Face Tracking by Mean shift Algorithm

Tianye Zhao
Zhaoxuan Qin
Bishop's University
Sherbrooke, Canada

With the rapidly growing popularity of the Internet and mobile Internet, there are huge demands of face tracking in many certain situation such like video meeting and identity verification, basically, the face tracking technology is focus on analysis the useful information like biometric information of a person in pictures or videos. This project is focusing on the application of mean shift algorithm in face tracking area. This project also detect the different influence of hardware on the result of face tracking, in this project, we are focusing on using CPU and GPU to finish face tracking independently to see the difference between them.

In this project, we choose to use mean shift algorithm to find and track human face in videos. Basically, this algorithm is focusing on catch the features of face and given a certain small window to it. When it moved in the camera, it will move the window to the right place which could match the features, it's a dynamic progress and it will get a best place that fit the features of face. Which is the mean method in the CPU program. In the GPU program, besides the mean shift algorithm, it also used a method named GPU Arrays, it could accelerate the speed to GPU when it catch the features

One thing need to be noticed it that there are two elements that may influence the result of the program:

a. The size of the image: over small image will affect the recognition effect, overlarge image will effect recognition speed. In order to avoid such problem, the tester should keep the same distance to the camera.
b. Image resolution: The lower the image resolution, the more difficult it is to identify, since we had use same camera, the effect of this element can be ignored.

Here are the results of using CPU and GPU:

1. CPU

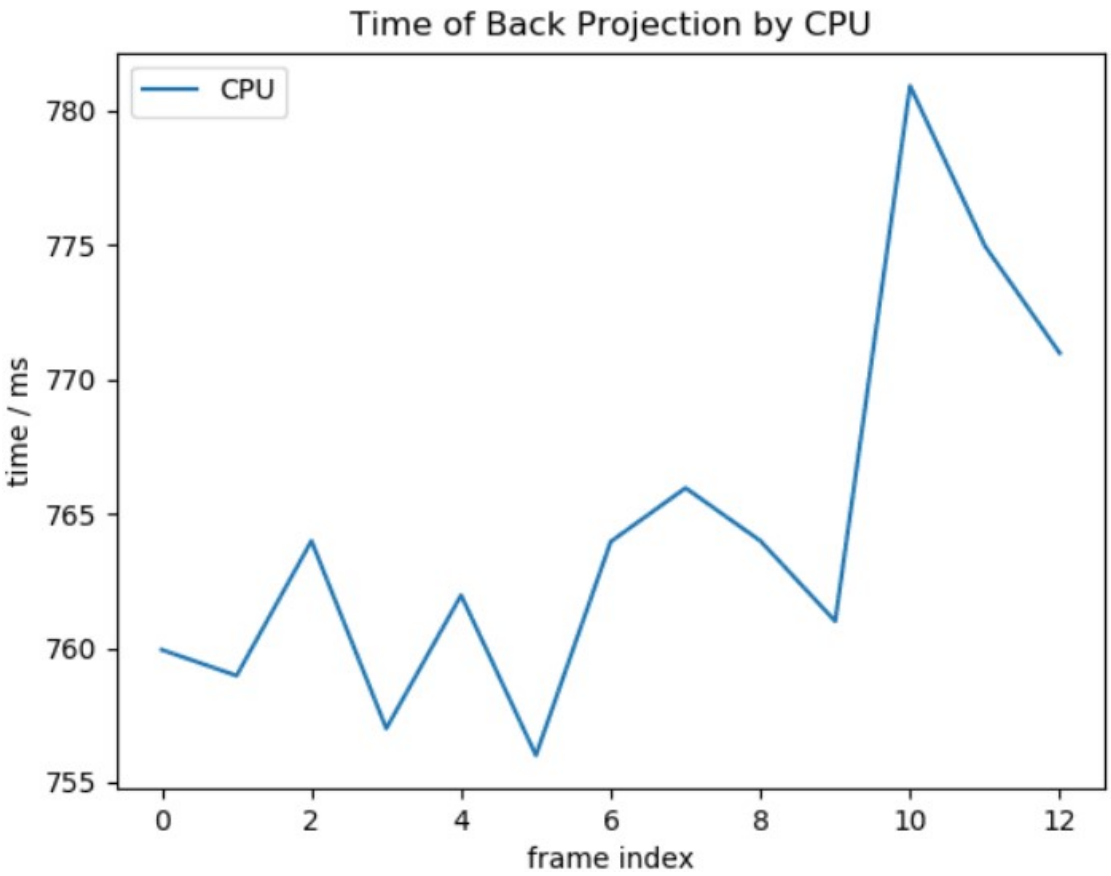
1.1. The effects of CPU mean shifting

In the test part, we record 15 seconds after the program start tracing the testers face. It can also generate the frame in a single test and the run-time image.

In order to illustrate the effects directly, we had recoding the testing video and preserve it as gif, from this gif, it is clear to see the speed of mean shift algorithm using in CPU is slow, it takes a lot of times to catch the face of tester.

https://drive.google.com/file/d/13NmlB9p51czy72iKu2ggzDUmaqRnlIAm/view?usp=sharing

1.2. frame-CPU and runtime



This image shows the relationship between time and the frame index. This figure shows the results of CPU mean shifting, compare with the result of GPU, it only generate few results in 15 seconds

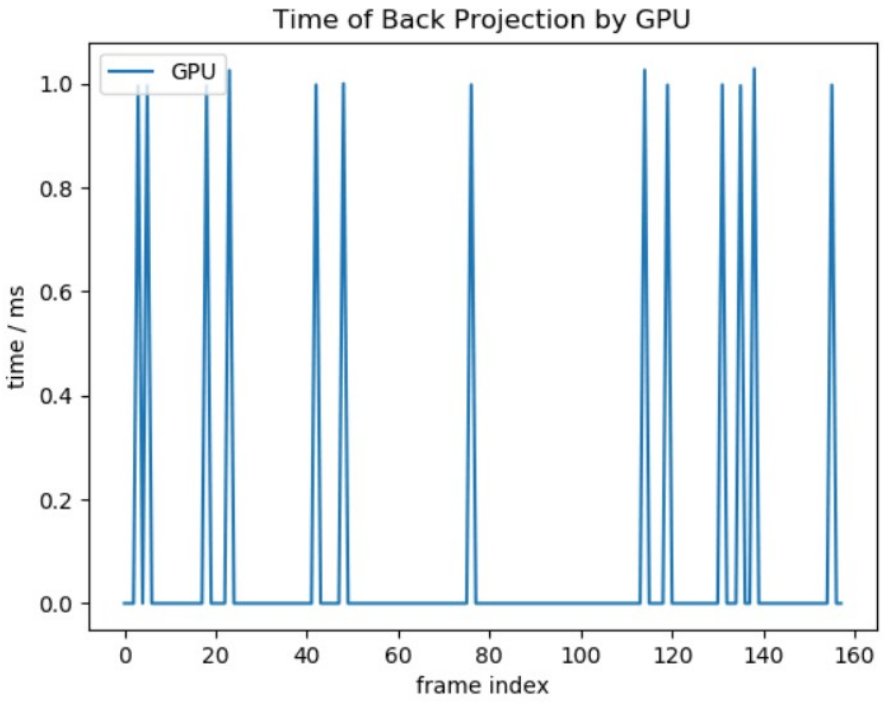
2. GPU

2.1. The effects of CPU mean shifting

In this part, we also record 15 seconds of test video and made it into gif, and from this gif, we can easily found that the speed of face tracking is faster than CPU method because the frame is smooth.

https://drive.google.com/file/d/1i6L8PB3JzvICeJUm6hnmtzqDKfvhrMGl/view?usp=sharing

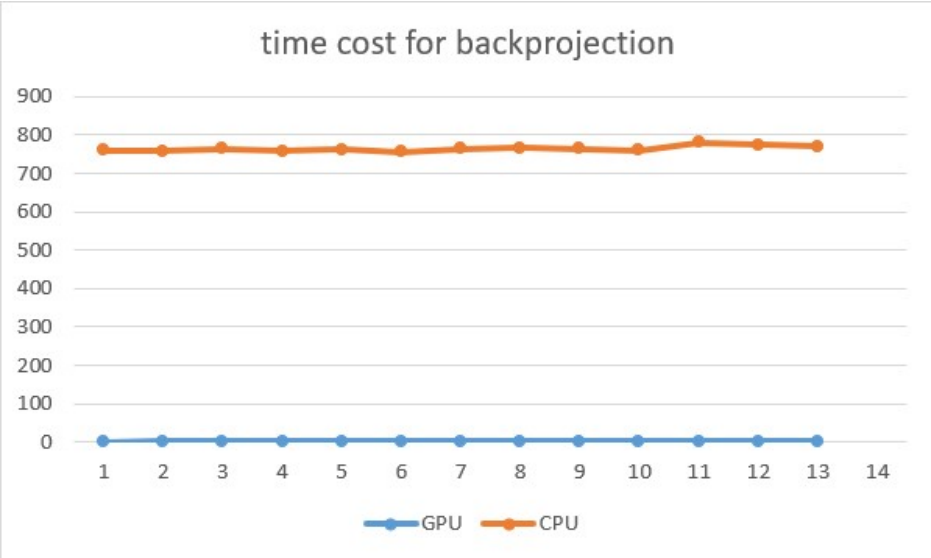
2.2. frame-GPU and runtime



This picture just shows a part of the results of GPU, we can conclude that the quantity of its results is larger than CPU, which means GPU has a better performance in face tracking. However, a brunch of results has small value, one possible reason may be the calculate speed of GPU is too fast so it can not fit every single frame.

3. Comparison between CPU and GPU

We use the results of CPU and GPU to draw a line chart to compare the performance between them.



This picture clearly shows that in the same period, the CPU only generates few results and the GPU generates 157 results (both of them are use mean shift method), it proves that in object tracking area the GPU has a better performance, compare to the CPU.

4. Conclusion

This project explored the mean shift algorithm in face tracking area, and also explored the different performance between CPU and GPU when use same mean shift method. The conclusion is that the GPU has a better performance than CPU. Also, The project have some place that could be improved, for example, the result of GPU has a brunch of small value, even zero value it could be improved by decrease the range of block size and greed number.