CSC4140 Project

Computer Graphics

May 24, 2022

HDR-Plus Pipiline for Video

Student ID: 119010434

Student Name: Zhang Qihang

This assignment represents my own work in accordance with University regulations.

Signature:

1 Overview

In this project, the program was extended to make more effects. Added support for microfacets, implemented environment light as a light source, and added the ability to change focal length/lens radius to simulate thin lenses.

2 Review

2.1 Code for jhfmat/ISP-pipeline-hdrplus

The code for this Repository is not general to different platform. Only work on windows now, so I can't access to improve it. Also, this code tends to use a kind of file(.raw) which is defined by developer himself (at least I can't find same convert algorithm in the internet). So I choose to find other Repository which implement the HDR-PLUS algorithm.

2.2 Code for timothybrooks/hdr-plus

The code for this Repository is great but still have some problem for me.

First, the environment build is complex. You will need many seldom seen c++ library. And some of these lib will need you to build yourself manually. For example, you will need around 9 GB space to build the llvm to support this project.

Second, the code is implemented many years ago, and the maintain for this code is not in time. For example, the code will only support by the Halide with version 10.0.1 or before, but in readme file, you are suggested to use "latest" Version of Halide(which will leads to an error).

Last, after the build, you are surprised to find that the "make" of the code will leads to an error, which is discovered by users(seen in the issue 74) but not fixed yet.

3 Choice of code: amonod/hdrplus-python

3.1 Advantage for the Code

First, this code is widely supported. As recommended, you can use "conda" to build your environment for this project. the using of virtual environment means that you can use this code in almost any platform you like. Also, using virtual environment can auto-build dependence for the project.

Second, this algorithm use the more common file type, "Adobe .dng" file. You can find many converter in the internet to convert RAW file for different camera company. This will be helpful when you want to get some test data.

3.2 Disadvantage for the Code

The code for python ineffective. It means that it will be time-consumed for our program. In my test case, a dng picture size with 1980 * 1080 will consume 6-6.5 seconds to process. Then you will need half hours to get a video(30FPS) within a few seconds.

4 Error may occur when Build(Important)

4.1 cv2 in Virtual Environment

After you build your own virtual environment, you may meet an error for function:

```
cv2.imwrite()
```

When you run the code, this function will leads to a:

```
segmentation fault
```

If you meet this error, you will need to use below command in the command line:

```
conda activate ${Your conda env name}
conda uninstall opencv-python
pip3 install opencv-python
```

4.2 Error caused by .dng file

If your .dng file is convert from some other kind of raw file. Some of data of the file will not correct and this may leads to an error during processing:

```
complex division by zero
```

Then you will needs to change some code in the dependent library in numbers.py:

```
        if not breal and not bimag:            # line 1068
            raise ZeroDivisionError("complex division by zero")
```

Change to:

```
        if not breal and not bimag:            # line 1068
            breal = 0.00000001
            bimag = 0.00000001
            # raise ZeroDivisionError("complex division by zero")
```

Address in mac:

```
/Users/&{Your User Name}/opt/anaconda3/envs/${Your conda env Name}/lib/
    python3.7/site-packages/numba/cpython/numbers.py
```

4.3 Error with Path

You need to make sure that current path when you execute the file is under "/hdrplus-python"

5 Implementation

The implementation for the project is not hard. The main task is to make an work flow which can process the picture in sequence and then gather all picture to form a video.

5.1 Work Flow

This part, we must make the algorithm. Then make sure that the algorithm can also work well when call by Work Flow code. Then we need read all file which will be used to form video.

Also, the output for the work flow will needed to be organized well. Then we can use "ffmpeg" to connect all frames then get a video.

5.2 User Interface

In this part, we can use pyqt5 to easily form a user interface, which can make our program easier to use. Also, you can give some basic error info in the interface.

However, when the program is broken, user can also see the detailed log about the program output.

6 Code Output

6.1 User Interface

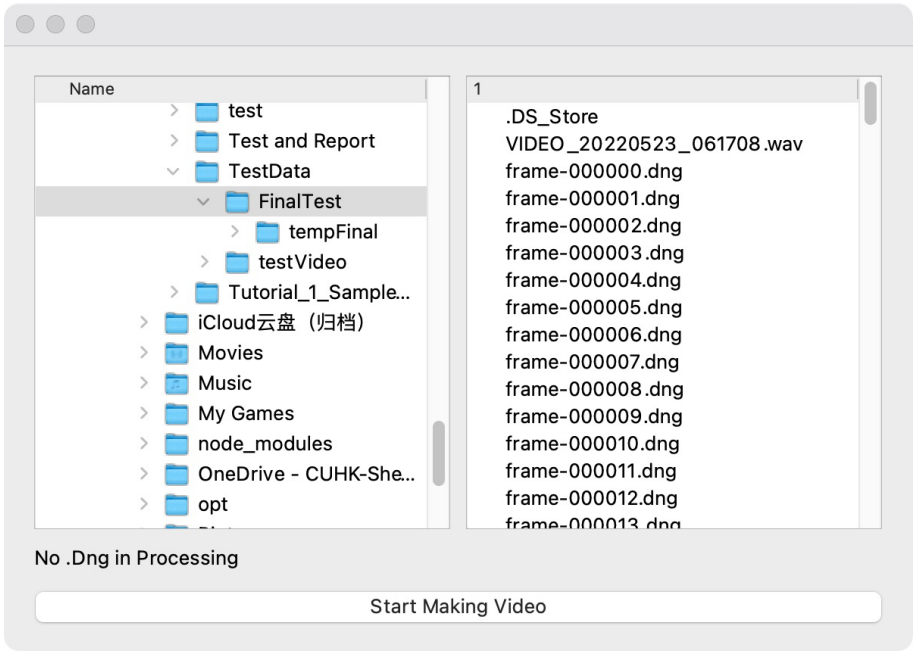

Figure 1: User Interface when Select

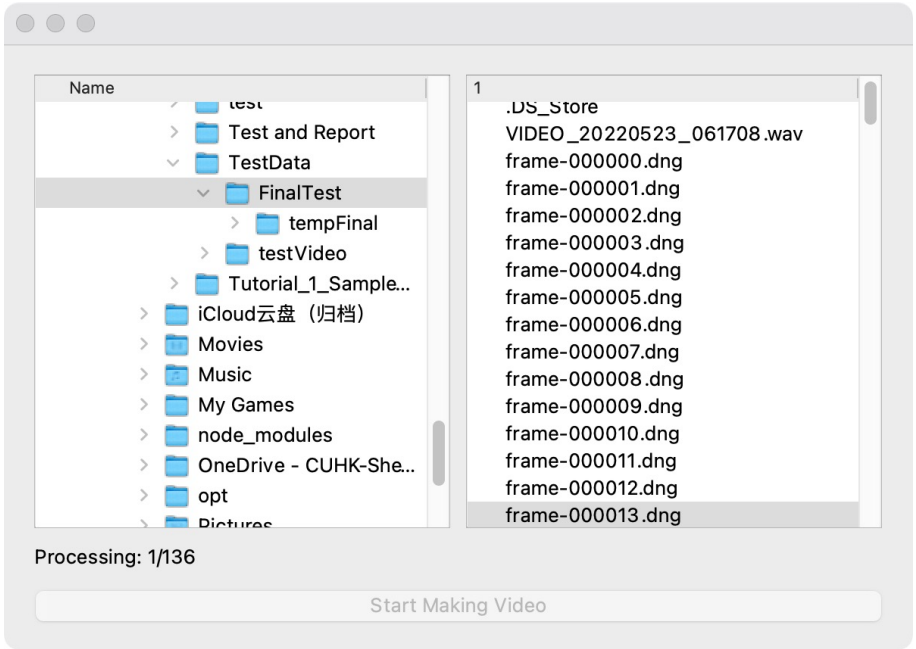

Figure 2: User Interface when Processing

6.2 Output for Single File

6.2.1 Data from HDR-Plus Project

Too Large so I use the screen shot.

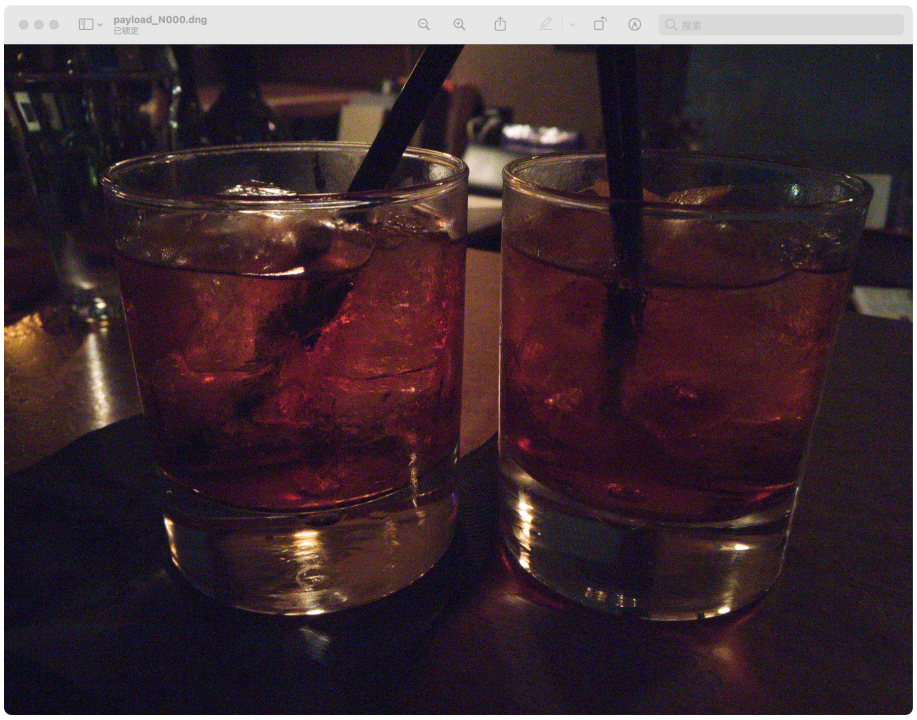

Figure 3: Direct Open .Dng

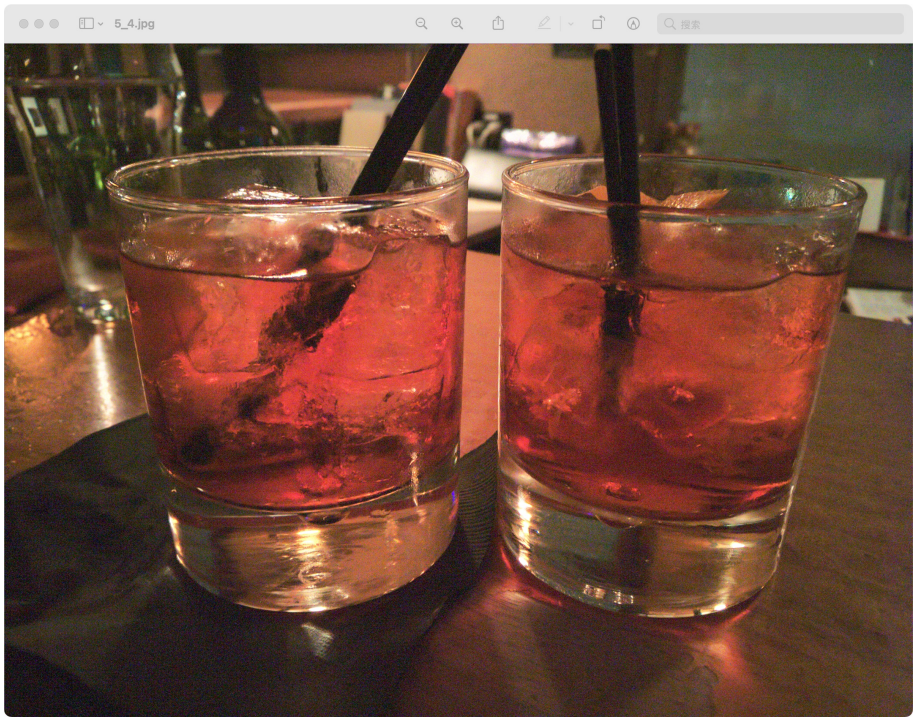

Figure 4: After Process .Jpg

6.2.2 Data from Myself

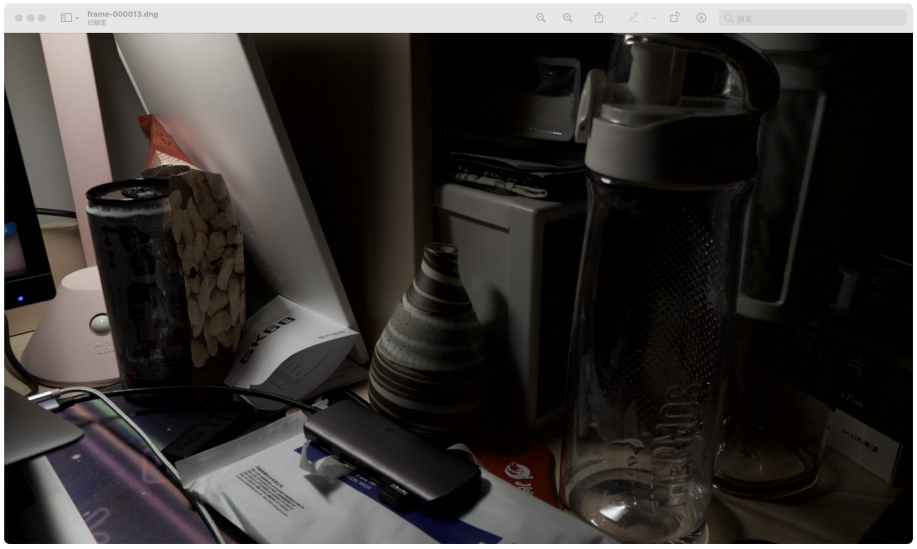

Figure 5: Direct Open .Dng

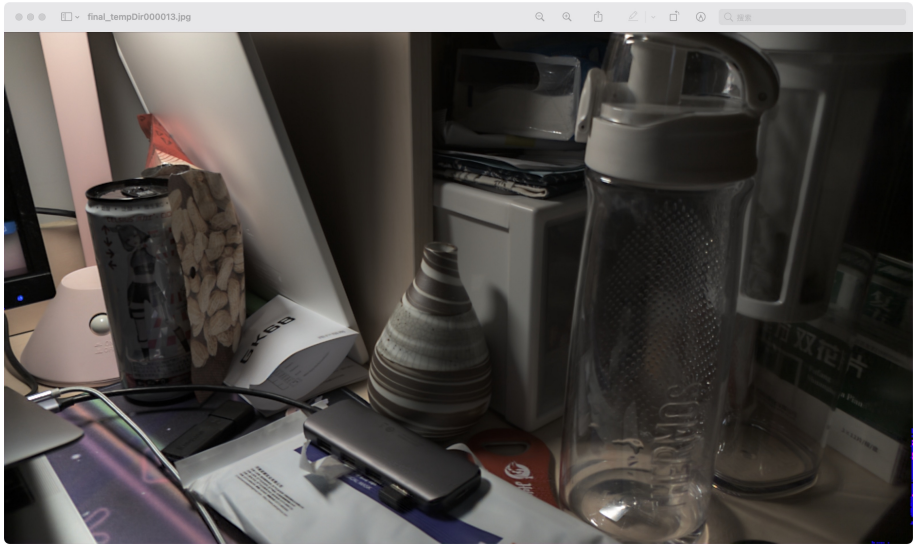

Figure 6: After Process .Jpg

6.3 Video for Myself

You can find Video In the Latex Dir, which named as Final.mp4. You can also find in tempDir/Final.mp4 in the data Dir.

7 Further Imporvement

7.1 Error when Color Change Greatly

When the color change greatly, the merge algorithm will leads to an screen blurred picture. Because the color change greatly in different file, then the picture afterwards will not give a good contribution to current picture. Then will lead to an blurred screen in small area.

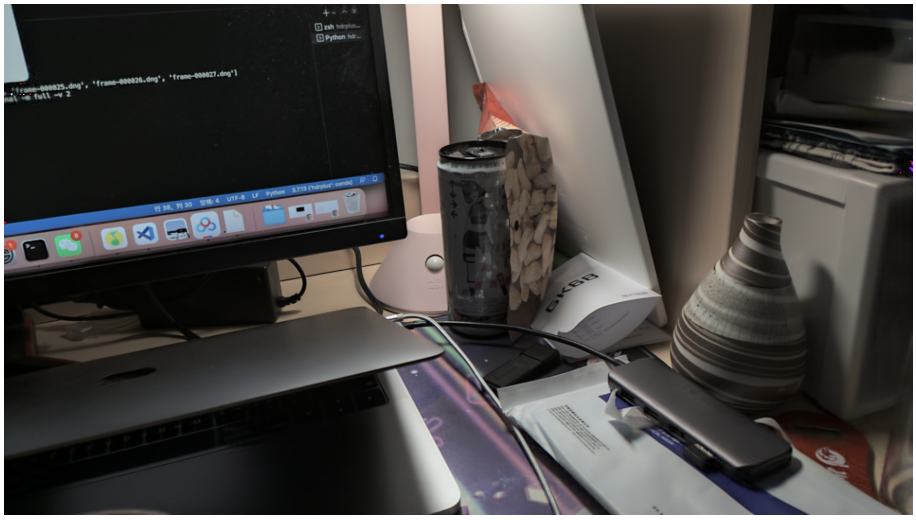

Figure 7: Before Screen Blurred

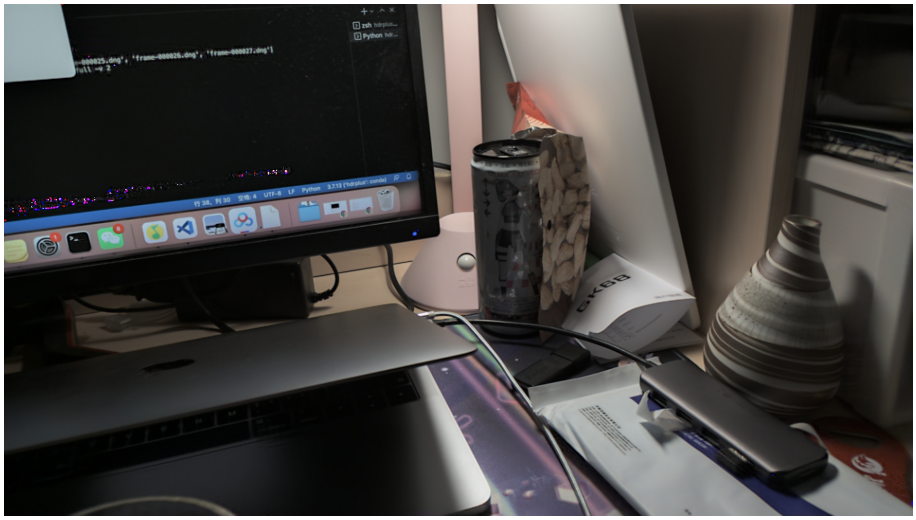

Figure 8: Screen Blurred

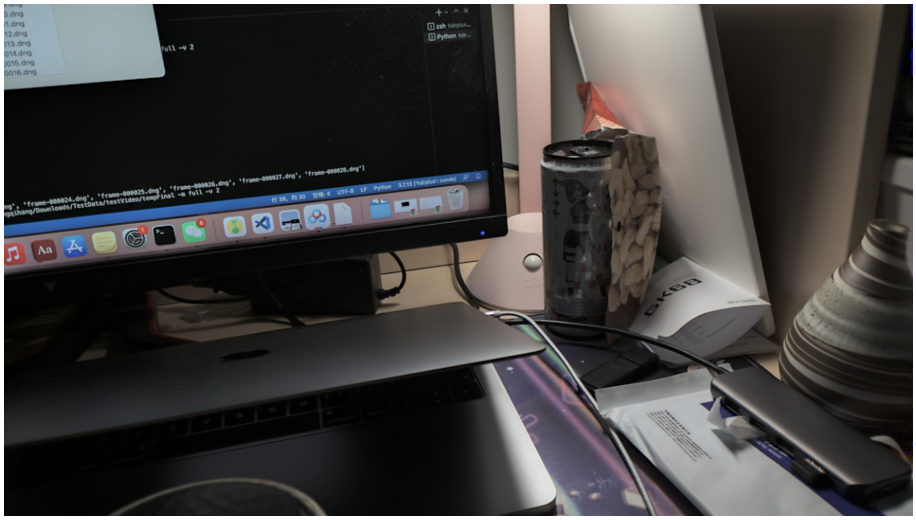

Figure 9: After Screen Blurred

You can see that when the color become white from black quickly, it's will be blurred.

7.2 Improve the Picture to Get it Beautiful

For .dng file, some software can change it's performance using "Presets", like Photoshop, Fujifilm Holdings... These software can use .xmp file to auto-change the picture in raw type. So, I can use Python XMP Toolkit to change the dng file. Then we can make the picture not only brighter and also more beautiful.

But because the time is limited, so I may complete it later.

8 Run

You can read the readme.md in the code file before run it.

9 Data Resources

I Use the Software Motion Cam to take the dng video.

DNG from Phone to Computer with HDR+: Brighter Picture, Brighter Future!