

FastChi – The Parallelized Fast Style Transfer toward Chi-chi video

Member: Yen Liu, Peter Chuang, Sunner Li

I. Introduction and Motivation

The style transfer is a very popular problem in recent year. Moreover, there're some applications which use such kind of transferring into their product. Prisma is an interesting example. The function of the application can do the style transferring toward any input image given by user with different style. By the effect of the application, it can let us to experience the daily life with different mood.

However, the speed and performance are important issues that should be conquered. In Prisma, the smart phone should spend amount of time to transfer the image. Moreover, the same shortage can be seen in the situation to transfer the style toward the whole film. Rather than dealing with single image, the program should spend more time to complete the task for the video.

In order to accelerate the style transferring procedure, we purpose the parallel structure toward this task, and speed up the procedure of transforming. The network will split the video as two parts, and use two GPU to transform the style to the content image in parallelism. For our expectation, the video which contains chi-chi (shiba-lnu dog) will be transferred in very quick speed. Moreover, the changing can make user be more pleasure and enjoy the daily life.

II. Related work

About the question of the artificial style transfer, Gatys et al. [1] adopt the deep neural network to achieve the great performance, and it's the first implementation to use convolution network to complete the task. However, the speed is the bottleneck. Ulyanov el al. [2] raised another creative idea which called instance normalization to

accelerate the whole transforming process. Moreover, the instance normalization can also reduce the correlation between batch training images.

It's another critical problem to design the appropriate loss function. In the previous work [1], the authors usually used pixel-to-pixel error to compute the loss. However, this method of loss computing didn't consider the spatial variance. For example, if the result which is generated by deep network shifts a little toward the structure of origin image, the value of loss computation becomes very large which isn't reasonable. To solve the correlative problem, Johnson et al. [3] purposed the combination of perceptual loss with usual forward network. The network would use feature map in higher-level to compute the loss value rather than using output of the network directly, and the designment can make the network preserve the capability which dealing with the situation of spatial difference.

In this work, how to transfer the style to the content image with more high speed is the problem we should consider. We purpose the implementation that uses parallel mechanism to accelerate this procedure.

III. Structure

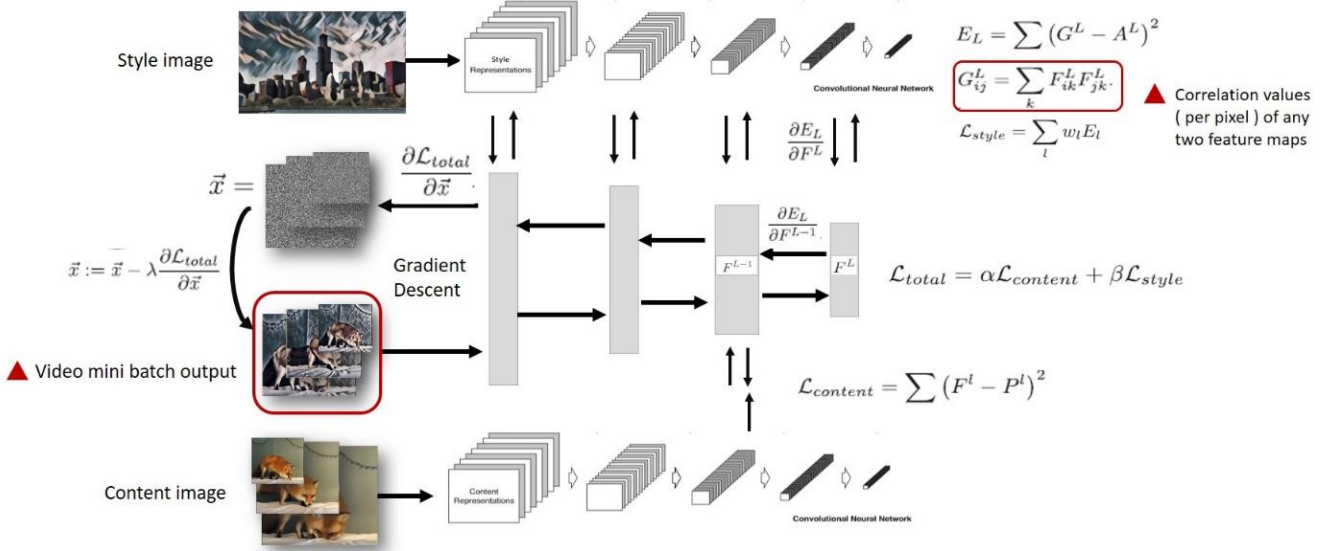


Fig. 1. The structure of our work

The Fig. 1 shows the structure diagram of the main part.

IV. Language selection

In our implementation, python programming language is selected to be used. On the other hand, we decide to adopt build the model in Python with Tensorflow as the deep learning framework. , so based on the decisionAs the result, we need to search about some parallel methods in Python.

The Joblib Python library maybe a suitable method to accelerate the main parts we which is mentioned on the above structure diagramin Fig. 1, video mini batch loading and correlation matrix calculation. It saves much error-prone programming in typical procedures just like caching and parallelization.

V. Statement of expected results

After paralleling the model, we expected not only solved the slow problem on single image processing, but also make a video which is buildt with many frames can

be displayed fluently. The result video can be generated faster than the original implementation. SecondAnother expectation, with the acceleration, is tothe application can attract more users to make their photos interesting without waiting a long time.

VI. Timetable Plan

In this section, the plan which should be finished are listed in order.

1. Before November 15, 2017, we should read the relative papers and information.
After this work, we can fully realize how the transferring mechanism it is.
2. Before November 30, 2017, the environment should be set. And the idea of original paper can be conducted.
3. Before December 15, 2017, we will read the code of original implementation and find the parts which can be deal with in parallelism.
4. Before the end of 2017, we should finish the parallel of the whole program.
5. Before January 10, 2018, we will spend time on merging the each part of the code and conquer the difficulty that we suffer from.
6. At last, we should prepare the presentation for the final course.

VII. Reference

- [1] Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge, "Image Style Transfer Using Convolutional Neural Networks," In *2016 Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, Nevada, USA, 27-30 June, 2016, pp. 2414-2423.
- [2] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky, "Instance Normalization: The Missing Ingredient for Fast Stylization," arXiv: 1607.08022v2 [cs.CV], Sep. 2016.
- [3] Justin Juhnson, Alexandre Alahi, and Li Dei-Dei, "Perceptual Losses for Real-Time Style Transfer and Super-Resolution," arXiv: 1603.08155v1 [cs.CV], March 2016.