- ***University**:* JNU
- ***Department**:* Computer Science and Technology
- ***Course**:* Human-Computer Interaction
- ***Project Title**:* GitHub Project (Facial detection and emotion recognition)
- ***Project Part**:* Part 4
- ***Author**:* 蒋云翔 2022102330（Yunxiang Jiang）（Accomplish the task by myself only）
- ***Instructor**:* 龙锦益（Jinyi Long）
- ***Date**:* December 4, 2024

Catalogue

1. Brief Introduction to GitHub project

1.1 Website URL:

serengil/deepface: A Lightweight Face Recognition and Facial Attribute Analysis (Age, Gender, Emotion and Race) Library for Python

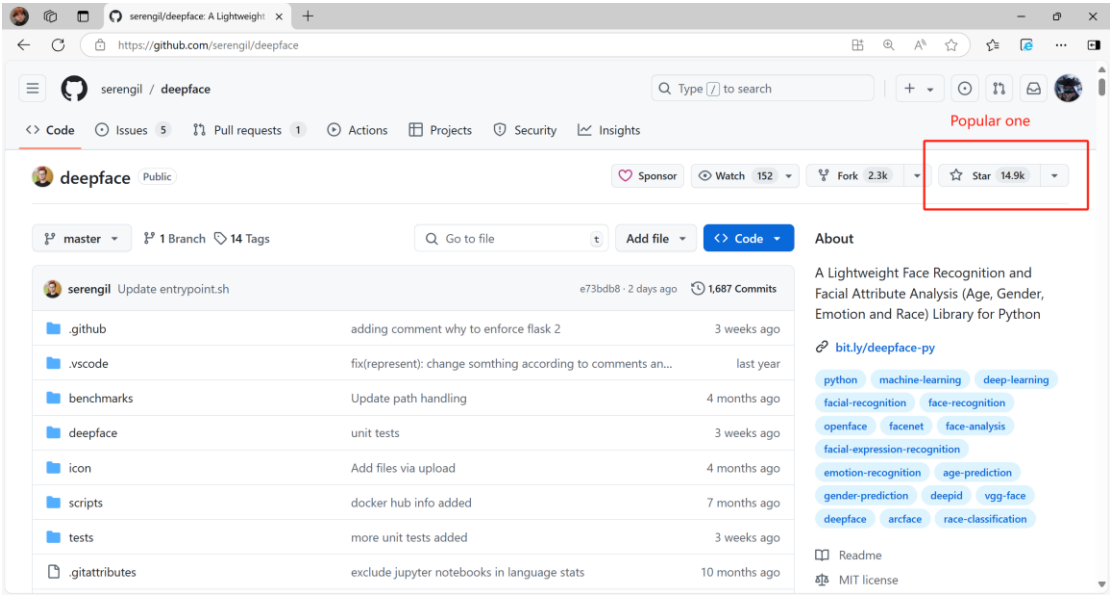


Figure 1: GitHub URL link

1.2 Project's brief introduction

DeepFace is a lightweight face recognition and facial attribute analysis (age, gender, emotion and race) framework for python. It is a hybrid face recognition framework wrapping state-of-the-art models: ***VGG-Face, FaceNet, OpenFace, DeepFace, DeepID, ArcFace, Dlib, SFace and GhostFaceNet***.

Experiments show that human beings have ***97.53%*** accuracy on facial recognition tasks whereas those models already reached and passed that accuracy level.

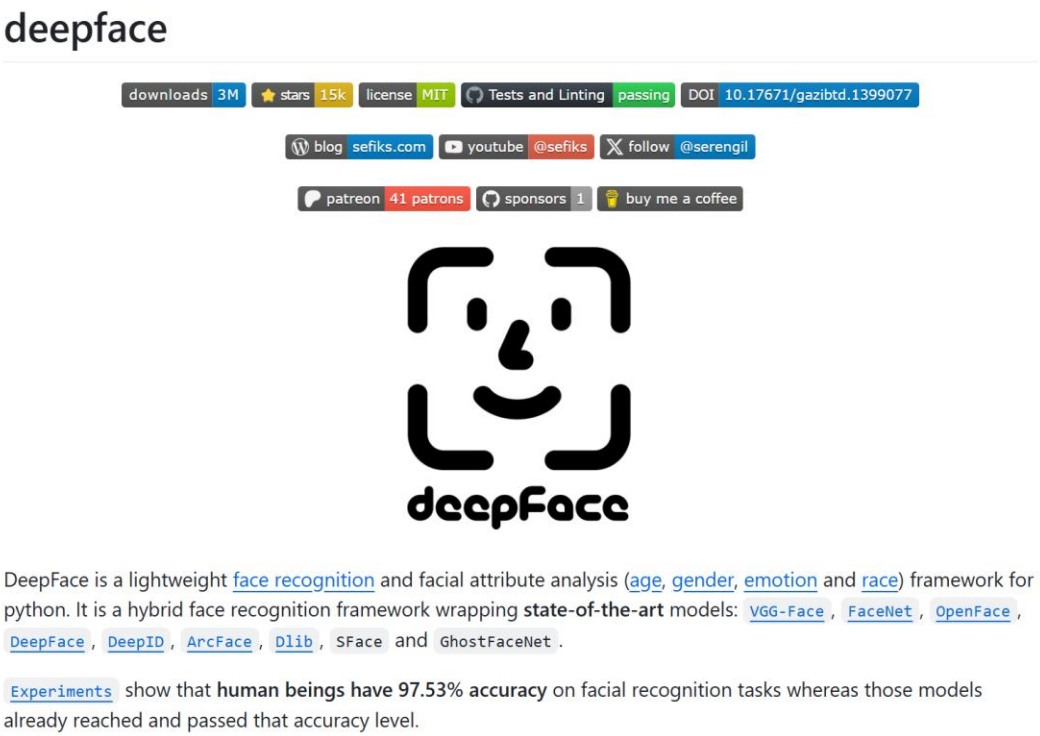


Figure 2: The README file of the project

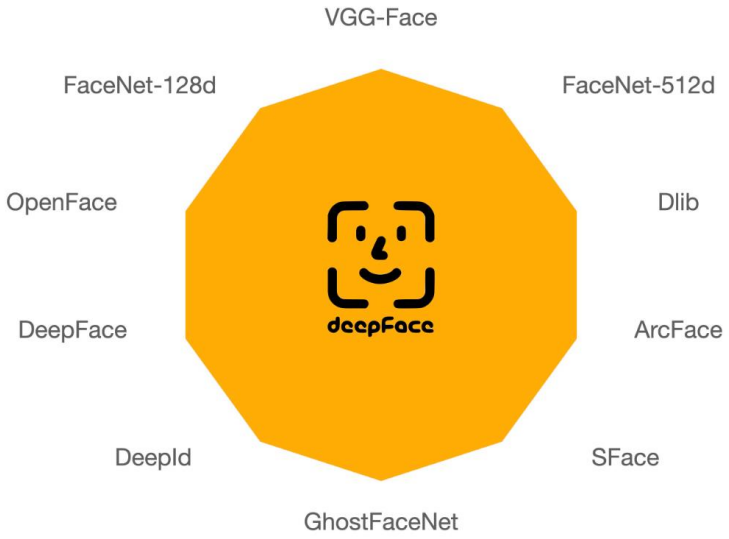


Figure 3: The related models that be used in deepFace

FaceNet, VGG-Face, ArcFace and Dlib are overperforming ones based on experiments - see **BENCHMARKS** for more details. You can find the **measured scores** of various models in DeepFace and the **reported scores** from their original studies in the following table.

Model	Measured Score	Declared Score
Facenet512	98.4%	99.6%
Human-beings	97.5%	97.5%
Facenet	97.4%	99.2%
Dlib	96.8%	99.3 %
VGG-Face	96.7%	98.9%
ArcFace	96.7%	99.5%
GhostFaceNet	93.3%	99.7%
SFace	93.0%	99.5%
OpenFace	78.7%	92.9%
DeepFace	69.0%	97.3%
DeepID	66.5%	97.4%

Figure 4: The measured and Declared score of different models

This can be used to detect and recognize a variety of information, such as age, gender, emotion, race, etc. Our project mainly applies its function of **identifying emotions**.

Figure 5: A using example of deepFace

2. Develop environment

This is an integrated package, which contains the we need such as TensorFlow, matplotlib, OpenCV python, NumPy, pandas and so on.

Installation pypi v0.0.93

The easiest way to install deepface is to download it from PyPI. It's going to install the library itself and its prerequisites as well.

```
$ pip install deepface
```

Perform this command in cmd

Alternatively, you can also install deepface from its source code. Source code may have new features not published in pip release yet.

```
$ git clone https://github.com/serengil/deepface.git
$ cd deepface
$ pip install -e .
```

Once you installed the library, then you will be able to import it and use its functionalities.

```
from deepface import DeepFace
```

Figure 6: The instruction of environment building method

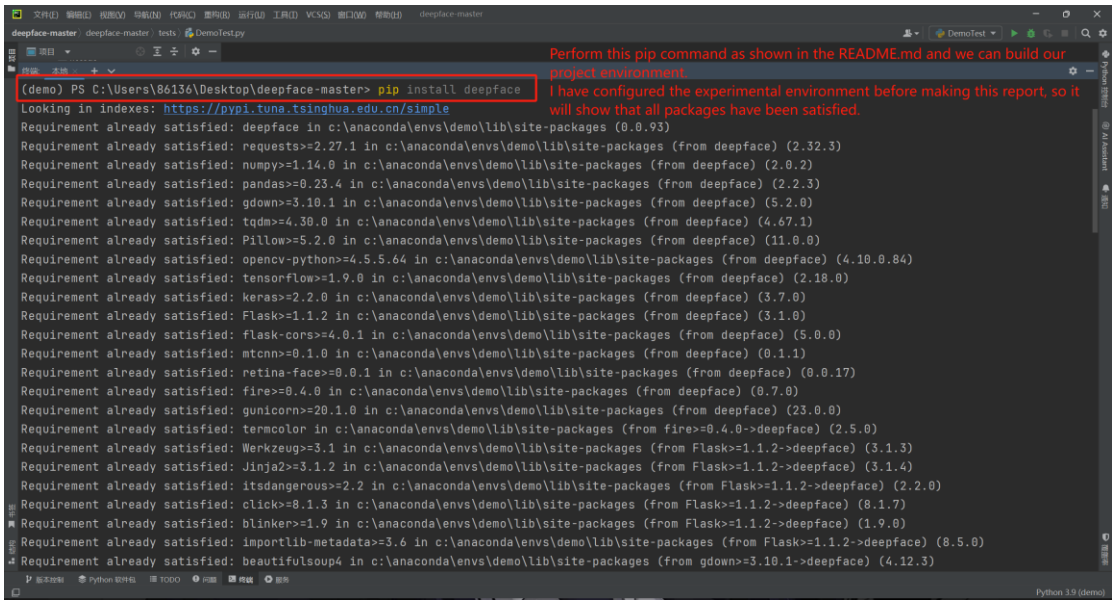
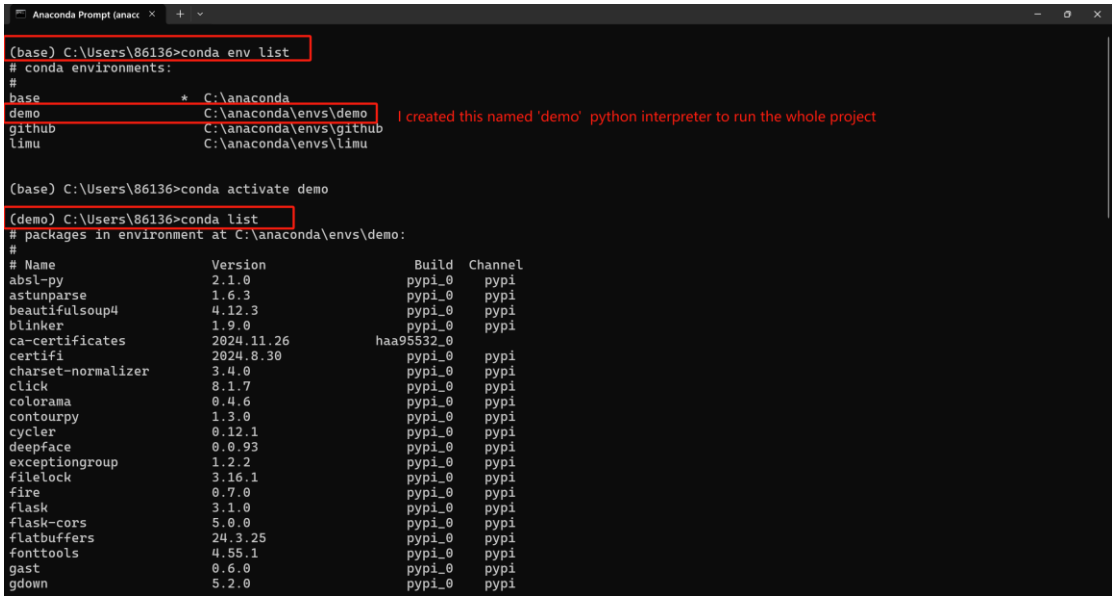


Figure 7: Configure our project environment

There were some ***minor incidents*** during the configuration experiment. Since this project was created several years ago, the configuration environment at that time was ***a little older than the current one***, so I needed to configure some packages with older versions to meet this project. So I used ***anaconda*** to create a python 3.9 virtual environment to run the project



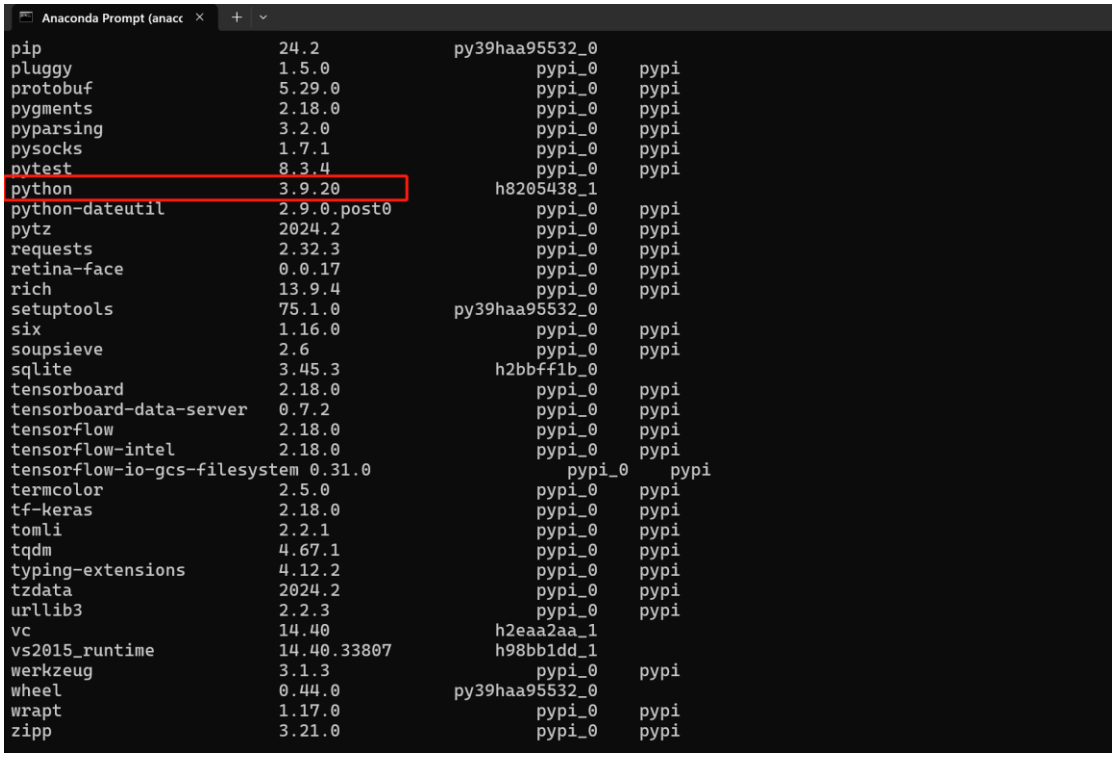
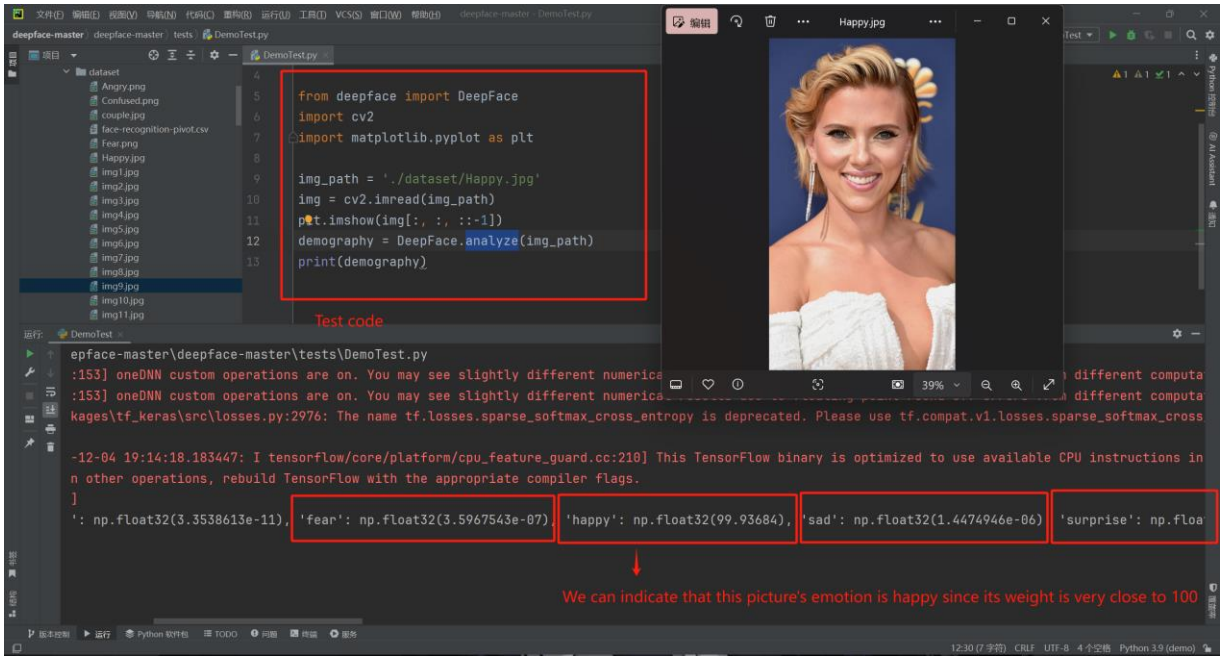


Figure 8: Using anaconda to create virtual env

3. Running result analysis

In the analyze module of deepFace, emotions can be recognized as follows: ***angry, disgust, fear, happy, sad, surprise and neutral***. This model will weigh and score the recognized emotions in the image, and finally get a ***dominant_emotion*** as the final judgment result



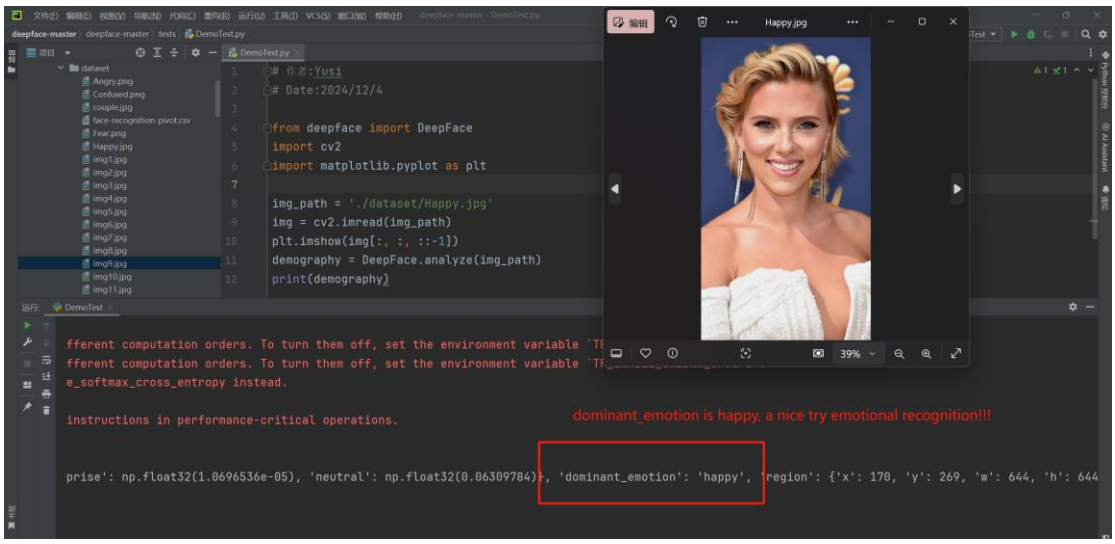


Figure 9: Test example (Happy emotion recognition)

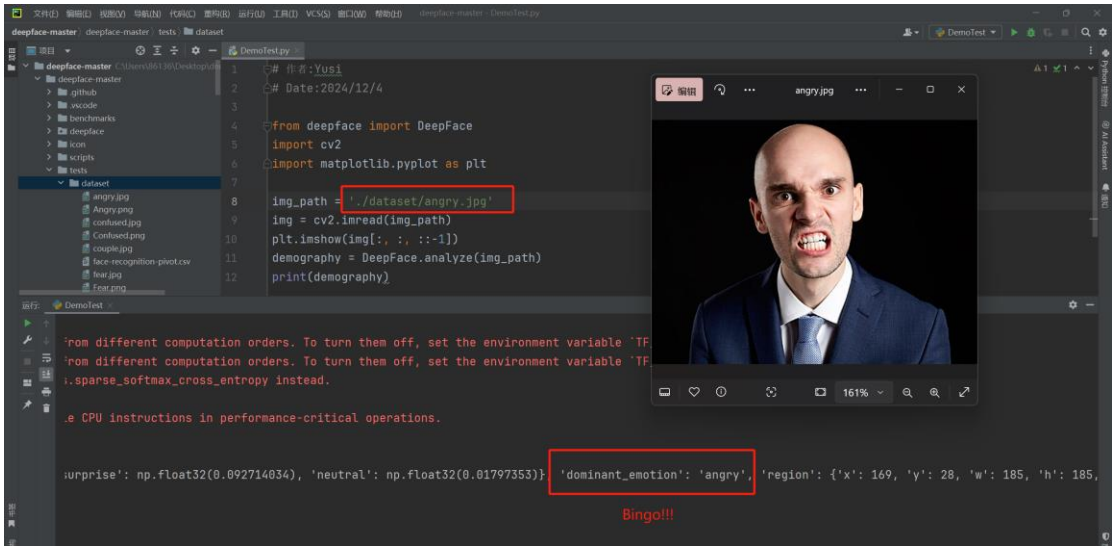


Figure 10: Test example (Angry emotion recognition)

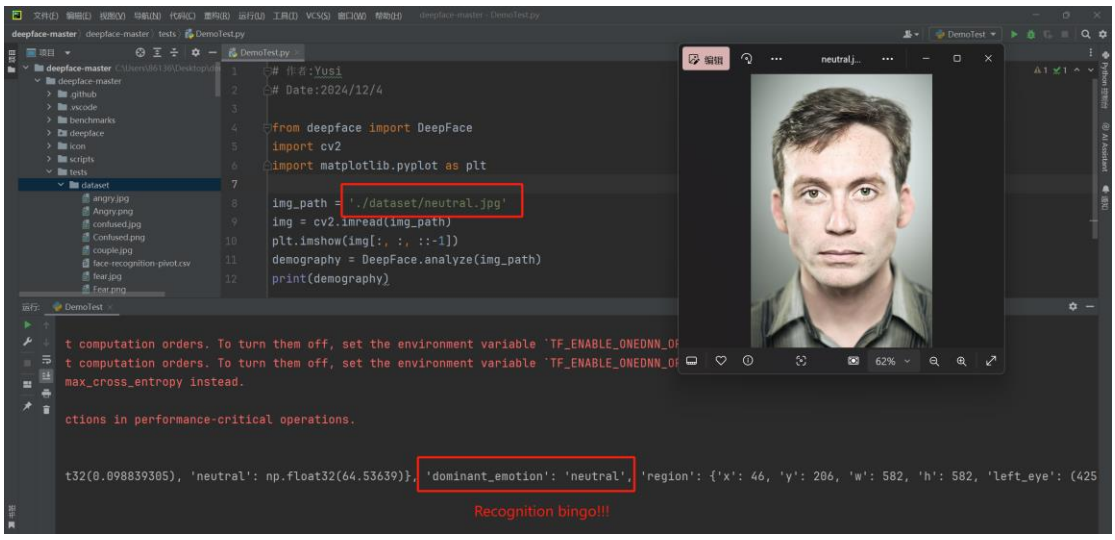


Figure 11: Test example (Neutral emotion recognition)

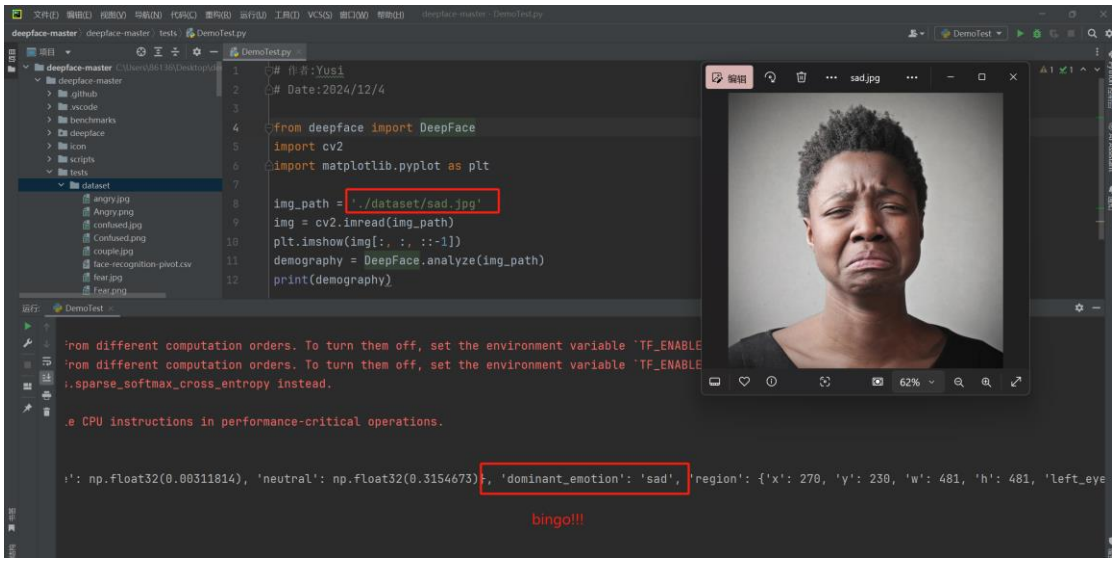


Figure 12: Test example (Sad emotion recognition)

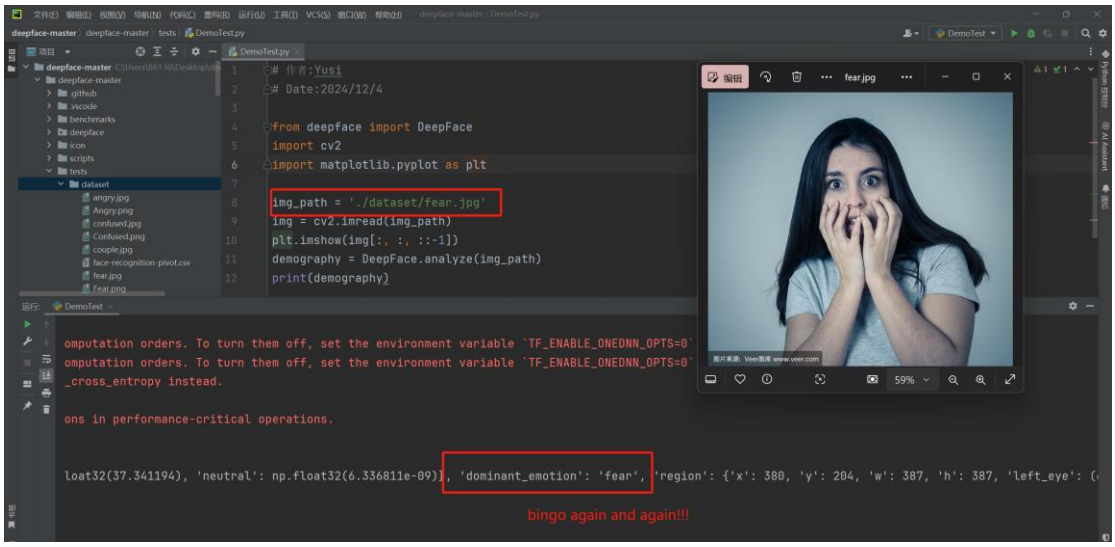


Figure 13: Test example (Fear emotion recognition)

As you can see, all our test cases produced correct mood predictions!!