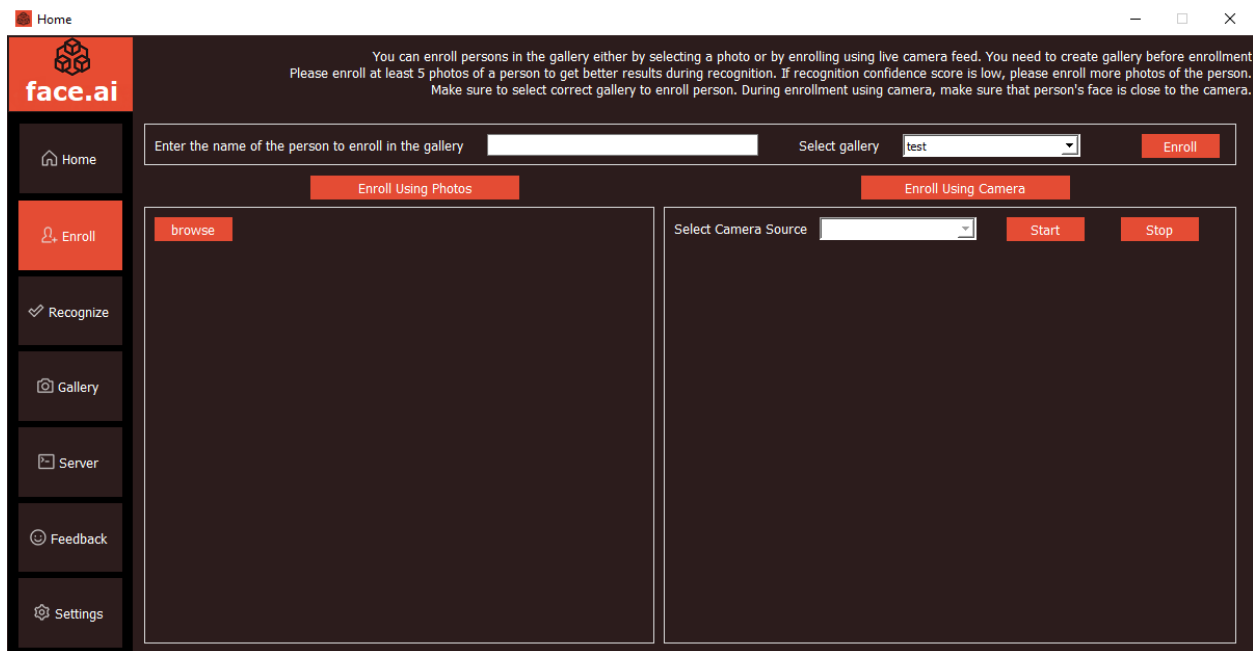
User can create gallery by specifying the name of the gallery and then clicking on Create button. It can also list the number of galleries created. User can also delete gallery by selecting the drop-down menu and click Delete button. This page also shows the list of the persons which have been enrolled in the gallery.

User can add as many galleries as required and can add as many persons in the gallery.

Note: This application does not save any image of the person. Only face data is saved as binary format

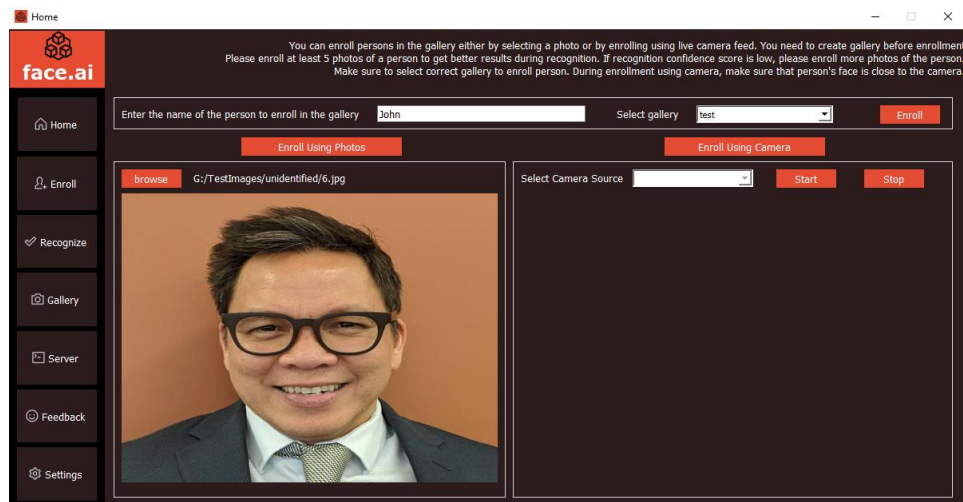
4. Enrollment

After the creation of the gallery, user can now proceed on to enrolling persons into the created gallery. Click on Gallery to open the gallery page which will look like below:



Before enrolling the person, user need mention the name of the person and select the appropriate gallery in which the person will be enrolled. Enrollment can be done in two ways either using photos or using live stream from camera.

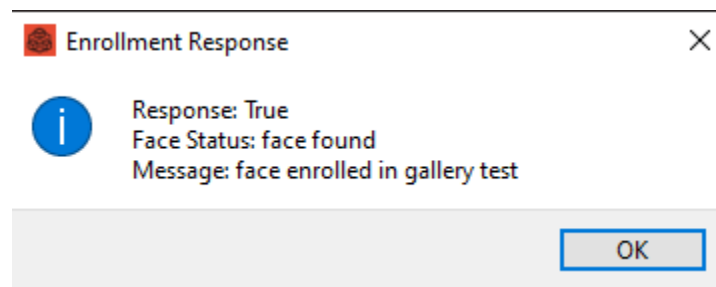
Click on Enroll Using Photos button to activate it. Then select browse button to browse for the photo. Once the photo is selected it will be shown on the application. Next click on Enroll button on top right corner to enroll the person.



If you want to enroll person using live camera stream, you can click on Enroll Using Camera button to activate it and then from the camera source drop down menu, select either the inbuilt webcam or external USB camera. When enrolling using camera, follow below guidelines:

1. Person need to look straight into the camera
2. Make sure face is not too far from the camera
3. Only one person should enroll at a time.
4. Please enroll at least 5 photos of the person. You can enroll more photos of the person, if the recognition results are not good.
5. If you want the application to recognize person from the side face, make sure to enroll 2-3 photos of the person as side face.

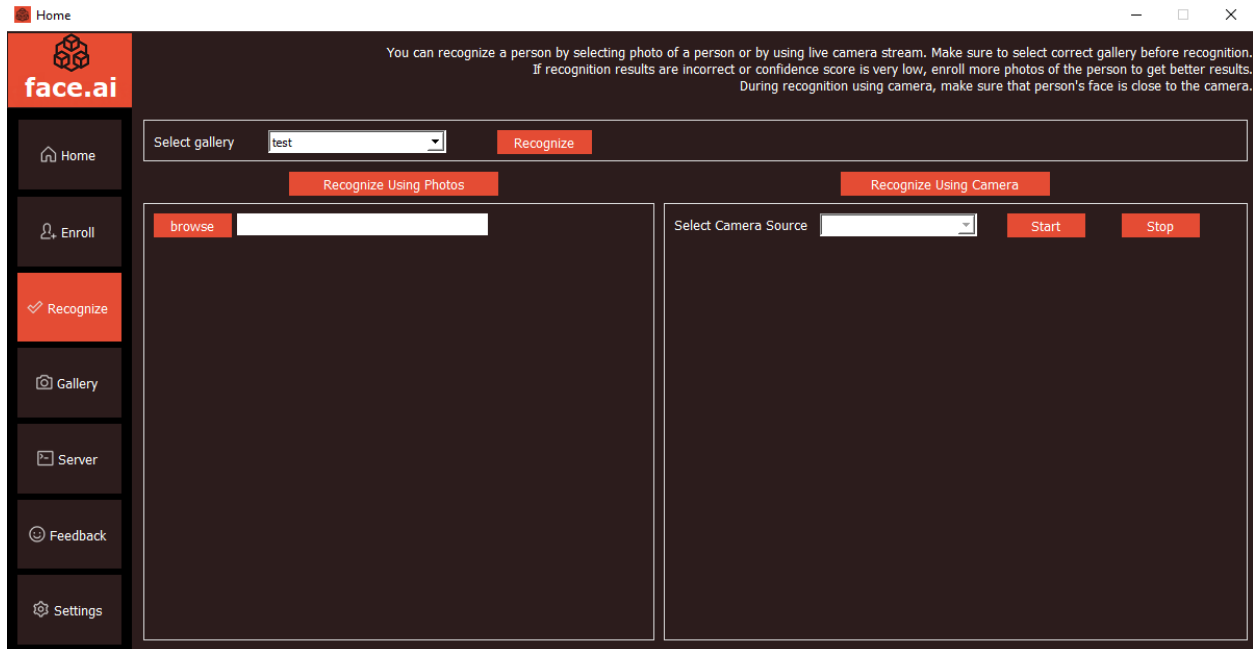
Once enrollment is done, below message will show the confirmation



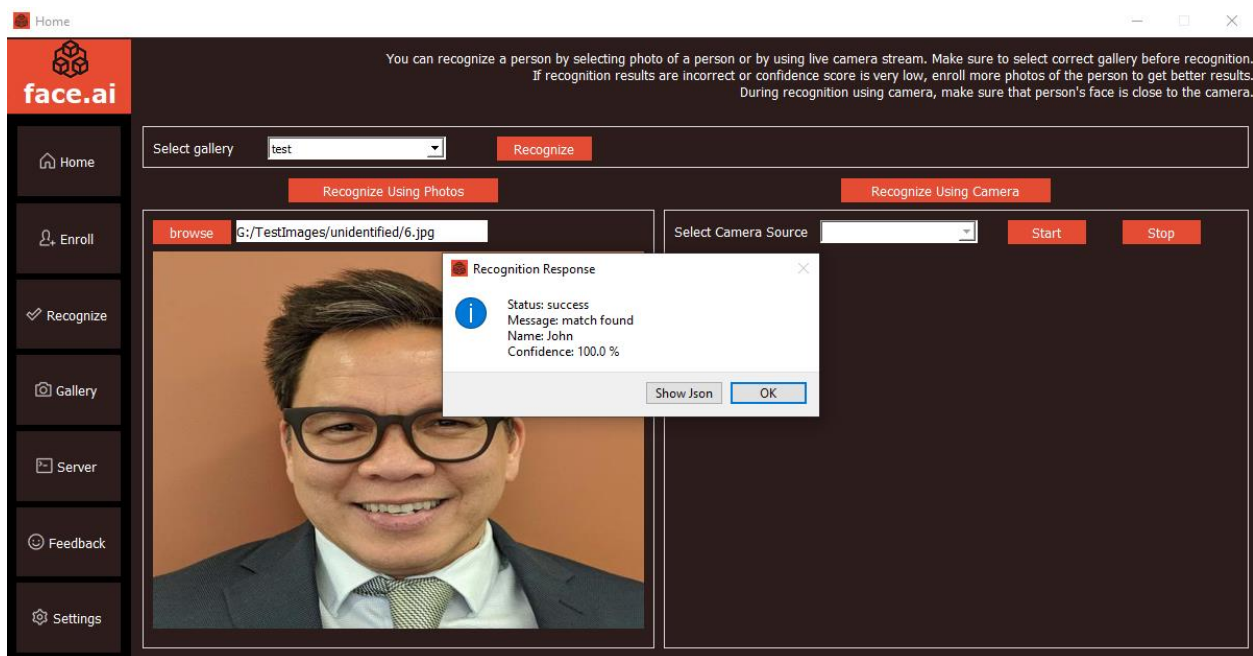
This means that person has been enrolled in the gallery. When you have enrolled all the persons, you can go into the gallery page and can list all the person added into the gallery.

5. Recognition

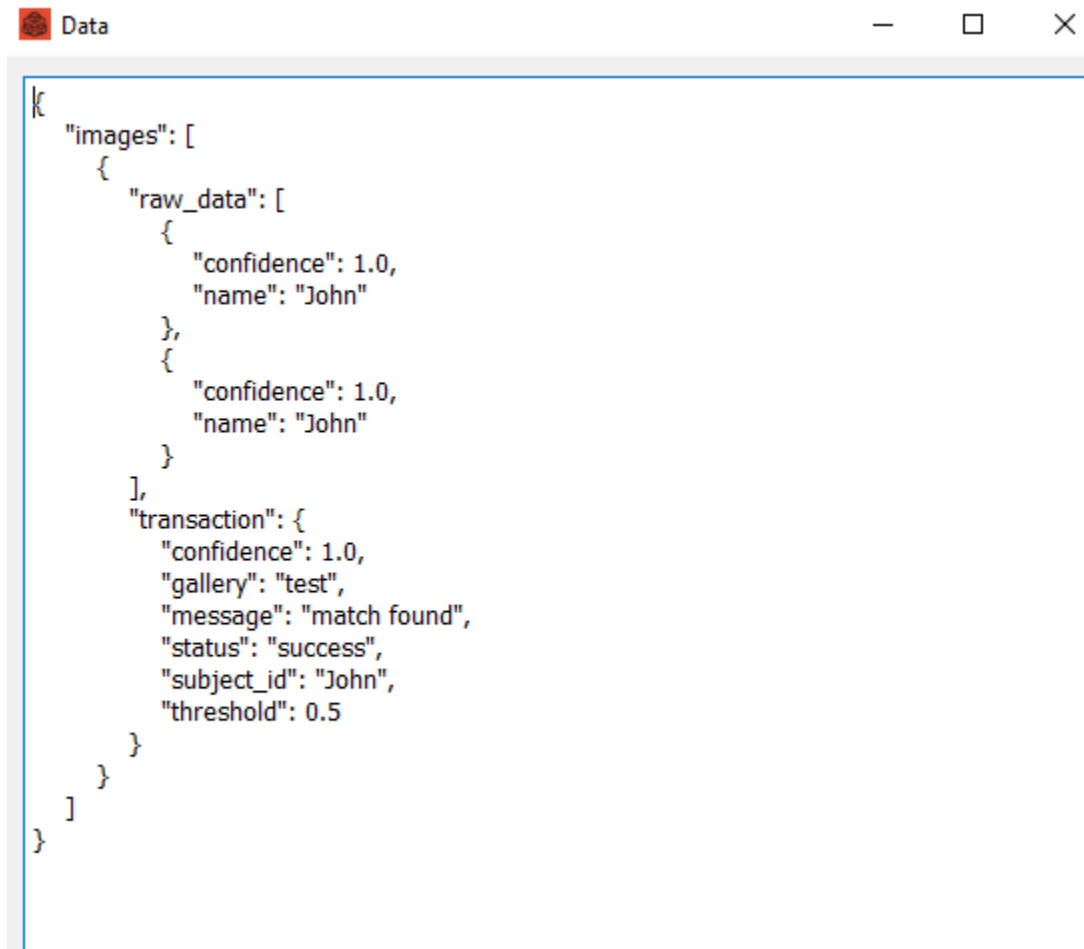
Once all the persons have been enrolled, its time to recognize them. Click on Recognition to open recognition page



First select the appropriate gallery. Then select if you want to recognize person using photos or from the camera stream. Once you have selected the photo or camera stream, click Recognize button to recognize the person. It will show below result



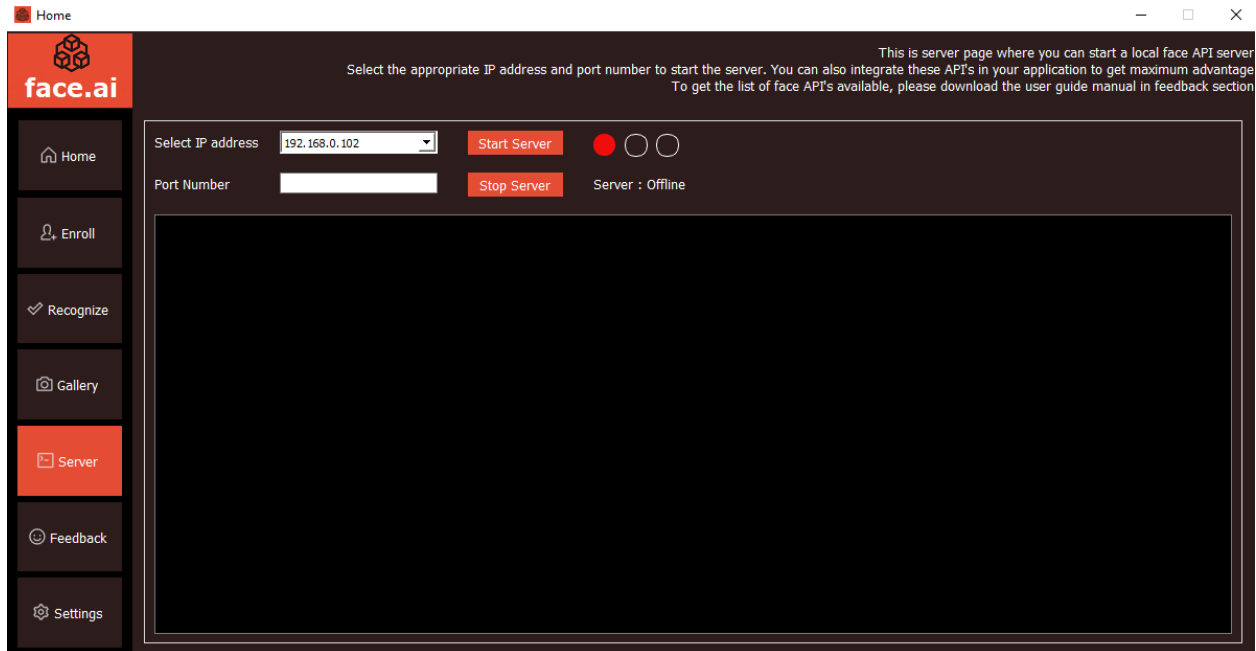
Here you can see that we have got 100% recognition confidence with name as John. This means our recognition has worked fine. If you are getting low confidence or false recognition results, then you need to enroll more photos of the person. You can also click on Show Json button to get the full json response from the face APIs



```
{
  "images": [
    {
      "raw_data": [
        {
          "confidence": 1.0,
          "name": "John"
        },
        {
          "confidence": 1.0,
          "name": "John"
        }
      ]
    },
    {
      "transaction": {
        "confidence": 1.0,
        "gallery": "test",
        "message": "match found",
        "status": "success",
        "subject_id": "John",
        "threshold": 0.5
      }
    }
  ]
}
```

6. Server

If you want to integrate this application, you can start the inbuilt server by clicking on Server button which will show below page



To start the server, you need select the IP address from the drop-down menu and also enter the port number. Click on Start Server button to start the server. Once the server is started, light will turn green and status will change to Online. You can stop the server by clicking on Stop Server button.

For example, the selected IP address is 192.168.0.102 and Port number is 80. Then below is the list of APIs available

Use below authentication method to authenticate APIs except status API:

Pass below variable as headers. Use username and password which is required for login

app_name: <username>

app_key: <password>

1. Server Status API

URL: <http://192.168.0.102:80/api/status>

Method: GET

Authentication: None

Response: {

 "date": "2020-09-14T09:19:45.149091",

 "msg": "server running",

 "status": true

}

2. Create Gallery API

URL: http://192.168.0.102:80/api/face/create_gallery

Method: POST

Authentication: as described above

Json Request: {

 "gallery_name": "test"

}

Response: {

 "message": "gallery created test",

 "status": "true"

}

3. Enroll API:

URL: <http://192.168.0.102:80/api/face/enroll>

Method: POST

Authentication: as described above

Json Request: {

```
"image": "<image_base64_url>",  
"subject_id": "<subject_id>",  
"gallery_name": "<gallery_name>"
```

}

Convert you image into base64 url and paste it as value from image. Use subject_id as name of the person.

Response: {

```
"message": "face enrolled in gallery test",  
"status": true
```

}

4. Recognize API

URL: <http://192.168.0.102:80/api/face/recognize>

Method: POST

Authentication: as described above

Json Request: {

```
"image": "<image_base64_url>",  
"gallery_name": "<gallery_name>"
```

}

Response:

{

```
"images": [  
  {
```

```
    {
```

```
      "raw_data": [  
        {
```

```
          {
```

```
            "confidence": 1.0,
```

```
      "name": "John"
    },
    {
      "confidence": 0.86771,
      "name": "John"
    },
    {
      "confidence": 0.86771,
      "name": "John"
    }
  ],
  "transaction": {
    "confidence": 1.0,
    "message": "match found",
    "gallery": "test",
    "status": "success",
    "threshold": 0.5,
    "subject_id": "John"
  }
}
```

In above response, the data under transaction consist of the final data.

5. View Gallery

URL: http://192.168.0.102:80/api/face/view_gallery

Method: POST

Authentication: as described above

Json Request: {

```
    "gallery_name": "test"
}
```

Response:

```
{
  "subject_id": [
    {
      "John": 3,
      "Andrew": 5
    }
  ]
}
```

This API will list all the persons added into the gallery along with the number of photos used per person.

6. Delete Gallery

URL: http://192.168.0.102:80/api/face/delete_gallery

Method: POST

Authentication: as described above

Json Request: {

```
    "gallery_name": "test"
}
```

```
Response: {  
  "msg": "gallery has been deleted successfully"  
}
```

7. List Galleries

URL: http://192.168.0.102:80/api/face/list_gallery

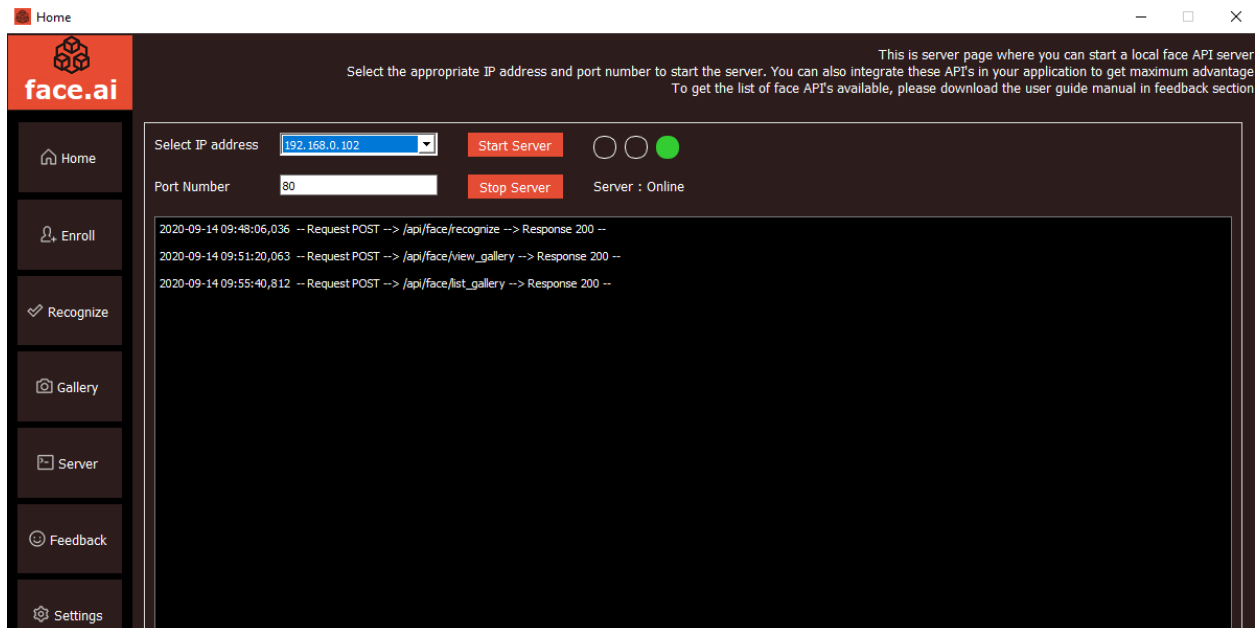
Method: GET

Authentication: as described above

```
Response: [  
  "test",  
  "test2"  
]
```

This API list all the galleries

All the logs of the APIs are shown on the server page itself.



7. Feedback & Support

If you have any questions related to the software, you can use the feedback options to give us the feedback. You can also directly mail us at mailrocketsystems@gmail.com

Home

face.ai

Please rate us or you can ask any query

Home

Enroll

Recognize

Gallery

Server

Feedback


Settings


Give us your valuable feedback or if you have any queires, you can ask them here


Full Name :


Email Id :

Please Rate Us :

☐ 

☐ 

☐ 

☐ 

Very Bad

Bad

Good

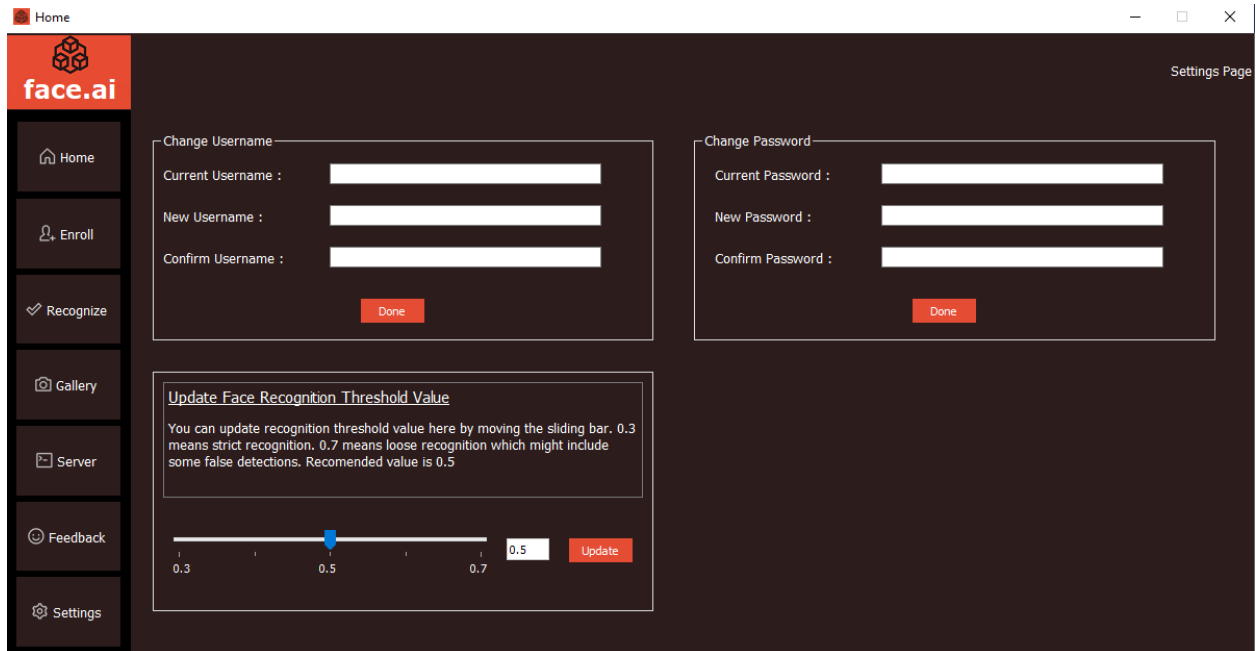
Very Good

Comments :

Submit

8. Settings

User can change the username and password in the settings option. This will also change the authentication required to authenticate server APIs.



The screenshot shows the 'face.ai' Settings Page. On the left is a sidebar with navigation links: Home, Enroll, Recognize, Gallery, Server, Feedback, and Settings. The main content area has three sections:

- Change Username:** Includes input fields for 'Current Username', 'New Username', and 'Confirm Username', with a 'Done' button below.
- Change Password:** Includes input fields for 'Current Password', 'New Password', and 'Confirm Password', with a 'Done' button below.
- Update Face Recognition Threshold Value:** Contains a text box explaining that the threshold can be adjusted via a slider from 0.3 (strict) to 0.7 (loose), with 0.5 as the recommended value. Below the text is a slider bar with a blue handle at 0.5, and an 'Update' button.

On this page, you can also change the recognition threshold value. Recommended value is 0.5 and the range starts from 0.3 being strict recognition and 0.7 being loose recognition.

By default, it comes with 0.5 but if you think recognition results are not correct, you can decrease the threshold by adjust the bar.