Bringing Impressionism to Life with Neural Style Transfer in *Come Swim*

Bhautik J Joshi*
Research Engineer, Adobe

Kristen Stewart
Director, *Come Swim*

David Shapiro
Producer, Starlight Studios

Figure 1: *Usage of Neural Style Transfer in Come Swim; left: content image, middle: style image, right: upsampled result. Images used with permission, (c) 2017 Starlight Studios LLC & Kristen Stewart.*

Abstract

Neural Style Transfer is a striking, recently-developed technique that uses neural networks to artistically redraw an image in the style of a source style image. This paper explores the use of this technique in a production setting, applying Neural Style Transfer to redraw key scenes in *Come Swim* in the style of the impressionistic painting that inspired the film. We document how the technique can be driven within the framework of an iterative creative process to achieve a desired look, and propose a mapping of the broad parameter space to a key set of creative controls. We hope that this mapping can provide insights into priorities for future research.

Keywords: style transfer, rendering, applied computer graphics

Concepts: •**Computing methodologies** → *Computer graphics; Image-based rendering;* •**Applied computing** → *Media arts;*

1 Introduction

In *Image Style Transfer Using Convolutional Neural Networks*, Gatys et al [Gatys et al. 2015] outline a novel technique using convolutional neural networks to re-draw a content image in the broad artistic style of a single style image. A wide range of implementations have been made freely available [Liu 2016; Johnson 2015; Athalye 2015], based varyingly on different neural network evaluators such as Caffe [Jia et al. 2014] and Tensorflow [Abadi and et al 2016] and wrappers such as Torch and PyCaffe [Bahrampour et al. 2015]. There has been a strong focus on automatic techniques, even with extensions to coherently process video [Ruder et al. 2016]. Workflows exist that permit painting and masking to guide the transfer more explicitly, as well as to reduce the size of the neural net for evaluation to significantly speed up execution [Ulyanov et al. 2016]. Creative control of these implementations are broadly handled via parameter adjustment.

Using the technique with a small set of carefully picked style examples is a natural optimization, allowing Neural Style Transfer to execute efficiently and predictably. In a production setting, however, a great deal of creative control is needed to tune the result, and a rigid set of algorithmic constraints run counter to the need for this creative exploration. While early investigations to better map the low-level neural net evaluations to stylistic effects are underway [Li et al. 2017], in our paper we focused on examining the higher-level parameter space for Neural Style Transfer and found a set of working shortcuts to map them to a reduced but meaningful set of creative controls.

2 Realizing Directorial Intent

Come Swim is a poetic, impressionistic portrait of a heartbroken man underwater. The film itself is grounded in a painting (figure 2 by coauthor Kristen Stewart) of man rousing from sleep. We take a novel artistic step by applying Neural Style Transfer to redraw key scenes in the movie in the style of the painting, realizing them almost literally painting that underpins the film.

The painting itself evokes the thoughts an individual has in the first moments of waking (fading in-between dreams and reality), and this theme is explored in the introductory and final scenes where this technique is applied. This directly drove the look of the shot, leading us to map the emotions we wanted to evoke to parameters in the algorithm as well as making use of more conventional techniques in the 2D compositing stage.

Figure 2: *The painting that drove the story and aesthetic for Come Swim. Image used with permission, (c) 2017 Starlight Studios LLC & Kristen Stewart.*

---

*e-mail:bjoshi@gmail.com

3 Initial Setup

Initial development of the look on the shots was arguably the longest and most difficult part of the process. The novelty of the technique gave a false sense of a high-quality result early on; seeing images redrawn as paintings is compelling enough that nearly any result seems passable. Repeated viewings were needed to factor out novelty and to focus on the primary metric we use to measure success - is the look in service of the story?

3.1 Pipeline for Evaluation

Given that the technique has not been used often in formal VFX production, we had to establish a baseline. The parameter space and corresponding creative possibilities are huge, so initial iteration was focused on shrinking down the number of possibilities down to relevant options.

Whilst the technique appears to deliver impressionism on demand, steering the technique in practice is difficult. We examined directable versions of the technique [Ulyanov et al. 2016] but time pressures meant that we were not tooled for this approach and preferred the whole-frame consistency - and serendipity - that came from applying the technique uniformly to the entire frame. We focused on techniques we could prototype locally and with modest hardware [Liu 2016]. Strong coherency between frames [Ruder et al. 2016] was not desired creatively for the sequences considered, which permitted an extra degree of latitude for developing the look.

For the neural network evaluation, we initially tried the `googlenet` [Szegedy et al. 2015] and `vgg19` [Simonyan and Zisserman 2014] neural nets; `vgg19` execution times were too high, and `googlenet` did not give the aesthetics we were seeking. We made an early choice to settle on `vgg16` [Simonyan and Zisserman 2014] since it provided the right balance between execution time and the quality of result we desired.

Early results were achieved by working on a single, representative frame to iterate on the look. Its also worth noting the near linear relationship between style transfer ratio and image size; working with an image that has half the image dimensions meant roughly halving the style transfer ratio. Faster, coarse iteration could then be performed on lower-resolution frames.
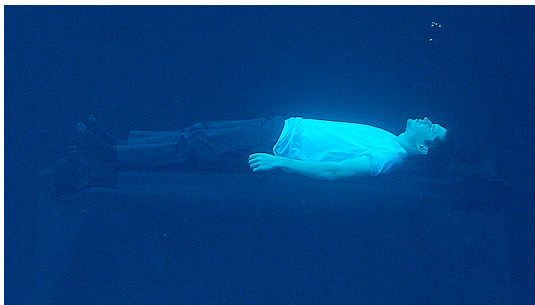
3.2 Iterating on the style image

Figure 3: *The reference image used for during tuning of the style image. Image used with permission, (c) 2017 Starlight Studios LLC & Kristen Stewart.*

An early observation during work on the setup of the style transfer was that the effect could be directed by changing the content of the style image itself. For the iterations presented here, we used a fixed content image as shown in figure 3.

We initially controlled for color and texture by closely cropping the style image. This set up a baseline for the look of texture transfer (figure 4, left). We experimented with the number of iterations in the algorithm, optimizing for speed to see how much it caused the look to deviate, giving us a reasonable range for this parameter. We added larger blocks of color and texture to the style image (figure 4, right). As the texture is built, other parameters for the style transfer can be more finely tuned and brought into a smaller, more manageable range for iteration on look.
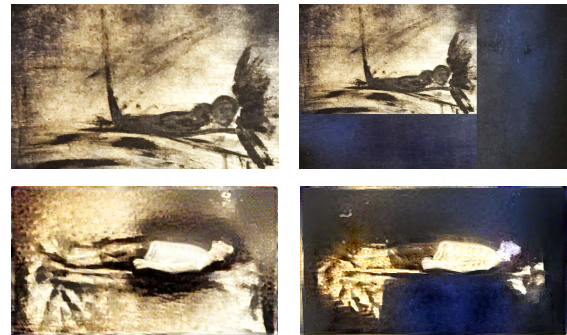

Figure 4: *Initial style image (left-top) and style transferred result (left-bottom). Larger blocks of the style image were added (right-top) generating a richer result (right-bottom).*

The quality of the input style image is also critical for achieving a high quality result. The painting that we used for the style image is texturally complex - the paint itself is reflective and is applied in high-frequency spatter across the canvas. A low-quality cellphone shot of the canvas drove initial work on the shots (figure 5, left). This low-quality image blew out the reflective highlights and flattened out the subtle texture on the paint, and this same artifact is can be seen in the style transfer. Replacing the style image with a well-lit, high-quality photograph significantly improves the result, allowing for more faithful transfer of the subtleties in the contrast, color and texture (figure 5, right).
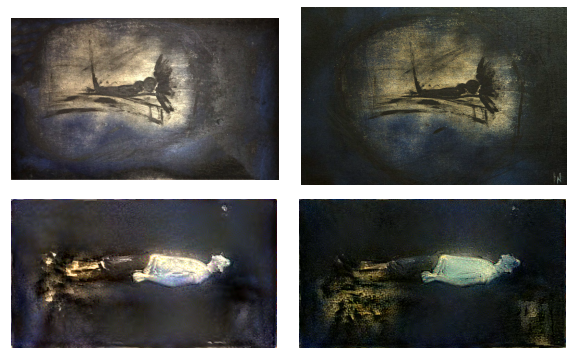

Figure 5: *The quality of the style image is critical - a low-quality photograph of the painting (top-left) leads to poor transfer of texture and contrast (bottom-left); a higher quality photograph (top-right) leads to better detail and texture preservation (bottom-right).*

3.3 Mapping look to parameters

Once the pipeline was initialized, we identified two major parameters as it applied to our look - the style transfer ratio and the number of iterations. Increasing the number of iterations increases the degree of texture that is transferred; in general it needs to be at least

128 to prevent noticeable artifacts in the content image. We found that in practice 256 iterations were sufficient for the style image we were using; much past this only produced negligible improvements in (the subjectively assessed) quality of the style transfer.

A fixed number of iterations, led to a near-constant execution time per-frame. Experimenting with the style transfer ratio led us to conclude that it needed to be exponent form for meaningful creative exploration. Subjectively, this exponential form gave us a useful measure of unrealness, u, a rough way to map how impressionistic the style transferred image looked:

$$style\ transfer\ ratio = 10^u \qquad (1)$$

As shown in figure 6, this mapping allowed us to iterate on the intensity of the style transfer. To achieve the artistic effect desired and to magnify the effect artistically, the style-transferred footage was cross-dissolved in and out of the original footage.
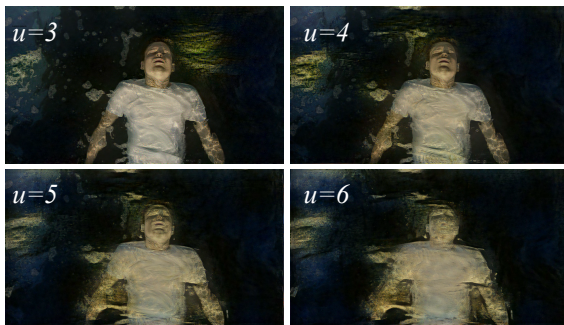

Figure 6: *Increasing the value of **u** gave us a control to fine-tune the degree of subjectively-measured impressionism present in the style transfer*

4   Production Notes

As mentioned earlier, we used low-resolution previews computed locally on modest hardware on the CPU to set up the production pipeline. However, to get us to the needed level of quality we needed a scalable solution that provided GPU resources on demand - in this case, we chose to use GPU instances on Amazon EC2.

We used custom ubuntu images on `g2.2xlarge` virtual instances to provide us with the necessary computing power to achieve the result in a reasonable amount of time. However, the process itself is fairly heavily limited by the amount of available GPU memory; in the configuration we used, we were realistically only able to produce 1024px wide images; the show called for 2048px images.

To work around this, we upscaled the images to 2048px resolution and denoised the images aggressively. This functioned as a type of bandpass filter, and worked well to achieve the sketch-based look we were after. Overall, we ended up with a compute time of around 40 minutes per frame per instance used.

5   Conclusion and Future Explorations

Far from being automatic, Neural Style Transfer requires many creative iterations when trying to work towards a specific look for a shot. The parameter space is very large - careful, structured guidance through this in collaboration with the creative leads on the project is needed to steer it towards a repeatable, visually pleasing result.

Gradually building the style image in terms of texture and color blocks to move towards the desired result can help frame the process and set baselines for look and execution time. Quick iteration can also be achieved by using low-resolution images with appropriately scaled style transfer ratios allows for representative results before fine-tuning with much longer-running, high-resolution iterations.

An initial future optimization that would seem obvious first would be to use an optimized, feed forward model [Ulyanov et al. 2016]. However, the single style transfer ratio proposed in such models would not meet the needs for flexibility around this parameter for creative iteration in shots. A more promising approach would be to a pastiche-based style transfer model [Dumoulin et al. 2016], but instead of using several different style images, use the same style image with a spread of style transfer ratios.

References

ABADI, M., AND ET AL, A. A. 2016. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *CoRR abs/1603.04467*.

ATHALYE, A., 2015. Neural style. https://github.com/anishathalye/neural-style. commit 4bfb9efb1f0e955922889187727e33185e97d934.

BAHRAMPOUR, S., RAMAKRISHNAN, N., SCHOTT, L., AND SHAH, M. 2015. Comparative study of caffe, neon, theano, and torch for deep learning. *CoRR abs/1511.06435*.

DUMOULIN, V., SHLENS, J., AND KUDLUR, M. 2016. A learned representation for artistic style. *CoRR abs/1610.07629*.

GATYS, L. A., ECKER, A. S., AND BETHGE, M. 2015. A neural algorithm of artistic style. *CoRR abs/1508.06576*.

JIA, Y., SHELHAMER, E., DONAHUE, J., KARAYEV, S., LONG, J., GIRSHICK, R., GUADARRAMA, S., AND DARRELL, T. 2014. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*.

JOHNSON, J., 2015. neural-style. https://github.com/jcjohnson/neural-style.

LI, Y., WANG, N., LIU, J., AND HOU, X. 2017. Demystifying Neural Style Transfer. *ArXiv e-prints* (Jan.).

LIU, F., 2016. style-transfer. https://github.com/fzliu/style-transfer.

RUDER, M., DOSOVITSKIY, A., AND BROX, T. 2016. *Artistic Style Transfer for Videos*. Springer International Publishing, Cham, 26–36.

SIMONYAN, K., AND ZISSERMAN, A. 2014. Very deep convolutional networks for large-scale image recognition. *CoRR abs/1409.1556*.

SZEGEDY, C., LIU, W., JIA, Y., SERMANET, P., REED, S., ANGUELOV, D., ERHAN, D., VANHOUCKE, V., AND RABINOVICH, A. 2015. Going deeper with convolutions. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1–9.

ULYANOV, D., LEBEDEV, V., VEDALDI, A., AND LEMPITSKY, V. 2016. Texture networks: Feed-forward synthesis of textures and stylized images. In *International Conference on Machine Learning (ICML)*.