Project Proposal

1. Title: Parallelized Fast Style Transfer toward chi-chi video
2. The participants: 劉晏 李政哲 莊侑穎
3. Introduction/motivation:

曾經用過一款名稱為Prisma 的app，它的功能是將使用者輸入的圖片進行不同風格的轉換，這樣的應用可以幫助我們將日常的照片，轉變為各個不同知名畫派的風格，讓我們可以用不同的角度，去體會日常的點點滴滴。而在github上，我們發現了一篇十分有趣的應用，叫做 ”Fast Style Transfer in Tensorflow”，他將圖片套用於CNN之中，讓圖片轉換為另一種風格，我們希望能夠將上課中所學，應用在這之中，因此，這次的Project我們將基於”Fast Style Transfer in Tensorflow”的code，去進行平行化的修改，讓他運行的更有效率！

1. Statement of the problem

在”Fast Style Transfer in Tensorflow”的程式之中，我們有發現一些可以平行化的部分，將會針對這些地方去進行程式碼的修改與撰寫，期許能夠達成運算加速之目的。

1. Proposed approaches

Algorithm流程圖 加上 加速部分

1. Language selection

Python flexible

1. Related work
2. Statement of expected results

顯著加速

1. A timetable

2017.11.15前：研讀相關論文與資料，充分了解其技術內容

2017.11.30前：環境架設完成，並使原始project可以順利執行

2017.12:15前：詳讀程式碼，尋找可以執行平行化的部分，並進行研究

2017.12.31前：完成程式平行化

2018.01.10前：統整於實作上遇到的困難與可行的解決辦法，並進行檢討

2018.01.15前：準備presentation的簡報與上台預演

1. References

<https://arxiv.org/pdf/1508.06576.pdf>

<https://arxiv.org/pdf/1603.08155.pdf>

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

Member: Yen Liu, Peter Chuang, Sunner Li

1. **Introduction and Motivation**

The style transfer is a very popular problem in recent year. Moreover, there’re some application which use such kind of transferring into their product. Prisma is an 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.

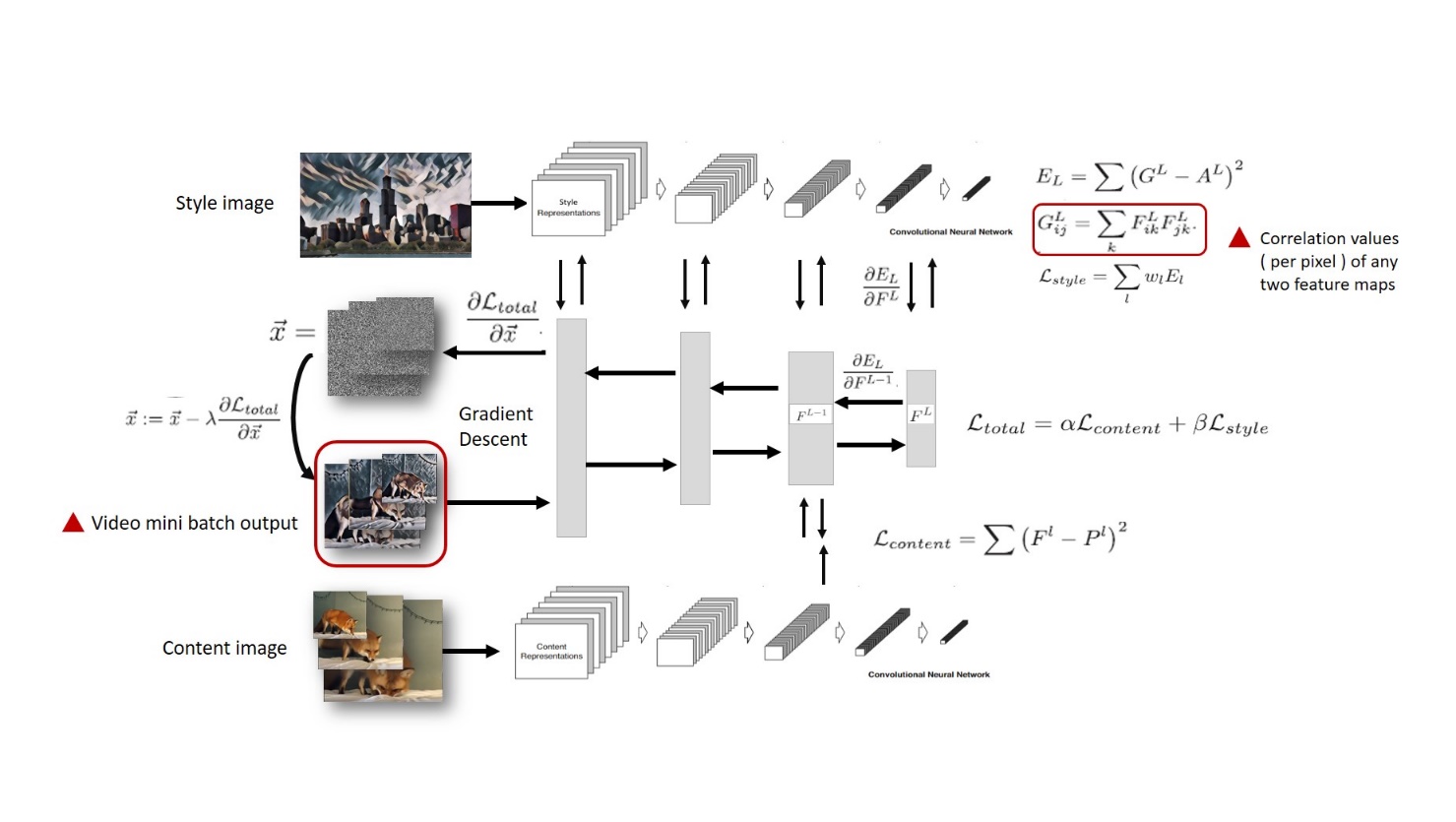
However, the speed and performance are important issues that should be conquered. In Prisma, the smart phone should spend amount of time to transfer the image. In order to accelerate the style transferring procedure, we purpose the parallel structure toward this task, and speed up the procedure of transforming. 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.

1. **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.

1. **Structure**
2. **Language selection**

We decide to build the model in Python with Tensorflow, so based on the decision, we need to search about some parallel methods in Python.

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

1. **Statement of expected results**

After paralleling the model, we expected not only solved the slow problem on single image processing, but also a video build with many frames can be displayed fluently. Another expectation, with the acceleration, is to attract more users to make their photos interesting without waiting a long time.

1. **Timetable Plan**
2. **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.