Project Outline of Emagine

jiahe.zhang 20205722

October 2022

1 Introduction

I have a strong interest in image processing and computer vision, and I have used a lot of mobile image processing software, and I think they have two main problems. First, they often give users specific processing templates to achieve a "preview-generate" image processing process, which greatly reduces the user's participation and innovation. Second, for high-quality original images and high-quality images after secondary processing, users are more independent to appreciate, and they cannot do so while having the original identity of the image, let the image circulate in a wide range of user groups to further enhance its actual value, or sell and resell the ownership of the image to earn practical benefits (these inspirations are largely derived from the concept of NFT).

Based on OpenCV for Android 4.4.0, Emagine will give users the absolute freedom to create their own filters and render images based on giving them enough image rendering templates. The image producer will permanently associate with the image and profit from it. Those high-quality original images, and the processed high-quality images, are not only images, but also given commodity value. Users can re-trade the key to an image to gain ownership of that image and put the image they own into their own "art gallery" to show to others.

All in all, this is an Android NFT trading community with rich image processing tools.

2 Technical requirements

Tools:

Development Tool: Android Studio
Back-end language: Java(main) and python
Main dependencies: OpenCV 4.4.0, Chaquopy, Wechat

3 Functions

3.1 User interface

Users can register and log in to the account, the program allows users to enter a username and password, and store the relevant information in the database. Each time the user logs in, it will be verified.

Only logged-in users can use the features of this software (including image processing and NFT-like communities).

3.2 Image Process interface

The program combines the native code of Java with some of the features of OpenCV's external libraries to provide users with a variety of image rendering templates. And to provide users with a high degree of freedom filter customization services.

Users can choose to upload images locally and use the tools to process them. Finally, as the author, the processed image artwork will be uploaded to the program.

3.3 NFT interface

Emagine provides a complete image gallery for all official users, allowing users to browse all uploaded high-quality images, as well as their creators and owners.

Each user will also have their own showcase to display their collections.

All image commodities are supported for user-to-user transactions.

4 Plan

4.1 Alpha

plan:

week 7:

1. Complete the configuration of OpenCV 4.4.0 on Android Studio and test the demo of related features.

2. Complete the support configuration for python code writing on Android Studio and conduct demo testing of related functions.

3. Complete the configuration of support for the LitePal database on Android Studio and establish the core data model.

4. Complete the support configuration for WeChat Pay on Android Studio and test the operation of the feature demo.

week 8:

1. Implement sign-in and sign-up functionality.

2. Implement image processing methods based on Java native code implementations.

3. After the user uploads the local image, the image can be uploaded to the Emagine software database after image rendering.

week 9:

1. Implement OpenCV-based image rendering templates.

2. Implement a custom OpenCV-based filter rendering filter.

3. Realize the function of art exhibition hall, display all the artworks in the database to users on the home page, and allow users to upload their own artworks to their own art museum.

4.2 Beta

plan:

week 10:

1. Enables the binding of the artwork image to the creator and the generation of the owner key of the artwork.

2. Enables users to browse artwork information and add it to likes to follow its flow record.

week 11:

1. Realize online transactions with the help of WeChat Pay.

2. Enables transaction record-based transfer of ownership of artworks.

week 12:

1. Realize the review of the artwork.

2. Implement comments for users.

3. Realize the review of a private art museum.

4.3 Final presentation

plan:

week 12: Based on the current project progress, rush work or enrich the aesthetics of the user interface to make it more user-friendly.

week 13: Repeatedly test the program to find and resolve bugs.