# Real-Time Video Arbitrary Style Transfer

*Mario Hany Hunter (48), Merit Victor (69)*

*Computer and Systems Engineering Department, Alexandria University, Alexandria, Egypt*

## I INTRODUCTION

Artistic Style Transfer has evolved greatly in the last years as a way to mimic human creative abilities. Many models have been proposed to style images. The main objective is how to make the machine learns the intricate structures and features from a style image and use them to transform an input into a target image. This process was thought to be reserved to humans solely, until very recently when many models achieved this goal with impressive results.

*Gatys et al.* [1] was one of the successful work done in the area of neural style transfer, which inspired many other researchers to follow in their footsteps. Their approach was simple, starting with white noise they solved an optimization problem that aims to produce content similar to the that of the original images while capturing the style of the style image. By defining two custom losses, the model was able to produce appealing results. The losses used in their approach were, namely, *Style loss* and *Content loss*. *Johnson et al.* [2] built upon this approach to solve the problem in a computationally efficient way, by computing perceptual losses instead. They are computed at the output of a layer of an image classification network, which essentially computes the losses of the high level features of the images.

This concept was then migrated to videos. Styling an entire video given a reference style image. This came with its own challenges. First, the computational constraints, which is a necessary requirement to produce results in real-time. Secondly, the coherent appearance of the video. Explicitly, when styling each frame independently from previous frames the results tends not to be exactly similar which causes what to be known as the flickering effect when rendered into a video sequence.

Those problems manifested to be areas of interest in many researches. Multiple milestones have been reached while solving each one of them, creating some sophisticated models influenced by the Image Style Transfer problem.

The first milestone, was to produced visually coherent styled videos which do not suffer from the flickering problem described before. *Ruder et al.* [3] tackled this problem by introducing additional loss terms to the objective function, the *temporal losses*. It essentially penalizes inconsistencies between frames in the short term and the long term. It suffered from very long runtime though, by making multiple iterations of forward and backward passes.

The second was tackling the computational constraints problem while maintaining coherence. This helped the model to produce results in real-time which made it possible to deploy the model in real life applications. However, that was of the cost of having the model being able to use an arbitrary style. The capabilities of the model styling was limited to that of one and only style, that was pre-trained. *Huang et al.* [4] and *Gao et al.* [5] were ones of those successfully solving the problem. Their approach was essentially to train a feed forward styling network, which is then used during inferences. They added additional constraints that helped preserving the temporal coherence between frames.

The next milestone to be sought is conceivably to achieve real-time styling for an arbitrary style given to the network as an input.

In this document, a new approach is proposed that can solve the problem of arbitrary style transfer for videos, i.e. can work with any style given at inference time, while maintaining real-time constraint and preserving coherence. The approach simply put tries to embed a style image into a vector space. By making use of the fact that the style images have similar structures as textures, the style vector network was built to predict these features and represent them in a single vector. Then, feeding this vector to another styling network that can style frames of the videos. The styling network is in essence an auto-encoder which uses the style vector to perform linear transformations on the layers activations. This trick can transform between different styles. The

transformation network itself is trained to preserve the content of the image and the coherence between consecutive frames. As a result, an arbitrary style can be fed to the network, and the frames are forward passed through the network to produce styled frames in real-time. In summary, the proposed approach is unique in the following aspects:

- It accepts as an input an arbitrary style, that can be embedded in vector space to use for styling.
- The feed forward network is trained to preserve the temporal consistency in the short-term and long-term.
- the inference require forward pass only which produce results in real-time.

## II RELATED WORK

### II-A *Image Style Transfer*

The first breakthrough in the problem of Artistic Style Transfer was proposed by *Gatys et al.* [1]. The problem was modeled as an energy optimization problem. It starts a picture of white noise and then tries to minimize the objective function

$$L_{total}(\ p,\ a,\ x) = \alpha L_{content}(\ p,\ x) + \beta L_{style}(\ a,\ x)$$

where $L_{content}$ defines the loss between the feature maps of the input image and the output at layer $l$ , and $L_{style}$ is defined to be the difference between the gram matrices at a layer $l$.

Even though the results of this approach was impressive it was computationally expensive. In [2], *Johnson et al* proposed a new approach. Instead of minimizing the loss at pixel level, it can make use of a pretrained ImageNet network such as *VGG*, and computes the losses at the high level features at layer $l$ of the VGG, which is used to train an Auto Encoder network that generates the stylized image from the input image. The stylized image along with the original image and the style image are fed to the VGG to compute the losses as defined by formulas in [1]. This improvement led to an increase of 3 orders of magnitude in run time.

Even though with these improvements the results needed multiple iteration until convergence and was not produced in real-time. The problem of Real-Time Arbitrary Image Style Transfer was tackled By *Ghiasi* [6]. Simply put, it trains a network to find *Style Embeddings*. Given a style image it predicts a style vector capturing the textures and features of the style. Then, this vector is fed to an auto-encoder network which uses it to transform its activations, which in turn leads to a different style output. The linear transformation is thus defined as

$$z = \gamma_s \frac{(z-\mu)}{\sigma} + \beta_s$$

where the $\gamma_s$ and $\beta_s$ constitute the style embedding vector. This led to transforming a style into another at inference time using only forward passes.

### II-B *Video Style Transfer*

Using any one of the previous approaches repeatedly and independently to each frame of a video does not yield a good result. This due to the fact that the optimization problem often converges to different local optima for each frame, leading to what is known to be a flickering effect, where the same object is represented differently at each frame of the output.

[3] tackled this problem by adding additional terms to the objective function that penalizes incoherence, thus guaranteeing short and long term consistency. This is done by penalizing the difference between the warped frame using an estimated optical flow and the past $j$ frames. The differences is masked at occluded regions. This way when a new object appears and it's newly generated, the model doesn't penalize the new creation. Their objective function was defined to be

$$L_{longterm}(p^{(i)}, a, x^{(i)}) = \alpha L_{content}(p^{(i)}, x^{(i)}) + \beta L_{style}(a, x^{(i)})$$
$$+\gamma \sum_{j \in J: i-j \geq 1} L_{temporal}(x^{(i)}, w^{i}_{i-j}(x^{(i-j)}), c^{(i-j,i)}_{long})$$

where $L_{content}$ and $L_{style}$ are defined as in *Gatys* [1], and $L_{temporal}$ is defined as per-pixel weighting of the loss between the image and the forward optical flow warp of the frame. Further, each frame is initialized using a warp function on the previous frame, which uses optical flow estimation produced by *DeepFlow*. The initialization and optimization is done over many passes. This, as *Gatys*, even though providing very good coherent results, suffers from very slow run times.

[4] and [5], both tackled the slow run time problem by using pre-trained style networks. Both used feed forward networks and minimized the temporal loss differently - i.e. using ground truth optical flow or estimated. The

feed forward algorithms are convenient for inference since they can compute predictions very fast.

## III NEED TO EXTEND

As aforementioned, no current model can produce real-time results for video style transfer for an arbitrary styles. This can easily be demanded by real life mobile applications where users can record videos and change their styles in real time with a user provided style. Or it can even be integrated in many video chat applications. Further, this represents a new milestone in the problem of Artistic video style transfer.

## IV METHODS AND PROCEDURES

Two approaches are proposed to solve the problem at hand. The first using a frame by frame styling using the model in [6]. The second to extend the model in [6] to work on videos and perserve the temporal coherence.

### IV-A *Frame by Frame Styling*

Since the model passes the images on a feed forward network, and was trained on high level features. It's conceivable that it will produce similar output for the same object, since it has the same high level feature representation across different frames. Thus, it can give a sense of coherence and temporal consistency.

The goal here is try this approach and provide qualitative analysis, along with a comparison with the real-time pretrained networks in [4] and [5] using their defined error quantities such as the stability error.

### IV-B *Knowledge Distillation*

The model in [6], is already trained to style arbitrary styles on images. Then, by continuing its training to minimize not only content and style losses, but also temporal losses as defined in [5] initialized with the published weights, the model can converge quickly to perserve temporal consistency while keeping the content/style loss at its minimal value.

The training adopts a two-frame synergic mechanism, two runs are used to compute the stylized output of two frames, and then are fed to the loss function to compute the gradients and update the network. The temporal loss is measured on multiple levels, using ground truth optical flows, which gives the model higher accuracy. First, the difference between the output and the warped previous output - generated using ground truth optical flows, is computed. Then, the difference in the Y channel of the input frame and the warped previous frame is computed. This quantity helps in cases where the constant brightness do not hold.The $L_2$ distance is computed of the difference between the two quantities and masked in occluded regions using a ground truth mask. The formula can be shown as,

$$L_{temp,o}(t1,t) = \sum_c \frac{1}{D} M_t ||(O_t - W_t(O_{t-1}))_c - (I_t - W_t(I_{t-1}))_Y||^2$$

This as shown in [5] produced better results than other real time models, in terms of stability. Further, it was shown that it produced competitive quantitative results as that produced by [3].

By enforcing this loss and retraining the transformer model in the arbitrary style transfer model for images, should produce higher temporal consistency.

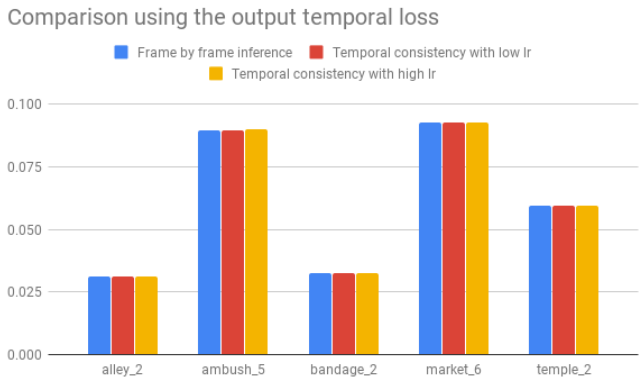


**Fig. 1.** Comparison using the temporal loss term.

## V RESULTS

Using the weights produced by the Arbitrary Style Transfer for images, the stability error was an order of magnitude higher than that of State-Of-The-Art models.

Continuing the training, pushed the model to decolorize the network and smudge contents a little. This is due to lack of hyper parameters tuning caused by high computational power needed and environment problems. A comparison between the state-of-the-art is depicted in fig.1 and fig.2.

The previous figures show that quantitatively there is no much difference in the results between the two experimented approaches.

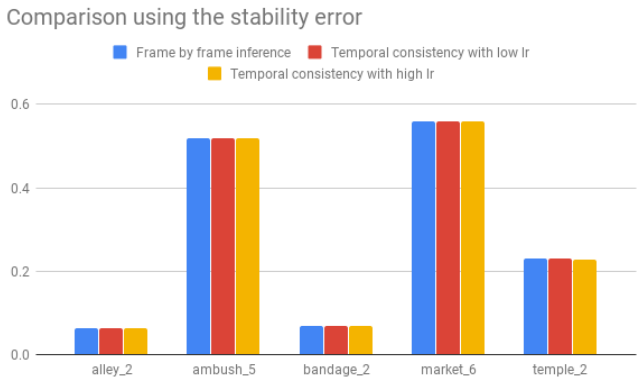


**Fig. 2.** Comparison using the stability error.

**Fig. 3.** The original frame.

However, qualitative analysis shows that the videos are visually pleasant and quite stable and works for any arbitrary style. This can be seen from fig.3 and fig.4.

The objects were styled in the same manner across frames, same textures and colors. However, color strokes may get smaller or larger. While this gives the video a painting feeling, it was penalized by stability error. This urges the need to create a new metric for comparison other than the stability error.

Further, the model produces around 10 frames per second. This is before performing knowledge distillation to produce smaller models and exporting TF-lite version for for mobile which is expected to make the model even faster meeting real-time benchmarks easily.

**Fig. 4.** The styled frame.

## VI FUTURE WORK

The temporal loss described in [5] can be further extended to include long-term consistency by training the past $j$ frames as described in [3]. The temporal loss is masked until the pixel finds its first traceable occurrence in the previous frames, thus enforcing shapes that gets covered and appear again to have the same representation in the later frames.

## REFERENCES

[1] L. A. Gatys, A. S. Ecker, and M. Bethge, "A neural algorithm of artistic style," 2015.

[2] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," 2016.

[3] M. Ruder, A. Dosovitskiy, and T. Brox, "Artistic style transfer for videos," *Pattern Recognition*, p. 26âĂŞ36, 2016. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-45886-1_3

[4] H. Huang, H. Wang, W. Luo, L. Ma, W. Jiang, X. Zhu, Z. Li, and W. Liu, "Real-time neural style transfer for videos," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[5] C. Gao, D. Gu, F. Zhang, and Y. Yu, "Reconet: Real-time coherent video style transfer network," 2018.

[6] G. Ghiasi, H. Lee, M. Kudlur, V. Dumoulin, and J. Shlens, "Exploring the structure of a real-time, arbitrary neural artistic stylization network," 2017.