Real-time Coherent Video Style Transfer
Project Plan

Chang Gao, Derun Gu

I. Introduction
With the increasing popularity of image style transfer softwares such as Prisma and Pikazo, image style transfer technology has become more well-known among people. At the same time, researchers are also chasing this hot topic, driving more and more promising results. While image style transfer has become a well-studied field and incubated many excellent works, the related field - video style transfer - shows slow progress and is commonly recognized as a much harder field. The main reason behind is that not only a real-time speed is required for streaming purposes, but also the coherence among consecutive frames needs to be maintained. This project aims to overcome these shortcomings based on pioneer works and achieve real-time video transfer with high temporal coherence. By the end of the project, an end-to-end computer program that could transfer a preload or streaming video into a given art style in real-time speed and high coherency is expected.

II. Background
In the past, transforming an image from one style to another required expert artist with considerable efforts, while transforming a full video was almost unimaginable. Recent research in machine learning and computer vision field casts light on this problem and has shown promising application prospects. This project aims to further explore based on previous results to achieve not only coherent but real-time video style transfer.

a. Single Image Style Transfer
Currently, deep neural network empowers computers to not only understand images but also generate works based on famous masterpieces. This process is summarized as stylization, which preserved the content of an image but transfers its artistic style to another target image. Single image style transfer gets rid of low-level texture transformation and was improved by the introduction of neural networks and Gram Matrix [1]. To further improve efficiency and accuracy, researchers exploited encoder-decoder framework with a representation extraction network [2]. By
encoder-decoder network, images can be compressed to reductive features and principal semantic information are encoded in hidden representation, which will then be projected into stylized images as preserved contents. To grasp a good understanding of the original image to reconstruct well, the neural network is forced to learn a good representation in the encoding stage. Inside the model, two objective functions are optimized simultaneously, namely the content loss and style loss. The content loss is defined as the Euclidean distance of abstract representations between generated and original images. The style loss is much difficult to define, while Gatys [1] proposed a good mathematical definition of style as the Gram matrix of hidden representations. Hence, alignment between target styles such as Starry Night by Van Gogh and stylized pictures is the process of minimizing style loss, which is the elementwise square distance of respective Gram matrices. In this way, a picture with contents from original images and style from another desired image is generated. This method showed excellent results in extracting pivotal information from the image and stylized images naturally. Base on the previous convincing results, it is also adopted as part of the network in this project.

b. Video Style Transfer

Frankly, a video could be considered a succession of images, and the simplest idea is to transform each frame into target style. However, videos contain rich temporal information and require coherence between two successive frames. This involves efforts to diminish obvious lighting and color changes, ghosting effects, artifacts and so on. To deal with video coherence, optical flows are intuitively applied as a temporal constraint between frames. [3] To speed up generation process, Chen et al. utilizes encoder-decoder-like models [2] with an additional neural network for estimation of optical flow [4]. Furthermore, Huang et al. implicitly apply another temporal loss, equivalent to the difference of inter-frame optical flow, to encoder-decoder model and performs in a more efficient way [5]. With the belief that a good model should work synergistically with tolerance on subcomponents’ imperfections, this project attempts to avoid model’s dependence on the explicit involvement of optical flow.

c. Multi-task Learning

Typically, in Machine Learning, the model trained is optimized for a single evaluation metric only. Nevertheless, due to lack of generalization, the model may not be able to perform well on other tasks with even very similar objectives. For example, both image classification and face detection function in a similar way, extracting the low-level representation of an image like edges and corners and then obtaining semantic abstraction of the image. However, an image classification model is good at categorizing pictures but could not locate faces in an image. To overcome this problem brought by single task training, researchers have exploited whether representation extraction could be shared between multiple tasks with only different objective functions. The alternative solution is called multi-task learning conversely. Nowadays, multi-task learning succeeds in various areas from natural language processing [6], speech recognition [7] and computer vision [8]. By bringing in additional proper tasks,
the network can work better on each specific task by utilizing knowledge learned from other tasks. There are two distinct methods for multi-task learning, called hard parameter sharing by sharing all hidden representation [9], and soft parameter sharing by regularizing constraints over separate layers. The keys to the success of multi-task learning include regularization on hidden parameters, eavesdropping for convergence of more difficult task, and dataset augmentation. Motivated by common procedures, such as encoder-decoder frameworks in stylized image generation and optical flow prediction, this project strives to combine the two tasks together and show that one single network could estimate optical flow and generate next stylized frames at the same time.

III. Objective

Video style transfer requires temporal coherence between consecutive frames, which is different from image style transfer. As mentioned before, existing methods make use of optical flow as a strong constraint for optimizing video stylization neural network model. Notwithstanding optical flow methods succeed to some extent, this project makes efforts not only stylizing video by leveraging estimated optical flow but also generating optical flow. Throughout all optical flow assisted model, Huang et al. consider it as part of objectives but results in less coherency[5], and Chen et al. estimate optical flow with an individual network at inference which slows down the whole system[4]. Here, this project tries to show that one single model is enough to capture both optical flow and style transformation, with split decoders and loss functions. Since only one single network is required, the efficiency is expected over Huang et al. and stylization performance is achieved comparably as the result of Chen et al.

IV. Methodology

This project aims to develop a new network structure with multi-task learning to achieve a stable, real-time and coherent video style converter. The goal of it is to deliver an end-to-end program that can run on GPU with real-time efficiency and high quality.

The whole project will be implemented on a computer specially equipped with an NVIDIA GTX 1080 Ti GPU, which can expedite the running speed much more than CPU. It will be implemented in Python with deep learning framework PyTorch, an excellent dynamic computational graph engine. The ground truth dataset for training our model is Sintel Optical Flow Dataset [10] from Max Planck Institute for Intelligent Systems.

The neural network model used in this project resembles previous network structure and utilizes encoder-decoder frameworks. As for multi-task learning, two successive
frames are sampled from dataset at each timestamp. These two frames will be encoded by two independent image encoders and the corresponding representations then are concatenated together. The ensembled features will pass through a shared encoder, and split to two decoders to generate the stylized image of next time frame and optical flow respectively. The loss function will be a linear combination of content loss, style loss and the difference between generated optical flow and ground truth, which is called temporal loss. Optimization method will be Adam [11] and the learning rate and momentum should be fine-tuned while training.

At the end of this project, a model which stylizes the video and predicts its optical flow at the same time will be expected. Especially, this model could work in a real-time manner, which means the resulted video should be generated more than thirty frames per second.

V. Feasibility assessment

a. Work package

Since the whole project is large in scope, all works are divided within the team. Chang is in charge of designing, implementing, and improving the model, as well as maintaining the project website. Derun will be following recent research progresses appear on CVPR, ICCV, and papers in arXiv that show intriguing results and are related to our project. Derun is also in charge of most of the writing work, as well as debugging and fine-tuning the model.

b. Risks

Since this project does not explicitly use similar networks as other existing works that share the same underlying methodologies, its approach has potential risk failing to achieve tantamount performance. However, using optical flow as one of the tasks can in another way help the model learn temporal consistency to some extent, and thus providing a guarantee of an acceptable accuracy.

VI. Schedule

From September to October, existing models will be implemented and results will be compared. The drawbacks will be analyzed so that the severity of these defects can be well studied. The machine learning network structure used in this project will then be proposed based on these studies.

From November to December, the initially designed details will be finalized and models will be implemented. Platforms, languages, and tools used will be finalized at
this stage. The design and coding style of the project will be discussed and guide the following experiments.

From January to February in the second semester, the main focus will be on optimisation and fine-tuning of the network structure. A renowned fact in machine learning area is that small changes of parameters can make a big difference in the final results. At this stage, previous works will be used for references to fine-tune our networks, thus achieving more desirable performance.

From March to May, the whole project will be wrapped up and results, including the finalized network structure and the end-to-end program, will be delivered. After the finalization of the network and algorithms, time will then be spent on summarization and documentation. A paper for computer vision conference can also be written as this time with description and impact of this work.

Reference