Exploring the Structure of a Real-time, Arbitrary Neural Artistic Stylization Network

Golnaz Ghiasi[1]
golnazg@google.com
Honglak Lee[1]
honglak@google.com
Manjunath Kudlur[1]
keveman@google.com
Vincent Dumoulin[2]
vi.dumoulin@gmail.com
Jonathon Shlens[1]
shlens@google.com

[1] Google Brain
1600 Amphitheatre Parkway
Mountain View, CA, USA

[2] MILA, Université de Montréal
Québec, Canada

Abstract

In this paper, we present a method which combines the flexibility of the neural algorithm of artistic style with the speed of fast style transfer networks to allow real-time stylization using any content/style image pair. We build upon recent work leveraging conditional instance normalization for multi-style transfer networks by learning to predict the conditional instance normalization parameters directly from a style image. The model is successfully trained on a corpus of roughly 80,000 paintings and is able to generalize to paintings previously unobserved. We demonstrate that the learned embedding space is smooth and contains a rich structure and organizes semantic information associated with paintings in an entirely unsupervised manner.

1 Introduction

Elmyr de Hory gained world-wide fame by forging thousands of pieces of artwork and selling them to art dealers and museums [13]. The forger's skill is a testament to the human talent and intelligence required to reproduce the artistic details of a diverse set of paintings. In computer vision, much work has been invested in teaching computers to likewise capture the artistic style of a painting with the goal of conferring this style in arbitrary photographs in a convincing manner.

Early work in this effort in computer vision arose out of visual texture synthesis. Such work focused on building non-parametric techniques for "growing" visual textures one pixel [5, 27] or one patch [4, 17] at a time. Interestingly, Efros et al. (2001) [4] demonstrated that one may transfer a texture to an arbitrary photograph to confer it with the stylism of a drawing. Likewise, Hertzmann et al. (2001) [11] demonstrated a non-parametric technique for imbuing an arbitrary filter to an image based on pairs of unfiltered and filtered images.

Figure 1: Stylizations produced by our network trained on a large corpus of paintings and textures. The left-most column shows four content images. Left: Stylizations from paintings in training set on paintings (4 left columns) and textures (4 right columns). Right: Stylizations from paintings never previously observed by our network.

In parallel to non-parametric approaches, a second line of research focused on building parametric models of visual textures constrained to match the marginal spatial statistics of visual patterns [15]. Early models focused on matching the marginal statistics of multi-scale linear filter banks [6, 21]. In recent years, spatial image statistics gleaned from intermediate features of state-of-the-art image classifiers [23] proved superior for capturing visual textures [8]. Pairing a secondary constraint to preserve the content of an image – as measured by the higher level layers of the same image classification network – extended this idea to artistic style transfer [9] (see also [10]).

Optimizing an image or photograph to obey these constraints is computationally expensive and contains no learned representation for artistic style. Several research groups addressed this problem by building a secondary network, i.e., *style transfer network*, to explicitly learn the transformation from a photograph a particular painting style [14, 16, 25]. Although this method confers computational speed, much flexibility is lost: a single style transfer network is learned for a single painting style and a separate style transfer network must be built and trained for each new painting style.

Most crucially, by partitioning the style transfer problem customized for a specific style of painting, these methods avoid the critical ability to learn a *shared* representation across paintings. Recent work by Dumoulin et al. [3] demonstrated that the manipulation of the normalization parameters was sufficient to train a single style transfer network across 32 varied painting styles. Such a network distilled the artistic style into a roughly 3000 dimensional space that is regular enough to permit smooth interpolation between these painting styles. Despite the promise, this model can cover only a limited number of styles and cannot generalize well to an unseen style. In this work, we extend these ideas further by building a style transfer network trained on about 80,000 painting and 6,000 visual textures. We demonstrate that this network can generalize to capture and transfer the artistic style of paintings never previously observed by the system (see Figure 1). Our contributions in this paper include:

1. Introduce a new algorithm for fast, arbitrary artistic style transfer trained on 80,000 paintings that can operate in real time on never previously observed paintings.

2. Represent all painting styles in a compact embedding space that captures features of the semantics of paintings.

3. Demonstrate that training with a large number of paintings uniquely affords the model

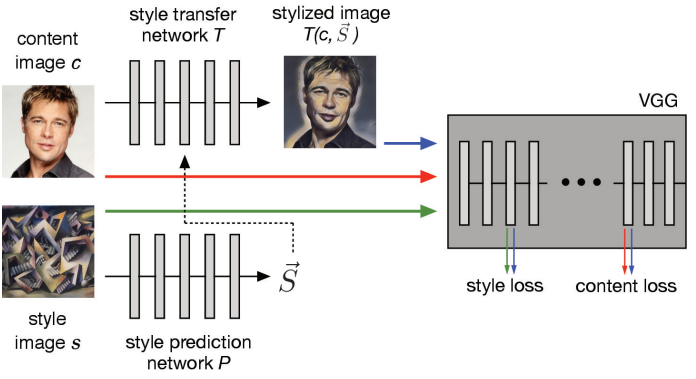

Figure 2: Diagram of model architecture. The style prediction network $P(\cdot)$ predicts an embedding vector $\vec{S}$ from an input style image, which supplies a set of normalization constants for the style transfer network. The style transfer network transforms the photograph into a stylized representation. The content and style losses [9] are derived from the distance in representational space of the VGG image classification network [23]. The style transfer network largely follows [8] and the style prediction network largely follows the Inception-v3 architecture [24].

the ability to predict styles never previously observed.

4. Embedding space permits novel exploration of artistic range of artist.

2 Methods

Artistic style transfer may be defined as creating a stylized image x from a content image c and a style image s. Typically, the content image c is a photograph and the style image s is a painting. A neural algorithm of artistic style [9] posits the content and style of an image may be defined as follows:

- Two images are similar in content if their high-level features as extracted by an image recognition system are close in Euclidean distance.
- Two images are similar in style if their low-level features as extracted by an image recognition system share the same spatial statistics.

The first definition is motivated by the observation that higher level features of pretrained image classification systems are tuned to semantic information in an image [14, 19, 28]. The second definition is motivated by the hypothesis that a painting style may be regarded as a visual texture [4, 9, 11]. A rich literature suggests that repeated motifs representative of a visual texture may be characterized by lower-order spatial statistics [6, 15, 21]. Images with identical lower-order spatial statistics appear perceptually identical and capture a visual texture [6, 8, 21, 26]. Assuming that a visual texture is spatially homogeneous implies that the lower-order spatial statistics may be represented by a Gram matrix expressing the spatially-averaged correlations across filters within a given layer's representation [6, 8, 21].

The complete optimization objective for style transfer may be expressed as

$$\min_x \mathcal{L}_c(x,c) + \lambda_s \mathcal{L}_s(x,s) \tag{1}$$

where $\mathcal{L}_c(x,c)$ and $\mathcal{L}_s(x,s)$ are the content and style losses, respectively and λ_s is a Lagrange multiplier weighting the relative strength of the style loss. We associate lower-level and higher-level features as the activations within a given set of lower layers $\mathcal{S}$ and higher layers $\mathcal{C}$ in an image classification network. The content and style losses are defined as

$$\mathcal{L}_s(x,s) = \sum_{i \in \mathcal{S}} \frac{1}{n_i} || \mathcal{G}[f_i(x)] - \mathcal{G}[f_i(s)] ||_F^2 \tag{2}$$

$$\mathcal{L}_c(x,c) = \sum_{j \in \mathcal{C}} \frac{1}{n_j} || f_j(x) - f_j(c) ||_2^2 \tag{3}$$

where $f_l(x)$ are the network activations at layer l, n_l is the total number of units at layer l and $\mathcal{G}[f_l(x)]$ is the Gram matrix associated with the layer l activations. The Gram matrix is a square, symmetric matrix measuring the spatially averaged correlation structure across the filters within a layer's activations.

Early work focused on iteratively updating an image to synthesize a visual texture [6, 8, 21] or transfer an artistic style to an image [9]. This optimization procedure is slow and precludes any opportunity to learn a representation of a painting style. Subsequent work introduced a second network, a *style transfer network* $T(\cdot)$, to learn a transformation from the content image c to its artistically rendered version $\hat{x}$ (i.e., $\hat{x} = T(c)$) [14, 16, 26]. The style transfer network is a convolutional neural network formulated in the structure of an encoder/decoder [14, 26]. The training objective is the combination of style loss and content loss obtained by replacing x in Eq. 1 with the network output $T(c)$. The parameters of the style transfer network are trained by minimizing this objective using a corpus of photographic images as content. The resulting network may artistically render an image dramatically faster, but a separate network must be learned for each painting style.

Training a new network for each painting is wasteful because painting styles share common visual textures, color palettes and semantics for parsing the scene of an image. Building a style transfer network that shares its representation across many paintings would provide a rich vocabulary for representing any painting. A simple trick recognized in [3] is to build a style transfer network as a typical encoder/decoder architecture but specialize the normalization parameters specific to each painting style. This procedure, termed *conditional instance normalization*, proposes normalizing each unit's activation z as

$$\tilde{z} = \gamma_s \left(\frac{z - \mu}{\sigma} \right) + \beta_s \tag{4}$$

where μ and σ are the mean and standard deviation across the spatial axes in an activation map [26]. γ_s and β_s constitute a linear transformation that specify the learned mean (β_s) and learned standard deviation (γ_s) of the unit. This linear transformation is unique to each painting style s. In particular, the concatenation $\vec{S} = \{\gamma_s, \beta_s\}$ constitutes a roughly 3000-d embedding vector representing the artistic style of a painting. We denote this style transfer network as $T(\cdot, \vec{S})$. The set of all $\{\gamma_s, \beta_s\}$ across $N = 32$ paintings constitute 0.2% of the network parameters. Dumoulin et al. [3] showed that such a network provides a fast stylization of artistic styles and the embedding space is rich and smooth enough to allow users to combine the painting styles by *interpolating* the learned embedding vectors of 32 styles.

Although an important step forward, this "N-style network" is still limited compared to the original optimization-based technique [9] because the network is limited to only work on the styles explicitly trained on. The goal of this work is to extend this model to (1) train

on $N \gg 32$ styles and (2) perform stylizations for unseen painting styles never previously observed. The latter goal is especially important because the degree to which the network generalizes to unseen painting styles measures the degree to which the network (and embedding space) represents the true breadth and diversity of all painting styles.

In this work, we propose a simple extension in the form of a *style prediction network* $P(\cdot)$ that takes as input an *arbitrary* style image s and *predicts* the embedding vector $\vec{S}$ of normalization constants, as illustrated in Figure 2. The crucial advantage of this approach is that the model can generalize to an unseen style image by predicting its proper style embedding at test time. We employ a pretrained Inception-v3 architecture [24] and compute the mean across each activation channel of the Mixed-6e layer which returns a feature vector with the dimension of 768. Then we apply two fully connected layers on top of it to predict the final embedding $\vec{S}$. The first fully connected layer is purposefully constructed to contain 100 units which is substantially smaller than the dimensionality of $\vec{S}$ in order to compress the representation. We find it sufficient to jointly train the style prediction network $P(\cdot)$ and style transfer network $T(\cdot)$ on a large corpus of photographs and paintings.

A parallel work has proposed another method for fast, arbitrary style transfer in real-time using deep networks [12]. Briefly, Huang et al (2017) employ the same transformation (Equation 4) to normalize activation channels, however they calculate γ_s and β_s as the mean and standard deviation across the spatial axes of an encoder network applied to a style image. Although the style transformation is simpler, it provides a fixed heuristic mapping from style image to normalization parameters, whereas our method learns the mapping from the style image to style parameters directly. Our experimental results indicate that the increased flexibility achieves better objective values in the optimization.

3 Results

We train the style prediction network $N(\cdot)$ and style transfer network $T(\cdot)$ on the ImageNet dataset as a corpus of training content images and the Kaggle *Painter By Numbers* (PBN) dataset[1], consisting of 79,433 labeled paintings across many genres, as a corpus of training style images. Additionally, we train the model when *Describable Textures Dataset* (DTD) is used as the corpus of training style images. This dataset consists of 5,640 images labeled across 47 categories [2]. In both cases, we agument the training style images. We randomly flip, rescale, crop the images and change the hue and contrast of them. We present our results on both training style dataset.

3.1 Trained network predicts arbitrary painting and texture styles.

Figure 1 (left) shows stylizations from the network trained on the DTD and the PBN datasets. The figure highlights a number of stylizations across a few photographs. We note that the networks were trained jointly and unlike previous work [3, 14], it was unnecessary to select a unique Lagrange multiplier λ_s for each painting style. That is, a single weighting of style loss suffices to produce reasonable results across all painting styles and textures.

Importantly, we employed the trained networks to predict a stylization for paintings and textures never previously observed by the network (Figure 1, right). Qualitatively, the artistic stylizations appear to be indistinguishable from stylizations produced by the network on

[1] https://www.kaggle.com/c/painter-by-numbers

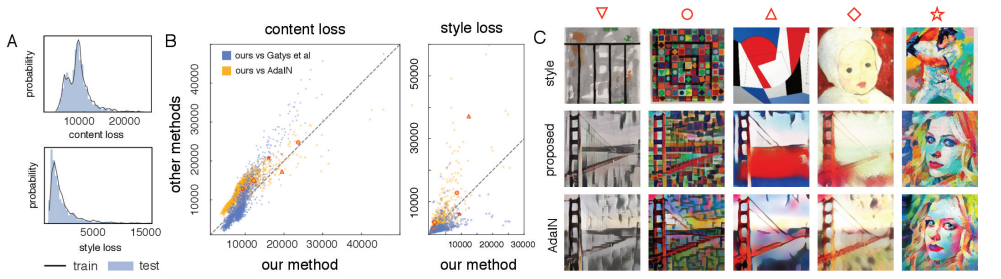

Figure 3: Generalization to unobserved painting styles. A. Distribution of style and content loss for stylization using observed and unobserved paintings from PBN training set. B. Comparison of style and content loss between proposed method, direct optimization [9] (blue) and AdaIN [12] (yellow). C. Sample images demonstrating stylization applied between proposed method and AdaIN [12] for selected points in panel B.

actual paintings and textures the network was trained against. We took this as an encouraging sign that the network learned a general method for artistic stylization that may be applied for arbitrary paintings and textures. In the following sections we quantify this behavior and measure the limits of this generalization.

3.2 Generalization to unobserved paintings.

Figure 1 indicates that the model is able to predict stylizations for paintings and textures never previously observed that are qualitatively indistinguishable from the stylizations on trained paintings and textures. In order to quantify this observation, we train a model on the PBN dataset and calculate the distribution of style and content losses across 2 photographs for 1024 observed painting styles (Figure 3A, black) and 1024 unobserved painting styles (Figure 3A, blue). The distribution of losses for observed styles (style: mean = $2.08e4 \pm 2.50e4$; content: mean = $8.92e4 \pm 3.13e4$) is largely similar to the distribution across unobserved styles (style: mean = $1.95e4 \pm 3.73e4$; content: mean = $8.94e4 \pm 3.55e4$). This indicates that the method performs stylizations on observed paintings with nearly equal fidelity as measured by the model objectives for unobserved styles. Importantly, if we train the model on a distinct but rich visual textures dataset (DTD) and test the stylizations on unobserved paintings from PBN, we find that the model produces similar artistic stylizations both quantitatively (style: mean = $2.67e4 \pm 6.49e4$; content: mean = $8.76e4 \pm 3.55e4$) and qualitatively (in terms of visual inspection). Due to space constraints, we provide detailed analysis in the supplementary material.

We next asked how well the learned networks perform on unobserved painting styles when compared to the original optimization-based method [9]. Figure 3B plots the content and style loss objectives for our proposed method (x-axis) and [9] (blue points). Note that even though [9] directly optimizes for these two objectives, the proposed method obtains content and style losses that are comparable (style: $1.95e4$ vs $1.12e4$; content: $8.94e4$ vs $9.09e4$). These results indicate that the learned representation is able to achieve an objective comparable to one obtained by direct optimization on the image itself.

We additionally compared our proposed method against a parallel work to perform fast, arbitrary stylization termed *AdaIN* [12]. We found that our proposed method achieved lower content and style loss. Specifically, (style: $1.95e4$ vs $2.56e4$; content: $8.94e4$ vs $12.3e4$). In

addition, paired t-test showed that these differences are statistically significant (style: p-value of 1.9×10^{-9} with t-statistic of -6.04; content: p-value of 0.0 with t-statistic of -91.9), indicating that our proposed model achieved consistently better dual objectives (Figure 3B, yellow points). See Figure 3C for a comparison of each method.

Figure 4 shows how the generalization ability of the model (measured in terms of style loss) is related to the proximity to training examples. Specifically, we plot style loss on unobserved paintings versus the minimum L_2 distance between the Gram matrix of unobserved painting and the set of all Gram matrices in the training dataset of paintings. The plot shows clear positive correlation ($r^2 = 0.9$), which suggests that our model achieves lower style loss when the unobserved image is similar to some of the training examples in terms of the Gram matrix. More discussion of this figure is found in the supplementary material.

Figure 4: Ability to generalize vs. proximity to training examples

3.3 Scaling to large numbers of paintings is critical for generalization.

A critical question we next asked was what endows these networks with the ability to generalize to paintings not previously observed. We had not observed this ability to generalize in previous work [3]. A simple hypothesis is that the generalization is largely due to the fact that the model is trained with a far larger number of paintings than previously attempted. To test this hypothesis, we trained style transfer and style prediction networks with increasing numbers of example painting styles without data augmentation. Figure 5A reports the distribution of content and style loss on unobserved paintings for increasing numbers of paintings.

First, we asked whether the model is better able to stylize photographs based on paintings in the training set by dint of having trained on larger numbers of paintings. Comparing leftmost and right-most points of the dashed curves in Figure 5A for the content and style loss indicate no significant difference. Hence, the quality of the stylizations for paintings in the training set do not improve with increasing numbers of paintings.

We next examined how well the model is able to generalize when trained on increasing numbers of painting styles. Although the content loss is largely preserved in all networks, the distribution of style losses is notably higher for unobserved painting styles and this distribution does not asymptote until roughly 16,000 paintings. Importantly, after roughly 16,000 paintings the distribution of content and style loss roughly match the content and style loss for the trained painting styles. Figure 5B shows three pairings of content and style images that are unobserved in the training data set and the resulting stylization as the model is trained on increasing number of paintings (Figure 5C). Training on a small number of paintings produces poor generalization whereas training on a large number of paintings produces reasonable stylizations on par with a model explicitly trained on this painting style.

3.4 Embedding space captures semantic structure of styles.

The style transfer model represents all paintings and textures in a style embedding vector $\vec{S}$ that is 2758 dimensional. The style prediction network predicts $\vec{S}$ from a lower dimensional representation (i.e., bottleneck) containing only 100 dimensions.

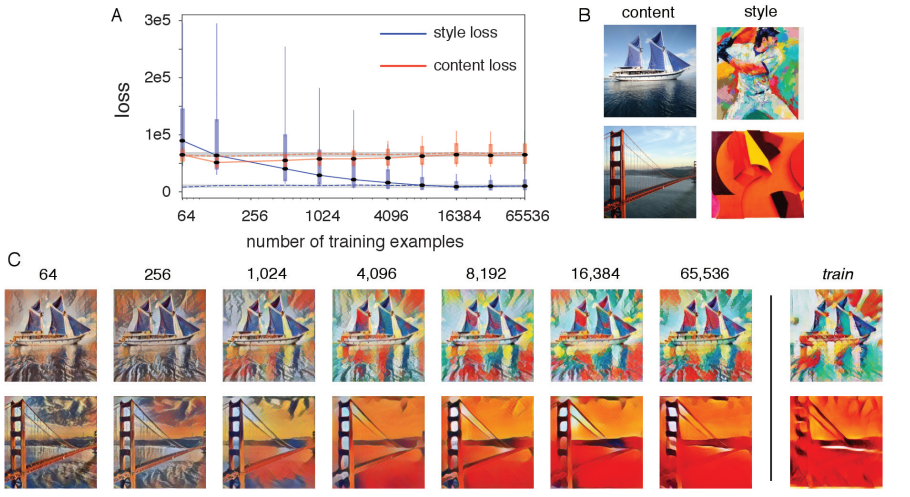

Figure 5: Training on a large corpus of paintings is critical for generalization. A. Distribution of style and content loss for stylizations applied to unseen painting styles for proposed method trained on increasing numbers of painting styles. Solid line indicates median with box showing $\pm 25\%$ quartiles and whiskers indicating 10% and 90% of the cumulative distributions. Dashed line and gray region indicate the mean and range of the corresponding losses for training images. Three sample pairs of content and style images (B) and the resulting stylization with the proposed method as the method is trained on increasing numbers of paintings (top number). For comparison, final column in (B) highlights stylizations for a model trained explicitly on the these styles.

Given the compressed representation for all artistic and texture styles, one might suspect that the network would automatically organize the space of artistic styles in a perceptually salient manner. Furthermore, the degree to which this unsupervised representation of artistic style matches our semantic categorization of paintings.

We explore this question by qualitatively examining the low dimensional representation for style internal to the style prediction network. A 100 dimensional space is too large to visualize, thus we employ the t-SNE dimensional reduction technique to reduce the representation to two dimensions [18]. Note that t-SNE will necessarily distort the representation significantly in order compress the representation to small dimensionality, thus we restrict our analysis to qualitative description.

Figure 6A (left) shows a two-dimensional t-SNE representation on a subset of 800 textures across 10 human-labeled categories. One may identify that regions of the embedding space cluster around perceptually similar visual textures: the bottom-right contains a preponderance of waffles; the middle contains many checkerboard patterns; top-center contains many zebra-like patterns. Figure 6B (left) shows the same representation for a subset of 3768 paintings across 20 artists. Similar clustering behavior may be observed across colors and spatial structure as well.

The structure of the low dimensional representation does not just contain visual similarity but also reflect semantic similarity. To highlight this aspect, we reproduce the t-SNE plot but replace the individual images with a human label (color coded). For the visual texture embedding (Figure 6A) we display a metadata label associated with each human-described texture. For the painting embedding (Figure 6B) we display the name of the artist for each painting. Interestingly, we find that resides a region of the low-dimensional space that con-

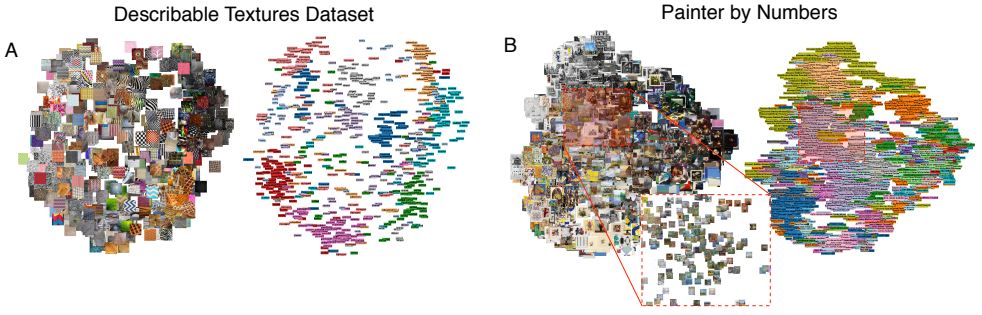

Figure 6: Structure of a low-dimensional representation of the embedding space. A: Two-dimensional representation using t-SNE for 800 textures [3] across 10 human-labeled categories. Right is the same as previous but texture replaced with a human annotated label. B: Same as previous but with Painting by Numbers dataset across for 3768 paintings across 20 labeled artists. Note the zoom-in highlighting a localized region of embedding space representing Monet paintings. Please zoom-in for details.

tains a large fraction of Impressionist paintings by Claude Monet (Figure 6B, magnified in inset). These results suggest that the style prediction network has learned a representation for artistic styles that is largely organized based on our perception of visual and semantic similarity without any explicit supervision.

3.5 The structure of the embedding space permits novel exploration.

To explore the embedding structure further, we examined whether we can generate reasonable stylizations by varying local style changes for a specific painting style. In detail, we calculate the average embedding of the paintings from a specific artist and vary the embedding vector along along the two principal components of the cluster. Figure 7 shows stylizations from these embedding variations in a 5x5 grid, together with actual paintings of the artist whose embeddings are nearby the grid. The stylizations from the grid captures two axis of style variations and correspond well to the neighboring embeddings of actual paintings. The results suggest that the model might capture a local manifold from an individual artist or painting style.

Although we trained the style prediction network on painting images, we find that embedding representation is extremely flexible. In particular, supplying the network with a content image (i.e. photograph) produces an embedding that acts as the identity tranformation. Figure 8 highlights the identity transformation on a given content image. Importantly, we can now interpolate between the identity stylization and arbitrary (in this case, unobserved) painting in order to effectively dial in the weight of the painting style.

4 Conclusions

We have presented a new method for performing fast, arbitrary artistic style transfer on images. This model is trained at a large scale and generalizes to perform stylizations based on paintings never previously observed. Importantly, we demonstrate that increasing the corpus of trained painting style confers the system the ability to generalize to unobserved painting styles. We demonstrate that the ability to generalize is largely predictable based on the proximity of the unobserved style to styles trained on by the model.

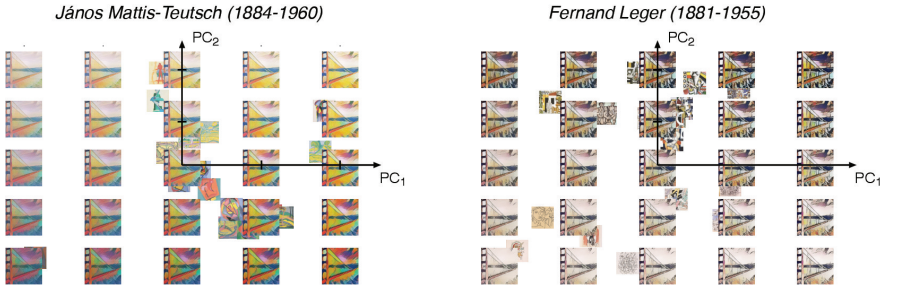

Figure 7: Exploring the artistic range of an artist using the embedding representation. Calculated two-dimensional principal components for a given artist and plotted paintings from artist in this space. The principal component space is graphically depicted by the artistic stylizations rendered on a photograph of the Golden Gate Bridge. The center rendering is the mean and each axis spans ± 4 standard deviations in along each axis. Each axis tick mark indicates 2 standard deviations. Left: Paintings and principal components of Janos Mattis-Teutsch (1884-1960). Right: Paintings and principal components of Fernand Leger (1881-1955). Please zoom in on electronic version for details.

Figure 8: Linear interpolation between identity transformation and unobserved painting. Note that the identity transformation is performed by feeding in the content image as the style image.

We find that the model architecture provides a low dimensional embedding space of normalization constants that captures many semantic properties of paintings. We explore this space by demonstrating a low dimensional space that captures the artistic range and vocabulary of a given artist. In addition, we introduce a new form of interpolation that allows a user to arbitrarily to dial in the strength of an artistic stylization.

This work offers several directions for future exploration. In particular, we observe that the embedding representation for paintings only captures a portion of the semantic information available for a painting. One might leverage metadata of paintings in order to refine the embedding representation through a secondary embedding loss [7, 20]. Another direction is to improve the visual quality of the artistic stylization through complementary methods that preserve the color of the original photograph or restrict the stylization to a spatial region of the image [10]. In addition, in a real time video, one could train the network to enforce temporal consistency between frames by appending additional loss functions [22].

Aside from providing another tool for manipulating photographs, artistic style transfer offers several applications and opportunities. Much work in robotics has focused on training models in simulated environments with the goal of applying this training in real world environments. Improved stylization techniques may provide an opportunity to improve generalization to real-world domains where data is limited [1]. Furthermore, by building models of paintings with low dimensional representation for painting style, we hope these representation might offer some insights into the complex statistical dependencies in paintings if not images in general to improve our understanding of the structure of natural image statistics.

References

[1] Konstantinos Bousmalis, George Trigeorgis, Nathan Silberman, Dilip Krishnan, and Dumitru Erhan. Domain separation networks. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 343–351. Curran Associates, Inc., 2016. URL http://papers.nips.cc/paper/6254-domain-separation-networks.pdf.

[2] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, , and A. Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[3] Vincent Dumoulin, Jonathon Shlens, and Manjunath Kudlur. A learned representation for artistic style. *International Conference of Learned Representations (ICLR)*, 2016.

[4] Alexei A Efros and William T Freeman. Image quilting for texture synthesis and transfer. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 341–346. ACM, 2001.

[5] Alexei A Efros and Thomas K Leung. Texture synthesis by non-parametric sampling. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 2, pages 1033–1038. IEEE, 1999.

[6] Jeremy Freeman and Eero P Simoncelli. Metamers of the ventral stream. *Nature neuroscience*, 14(9):1195–1201, 2011.

[7] Andrea Frome, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Tomas Mikolov, et al. Devise: A deep visual-semantic embedding model. In *Advances in neural information processing systems*, pages 2121–2129, 2013.

[8] Leon Gatys, Alexander S Ecker, and Matthias Bethge. Texture synthesis using convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 262–270, 2015.

[9] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576*, 2015.

[10] Leon A. Gatys, Alexander S. Ecker, Matthias Bethge, Aaron Hertzmann, and Eli Shechtman. Controlling perceptual factors in neural style transfer. *CoRR*, abs/1611.07865, 2016. URL http://arxiv.org/abs/1611.07865.

[11] Aaron Hertzmann, Charles E Jacobs, Nuria Oliver, Brian Curless, and David H Salesin. Image analogies. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 327–340. ACM, 2001.

[12] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. *arXiv preprint arXiv:1703.06868*, 2017.

[13] Clifford Irving. *Fake: the story of Elmyr de Hory: the greatest art forger of our time.* McGraw-Hill, 1969.

[14] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. *arXiv preprint arXiv:1603.08155*, 2016.

[15] Bela Julesz. Visual pattern discrimination. *IRE Trans. Info Theory*, 8:84–92, 1962.

[16] Chuan Li and Michael Wand. Precomputed real-time texture synthesis with markovian generative adversarial networks. *ECCV*, 2016.

[17] Lin Liang, Ce Liu, Ying-Qing Xu, Baining Guo, and Heung-Yeung Shum. Real-time texture synthesis by patch-based sampling. *ACM Transactions on Graphics (ToG)*, 20 (3):127–150, 2001.

[18] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008.

[19] Alexander Mordvintsev, Christopher Olah, and Mike Tyka. Inceptionism: Going deeper into neural networks, June 2015. URL http://googleresearch.blogspot.com/2015/06/inceptionism-going-deeper-into-neural.html.

[20] Mohammad Norouzi, Tomas Mikolov, Samy Bengio, Yoram Singer, Jonathon Shlens, Andrea Frome, Greg S Corrado, and Jeffrey Dean. Zero-shot learning by convex combination of semantic embeddings. *arXiv preprint arXiv:1312.5650*, 2013.

[21] Javier Portilla and Eero Simoncelli. A parametric texture model based on joint statistics of complex wavelet coefficients. *International Journal of Computer Vision*, 40:49–71, 1999.

[22] Manuel Ruder, Alexey Dosovitskiy, and Thomas Brox. Artistic style transfer for videos. In *German Conference on Pattern Recognition*, pages 26–36. Springer, 2016.

[23] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[24] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. *IEEE Computer Vision and Pattern Recognition (CVPR)*, 2015.

[25] Dmitry Ulyanov, Vadim Lebedev, Andrea Vedaldi, and Victor Lempitsky. Texture networks: Feed-forward synthesis of textures and stylized images. *arXiv preprint arXiv:1603.03417*, 2016.

[26] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016.

[27] Li-Yi Wei and Marc Levoy. Fast texture synthesis using tree-structured vector quantization. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 479–488. ACM Press/Addison-Wesley Publishing Co., 2000.

[28] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision*, pages 818–833. Springer, 2014.