Artistic Style Transfer Using Convolutional Neural Networks

ARMAN MANN 13BCE0073
PRATEEK BHATNAGAR 13BCE0409

1. Problem Statement

To utilise a method to transfer the style of one image to the subject of another image. Identify and preserve the content of the subject image, identify and merge the style of the artist image with the subject image. The resultant image preserves the content of the subject image and has an artistic style that is inherited and merged from the artistic image.

2. Introduction

The class of Deep Neural Networks that are most powerful in image processing tasks are called Convolutional Neural Networks. Convolutional Neural Networks consist of layers of small computational units that process visual information hierarchically in a feed-forward manner.

In machine learning, a convolutional neural network (CNN, or ConvNet) is a type of feed-forward artificial neural network in which the connectivity pattern between its neurons is inspired by the organization of the animal visual cortex. Individual cortical neurons respond to stimuli in a restricted region of space known as the receptive field. The receptive fields of different neurons partially overlap such that they tile the visual field. The response of an individual neuron to stimuli within its receptive field can be approximated mathematically by a convolution operation. [1]

The convolutional neural network is also known as shift invariant or space invariant artificial neural network (SIANN), which is named based on its shared weights architecture and translation invariance characteristics. Convolutional neural networks model animal visual perception, and can be applied to visual recognition tasks. The representations of content and style in the Convolutional Neural Network are separable. That is, we can manipulate both representations independently to produce new, perceptually meaningful images. To demonstrate this finding, we generate images that mix the content and style representation from two different source images. In particular, we match the content representation of a photograph depicting the “Neckarfront” in Tubingen, Germany and the style representations of several well-known artworks taken from ¨ different periods of art (Fig 2).

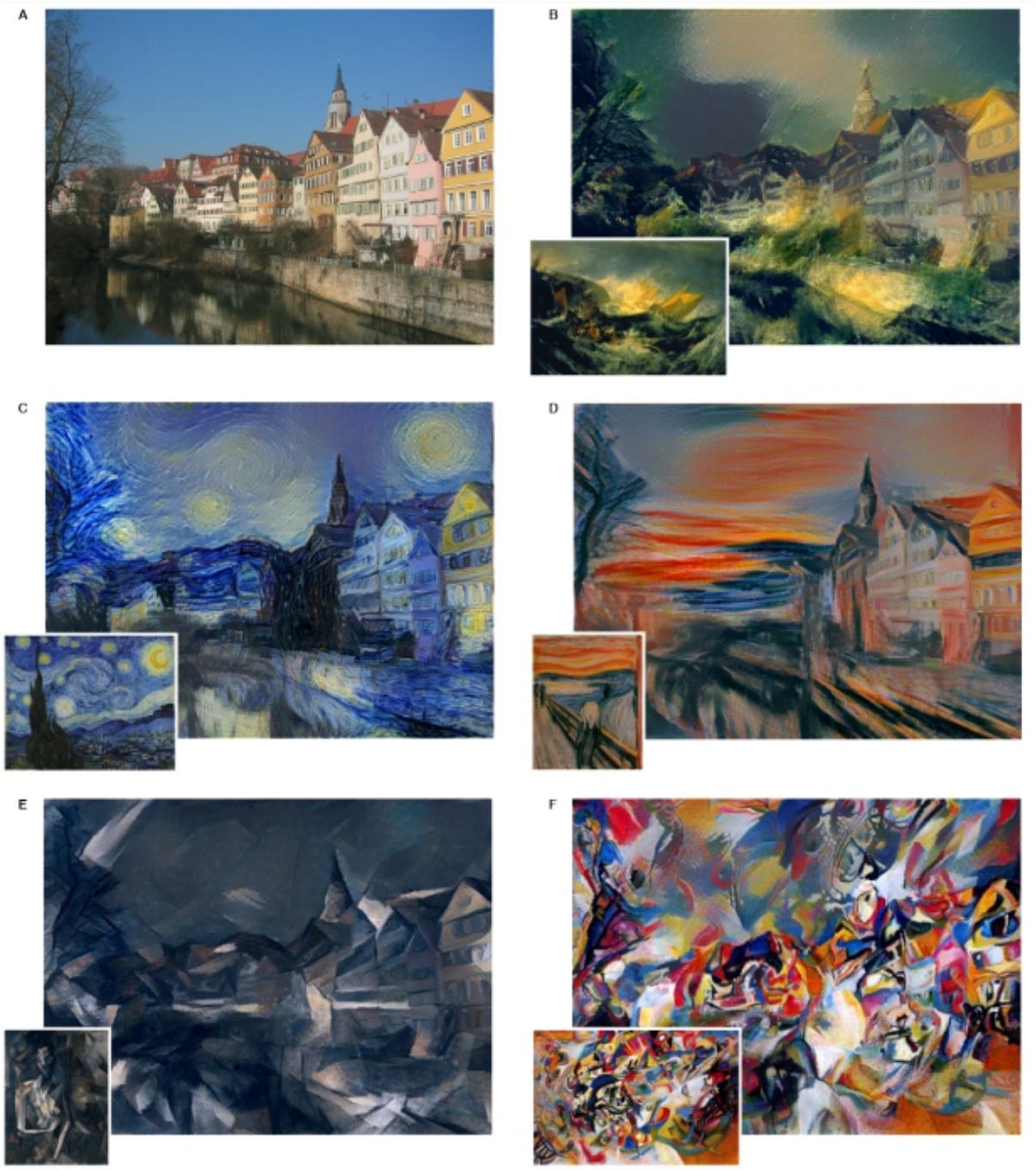

Figure 2: Images that combine the content of a photograph with the style of several well-known artworks. The images were created by finding an image that simultaneously matches the content representation of the photograph and the style representation of the artwork (see Methods). The original photograph depicting the “Neckarfront” in Tubingen, Germany, is shown in ¨ A (Photo: Andreas Praefcke). The painting that provided the style for the respective generated image is shown in the bottom left corner of each panel. B The Shipwreck of the Minotaur by J.M.W. Turner, 1805. C The Starry Night by Vincent van Gogh, 1889. D Der Schrei by Edvard Munch, 1893. E Femme nue assise by Pablo Picasso, 1910. F Composition VII by Wassily Kandinsky, 1913

3. Literature Review

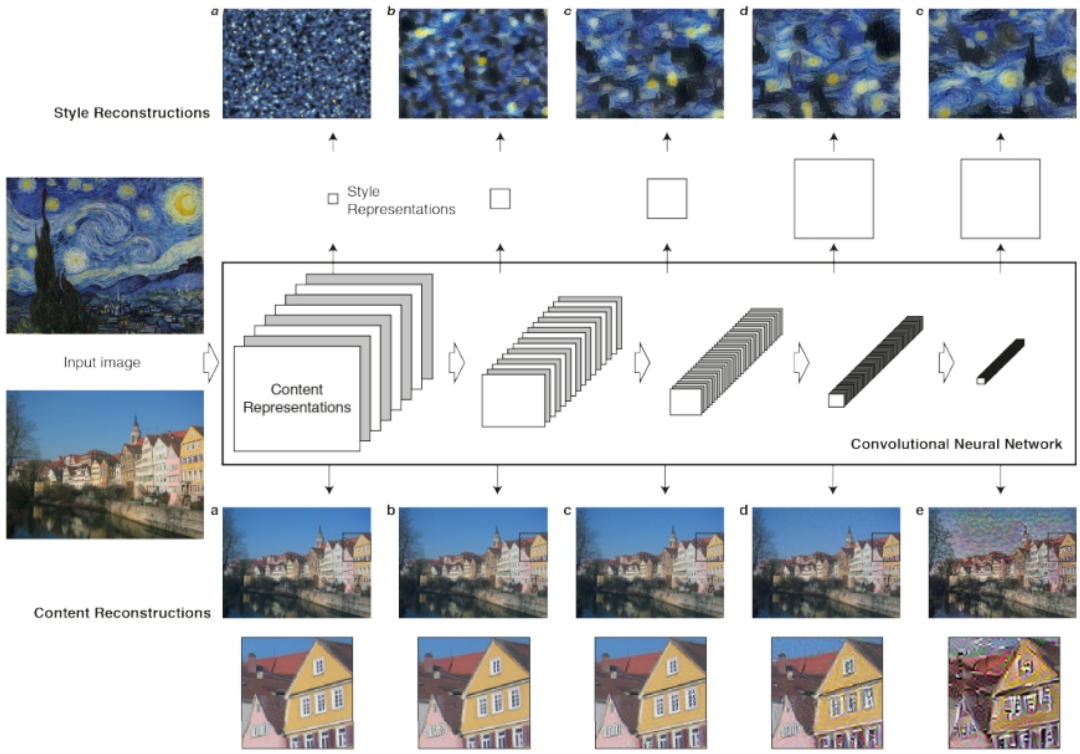

Figure 1: Convolutional Neural Network (CNN). A given input image is represented as a set of filtered images at each processing stage in the CNN. While the number of different filters increases along the processing hierarchy, the size of the filtered images is reduced by some downsampling mechanism (e.g. max-pooling) leading to a decrease in the total number of units per layer of the network. Content Reconstructions. We can visualise the information at different processing stages in the CNN by reconstructing the input image from only knowing the network's responses in a particular layer. We reconstruct the input image from from layers 'conv1 1' (a), 'conv2 1' (b), 'conv3 1' (c), 'conv4 1' (d) and 'conv5 1' (e) of the original VGG-Network. We find that reconstruction from lower layers is almost perfect (a,b,c). In higher layers of the network, detailed pixel information is lost while the high-level content of the image is preserved (d,e). Style Reconstructions. On top of the original CNN representations we built a new feature space that captures the style of an input image. The style representation computes correlations between the different features in different layers of the CNN. We reconstruct the style of the input image from style representations built on different subsets of CNN layers ('conv1 1' (a), 'conv1 1' and 'conv2 1' (b), 'conv1 1', 'conv2 1' and 'conv3 1' (c), 'conv1 1', 'conv2 1', 'conv3 1' and 'conv4 1' (d), 'conv1 1', 'conv2 1', 'conv3 1', 'conv4 1'and 'conv5 1' (e)). This creates images that match the style of a given image on an increasing scale while discarding information of the global arrangement of the scene.

Again, we can visualise the information captured by these style feature spaces built on different layers of the network by constructing an image that matches the style representation of a given input image (Fig 1, style reconstructions).10, 11 Indeed reconstructions from the

style features produce texturised versions of the input image that capture its general appearance in terms of colour and localised structures. Moreover, the size and complexity of local image structures from the input image increases along the hierarchy, a result that can be explained by the increasing receptive field sizes and feature complexity. We refer to this multi-scale representation as style representation

Of course, image content and style cannot be completely disentangled. When synthesising an image that combines the content of one image with the style of another, there usually does not exist an image that perfectly matches both constraints at the same time. However, the loss function we minimise during image synthesis contains two terms for content and style respectively, that are well separated (see Methods). We can therefore smoothly regulate the emphasis on either reconstructing the content or the style (Fig 3, along the columns). A strong emphasis on style will result in images that match the appearance of the artwork, effectively giving a texturised version of it, but hardly show any of the photograph's content (Fig 3, first column). When placing strong emphasis on content, one can clearly identify the photograph, but the style of the painting is not as well-matched (Fig 3, last column). For a specific pair of source images one can adjust the trade-off between content and style to create visually appealing images.

Figure 3: Detailed results for the style of the painting Composition VII by Wassily Kandinsky. The rows show the result of matching the style representation of increasing subsets of the CNN layers (see Methods). We find that the local image structures captured by the style representation increase in size and complexity when including style features from higher layers of the network. This can be explained by the increasing receptive field sizes and feature complexity along the network's processing hierarchy. The columns show different relative weightings between the content and style reconstruction. The number above each column indicates the ratio α/β between the emphasis on matching the content of the photograph and the style of the artwork (see Methods).

4. Architecture / Framework

The results presented above were generated on the basis of the VGG-Network Convolutional Neural Network that rivals human performance on a common visual object recognition benchmark task and was introduced and extensively described in [22] We used the feature space provided by the 16 convolutional and 5 pooling layers of the 19 layer VGGNetwork. We do not use any of the fully connected layers.The model is publicly available and can be explored in the caffe-framework. [24] For image synthesis we found that replacing the max-pooling operation by average pooling improves the gradient flow and one obtains slightly more appealing results, which is why the images shown were generated with average pooling. Generally each layer in the network defines a non-linear filter bank whose complexity increases with the position of the layer in the network. Hence a given input image ~x is

encoded in each layer of the CNN by the filter responses to that image. A layer with Nl distinct filters has Nl feature maps each of size Ml , where Ml is the height times the width of the feature map. So the responses in a layer l can be stored in a matrix Fl ∈RNl×Ml where Flij is the activation of the i th filter at position j in layer l. To visualise the image information that is encoded at different layers of the hierarchy (Fig 1, content reconstructions) we perform gradient descent on a white noise image to find another image that matches the feature responses of the original image. So let ~p and ~x be the original image and the image that is generated and Pl and Fl their respective feature representation in layer l. We then define the squared-error loss between the two feature representations.

Lcontent(~p, ~x, l) = 1 2 X i,j F l ij − P l ij2 . (1)

The derivative of this loss with respect to the activations in layer l equals

$$\frac{\partial \mathcal{L}_{content}}{\partial F_{ij}^{l}} = \begin{cases} \left(F^{l} - P^{l}\right)_{ij} & \text{if } F_{ij}^{l} > 0 \\ 0 & \text{if } F_{ij}^{l} < 0 \,. \end{cases} \tag{2}$$

from which the gradient with respect to the image ~x can be computed using standard error back-propagation. Thus we can change the initially random image ~x until it generates the same response in a certain layer of the CNN as the original image ~p. The five content reconstructions in Fig 1 are from layers ‘conv1 1’ (a), ‘conv2 1’ (b), ‘conv3 1’ (c), ‘conv4 1’ (d) and ‘conv5 1’ (e) of the original VGG-Network. On top of the CNN responses in each layer of the network we built a style representation that computes the correlations between the different filter responses, where the expectation is taken over the spatial extend of the input image. These feature correlations are given by the Gram matrix Gl ∈RNl×Nl , where Gl ij is the inner product between the vectorised feature map 10 i and j in layer l:

$$G_{ij}^{l} = \sum_{k} F_{ik}^{l} F_{jk}^{l}. \tag{3}$$

To generate a texture that matches the style of a given image (Fig 1, style reconstructions), we use gradient descent from a white noise image to find another image that matches the style representation of the original image. This is done by minimising the mean-squared distance between the entries of the Gram matrix from the original image and the Gram matrix of the image to be generated. So let ~a and ~x be the original image and the image that is generated and Al and Gl their respective style representations in layer l. The contribution of that layer to the total loss is then

$$E_l = \frac{1}{4N_l^2 M_l^2} \sum_{i,j} \left(G_{ij}^l - A_{ij}^l\right)^2 \tag{4}$$

and the total loss is

$$\mathcal{L}_{style}(\vec{a}, \vec{x}) = \sum_{l=0}^{L} w_l E_l \tag{5}$$

where wl are weighting factors of the contribution of each layer to the total loss (see below for specific values of wl in our results). The derivative of El with respect to the activations in layer l can be computed analytically:

$$\frac{\partial E_l}{\partial F_{ij}^l} = \begin{cases} \frac{1}{N_l^2 M_l^2} \left((F^l)^{\mathrm{T}} \left(G^l - A^l\right)\right)_{ji} & \text{if } F_{ij}^l > 0 \\ 0 & \text{if } F_{ij}^l < 0\,. \end{cases} \tag{6}$$

The gradients of El with respect to the activations in lower layers of the network can be readily
computed using standard error back-propagation. The five style reconstructions in Fig 1 were generated by matching the style representations on layer ‘conv1 1’ (a), ‘conv1 1’ and ‘conv2 1’ (b), ‘conv1 1’, ‘conv2 1’ and ‘conv3 1’ (c), ‘conv1 1’, ‘conv2 1’, ‘conv3 1’ and ‘conv4 1’ (d), ‘conv1 1’, ‘conv2 1’, ‘conv3 1’, ‘conv4 1’ and ‘conv5 1’ (e).
To generate the images that mix the content of a photograph with the style of a painting (Fig 2) we jointly minimise the distance of a white noise image from the content representation of the photograph in one layer of the network and the style representation of the painting in a number of layers of the CNN. So let ~p be the photograph and ~a be the artwork.

The loss function we minimise is:

$$\mathcal{L}_{total}(\vec{p}, \vec{a}, \vec{x}) = \alpha \mathcal{L}_{content}(\vec{p}, \vec{x}) + \beta \mathcal{L}_{style}(\vec{a}, \vec{x}) \tag{7}$$

where α and β are the weighting factors for content and style reconstruction respectively. For the images shown in Fig 2 we matched the content representation on layer ‘conv4 2’ and the style representations on layers ‘conv1 1’, ‘conv2 1’, ‘conv3 1’, ‘conv4 1’ and ‘conv5 1’ (wl = 1/5 in those layers, wl = 0 in all other layers) . The ratio α/β was either 1×10−3 (Fig 2 B,C,D) or 1 × 10−4 (Fig 2 E,F). Fig 3 shows results for different relative weightings of the content and style reconstruction loss (along the columns) and for matching the style representations only on layer ‘conv1 1’ (A), ‘conv1 1’ and ‘conv2 1’ (B), ‘conv1 1’, ‘conv2 1’ and ‘conv3 1’ (C), ‘conv1 1’, ‘conv2 1’, ‘conv3 1’ and ‘conv4 1’ (D), ‘conv1 1’, ‘conv2 1’, ‘conv3 1’, ‘conv4 1’ and ‘conv5 1’ (E). The factor wl was always equal to one divided by the number of active layers with a non-zero loss-weight.

5. Methodology

Our algorithm takes two images: an *input* image which is usually an ordinary photograph and a stylized and retouched reference image, the *reference style image*. We seek to transfer the style of the reference to the input while keeping the result photorealistic. We use a photorealism regularization term in the objective function during the optimization, constraining the reconstructed image to be represented by locally affine color transformations of the input to prevent distortions.

We now describe how we regularize this optimization scheme to preserve the structure of the input image and produce photorealistic outputs. Our strategy is to express this constraint not on the output image directly but on the transformation that is applied to the input image. Characterizing the space of photorealistic images is an unsolved problem. Our insight is that we do not need to solve it if we exploit the fact that the input is already photorealistic. Our strategy is to ensure that we do not lose this property during the transfer by adding a term to a function that penalizes image distortions. Our solution is to seek an image transform that is locally affine in color space, that is, a function such that for each output patch, there is an affine function that maps the input RGB values onto their output counterparts. Each patch can have a different affine function, which allows for spatial variations. To gain some intuition, one can consider an edge patch. The set of affine combinations of the RGB channels spans a broad set of variations but the edge itself cannot move because it is located at the same place in all channels. A limitation of the style term is that the Gram matrix is computed over the entire image. Since a Gram matrix determines its constituent vectors up to an isometry, it implicitly encodes the exact distribution of neural responses, which limits its ability to adapt to variations of semantic context and can cause “spillovers”. We address this problem with a semantic segmentation method to generate image segmentation masks for the input and reference images for a set of common labels (sky, buildings, water, etc.). We add the masks to the input image as additional channels and augment the neural style algorithm by concatenating the segmentation channels and updating the style loss.

To avoid “orphan semantic labels” that are only present in the input image, we constrain the input semantic labels to be chosen among the labels of the reference style image. While this may cause erroneous labels from a semantic standpoint, the selected labels are in general equivalent in our context, e.g., “lake” and “sea”. We have also observed that the segmentation does not need to be pixel accurate since eventually the output is constrained by our regularization. Thus, we formulate the photorealistic style transfer objective by combining all components together.

6. Result Discussion

In fine art, especially painting, humans have mastered the skill to create unique visual experiences through composing a complex interplay between the content and style of an image. Thus far the algorithmic basis of this process is unknown and there exists no artificial system with similar capabilities comparable in results. However, in other key areas of visual perception such as object and face recognition near-human performance was recently demonstrated by a class of biologically inspired vision models called Deep Neural Networks. Here we introduced an artificial system based on a Deep Neural Network that creates artistic images of high perceptual quality. The system uses neural representations to separate and recombine content and style of arbitrary images, providing a neural algorithm for the creation of artistic images. Moreover, in light of the striking similarities between performance-optimised artificial neural networks and biological vision, our work offers a path forward to an algorithmic understanding of how humans create and perceive artistic imagery.

The results of some of our implementations of the style transfer algorithm have been added to the disk and may be observed to judge the quality and potential of the system. Each iteration outputs an image and this sequence helps us see, first hand, how the style and content loss in minimized as we move forward.

7. Applications

Style transfer may be applied to images in the following cases:

- For modification of effects or beautification
- For copying physical attributes of another image
- For copying non physical characteristics
- To create artwork
- In mobile applications to offer quick video or image style customization
- To copy styles of paintings with certain characteristic artistic styles
- For generating artistic/stylistic video imagery

8. Conclusion

The code developed during the progress of the project as implemented by the main paper was tested and successfully implemented. Results obtained by the code are published in this report.

9. References

[1] "Convolutional Neural Networks (LeNet) – DeepLearning 0.1 documentation". DeepLearning 0.1. LISA Lab. Retrieved 31 August 2013.

[2] Taigman, Y., Yang, M., Ranzato, M. & Wolf, L. Deepface: Closing the gap to human-level performance in face verification. In Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, 1701–1708 (IEEE, 2014).
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6909616.

[3]. Guc¸l ¨ u, U. & Gerven, M. A. J. v. Deep Neural Networks Reveal a Gradient in the Complexity of Neural Representations across the Ventral Stream. The Journal of Neuroscience 35, 10005–10014 (2015). URL http://www.jneurosci.org/content/35/27/10005.

[4]. Yamins, D. L. K. et al. Performance-optimized hierarchical models predict neural responses in higher visual cortex. Proceedings of the National Academy of Sciences 201403112 (2014). URL http://www.pnas.org/content/early/2014/05/08/1403112111.

[5] Cadieu, C. F. et al. Deep Neural Networks Rival the Representation of Primate IT Cortex for Core Visual Object Recognition. PLoS Comput Biol 10, e1003963 (2014).
URL http://dx.doi.org/10.1371/journal.pcbi.1003963.

[6] Kummerer, M., Theis, L. & Bethge, M. Deep Gaze I: Boosting Saliency Prediction with Feature Maps Trained on ImageNet. In ICLR Workshop (2015).
URL /media/publications/1411.1045v4.pdf.

[7]. Khaligh-Razavi, S.-M. & Kriegeskorte, N. Deep Supervised, but Not Unsupervised, Models May Explain IT Cortical Representation. PLoS Comput Biol 10, e1003915 (2014).
URL http://dx.doi.org/10.1371/journal.pcbi.1003915.

[8] Gatys, L. A., Ecker, A. S. & Bethge, M. Texture synthesis and the controlled generation of natural stimuli using convolutional neural networks. arXiv:1505.07376 [cs, q-bio] (2015).
URL http://arxiv.org/abs/1505.07376. ArXiv: 1505.07376.

[9] Mahendran, A. & Vedaldi, A. Understanding Deep Image Representations by Inverting Them. arXiv:1412.0035 [cs] (2014). URL http://arxiv.org/abs/1412.0035.ArXiv: 1412.0035.

[10] Heeger, D. J. & Bergen, J. R. Pyramid-based Texture Analysis/Synthesis. In Proceedings of the 22Nd Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH ’95, 229–238 (ACM, New York, NY, USA, 1995). URL http://doi.acm.org/10.1145/218380.218446.

[11] Portilla, J. & Simoncelli, E. P. A Parametric Texture Model Based on Joint Statistics of Complex Wavelet Coefficients. International Journal of Computer Vision 40, 49–70 (2000). URL http://link.springer.com/article/10.1023/A%3A1026553619983.

[12] Tenenbaum, J. B. & Freeman, W. T. Separating style and content with bilinear models. Neural computation 12, 1247–1283 (2000).
URL http://www.mitpressjournals.org/doi/abs/10.1162/089976600300015349.

[13] Elgammal, A. & Lee, C.-S. Separating style and content on a nonlinear manifold. In Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on, vol. 1, I–478 (IEEE, 2004).
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1315070.

[14] Kyprianidis, J. E., Collomosse, J., Wang, T. & Isenberg, T. State of the ”Art”: A Taxonomy of Artistic Stylization Techniques for Images and Video. Visualization and Computer Graphics, IEEE Transactions on 19, 866–885 (2013).
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6243138.

[15] Hertzmann, A., Jacobs, C. E., Oliver, N., Curless, B. & Salesin, D. H. Image analogies. In Proceedings of the 28th annual conference on Computer graphics and interactive techniques, 327–340 (ACM, 2001). URL http://dl.acm.org/citation.cfm?id=383295.

[16] Ashikhmin, N. Fast texture transfer. IEEE Computer Graphics and Applications 23, 38–43 (2003).

[17] Efros, A. A. & Freeman, W. T. Image quilting for texture synthesis and transfer. In Proceedings of the 28th annual conference on Computer graphics and interactive techniques, 341–346 (ACM, 2001). URL http://dl.acm.org/citation.cfm?id=383296.

[18] Lee, H., Seo, S., Ryoo, S. & Yoon, K. Directional Texture Transfer. In Proceedings of the 8th International Symposium on Non-Photorealistic Animation and Rendering, NPAR ’10, 43–48 (ACM, New York, NY, USA, 2010). URL http://doi.acm.org/10.1145/1809939.1809945.

[19] Xie, X., Tian, F. & Seah, H. S. Feature Guided Texture Synthesis (FGTS) for Artistic Style Transfer. In Proceedings of the 2Nd International Conference on Digital Interactive Media in Entertainment and Arts, DIMEA ’07, 44–49 (ACM, New York, NY, USA, 2007).
URL http://doi.acm.org/10.1145/1306813.1306830.

[20] Karayev, S. et al. Recognizing image style. arXiv preprint arXiv:1311.3715 (2013).
URL http://arxiv.org/abs/1311.3715.15

[21] Adelson, E. H. & Bergen, J. R. Spatiotemporal energy models for the perception of motion.JOSA A 2, 284–299 (1985). URL http://www.opticsinfobase.org/josaa/fulltext.cfm?uri=josaa-2-2-284.

[22] Simonyan, K. & Zisserman, A. Very Deep Convolutional Networks for Large-Scale ImageRecognition. arXiv:1409.1556 [cs] (2014). URL http://arxiv.org/abs/1409.1556. ArXiv: 1409.1556.

[23] Russakovsky, O. et al. ImageNet Large Scale Visual Recognition Challenge. arXiv:1409.0575 [cs] (2014). URL http://arxiv.org/abs/1409.0575. ArXiv: 1409.0575.

[24] Jia, Y. et al. Caffe: Convolutional architecture for fast feature embedding. In Proceedings of the ACM International Conference on Multimedia, 675–678 (ACM, 2014). URL http://dl.acm.org/citation.cfm?id=2654889.