Novel View Synthesis with Style Transfer via 3D feature embeddings

Shubham Agrawal Somendra Tripathi Najim Yaqubie

Abstract

The advent of adversarial models has given rise to very powerful image synthesis. However, despite the rich depth in 2D image models, mathematical models of 3D environments have not been explored thoroughly. We propose to apply a hybrid generational model for style transfer upon an underlying 3D scene structure. We apply a similar dataset and quantitative metrics across multiple models in order for direct comparison. We find difficulties in the DeepVoxels model and promising results with Neural Meshes.

1. Introduction

Recent years have seen considerable progress in applying machine learning techniques to develop a 3D representation of a particular scene from 2D images. With the introduction of efficient 3D representations such as DeepVoxels (Sitzmann et al., 2019a), Scene Representation Networks (SRNs) (Sitzmann et al., 2019b) and Neural Meshes (Kato et al., 2018a), these neural networks are able to generate novel views of an object learned from a set of 2D images.

Which type of 3D representation is most appropriate for novel view synthesis, transformations and style transfer? Polygon mesh approaches are promising because they are scalable and have surfaces for lighting and textures. Polygon meshes are sets of vertices and surfaces that are incredibly compact, can represent 3D shapes with a small number of parameters and are simple to transform. We explore and expand style transfer upon mesh representations.

In contrast, voxel grid approaches are memory and computationally expensive limiting their scalability. However, because they are regularly sampled from 3D space, they are more expressible representations of the objects. As natural extensions of 2D pixels, voxel representations can be processed by Convolutional Neural Networks (CNNs) and therefore take advantage of extensive progress made in understanding 2D scenes.

Especially noteworthy are applications of style transfer upon voxel representations. Medical imaging uses voxels for representing rich MRI/CT scans (Litjens et al., 2017) where coloring or texturing images is useful for diagnosing conditions (Butler & Team, 2019). Industries such as Video games and marketing where model design and physics simulations are computed via voxels can significantly cut down on creative costs (Gao et al., 2018). Lastly, 3D printing such as unique prosthetic eyeballs (Robinson & Furneaux, 2019) or additive manufacturing for fast prototyping of concepts (Baumann & Roller, 2017) use voxels as an underlying representation for rich 3D models. We explore style transfer upon a voxel representation network through DeepVoxels.

2. Related Work

Our approach incorporates multiple active research areas such as novel view synthesis via 3D scene representations, style transfer through CNN's, and perceptual metrics.

2.1. Novel View Synthesis

There are many deep neural networks proposed to solve novel view synthesis that are multi-view consistent. Many techniques use various underlying structures such as voxel grids (Sitzmann et al., 2019a), continuous functions (Sitzmann et al., 2019b), or approximate gradients (Kato et al., 2018a). Approximate gradient methods using neural meshes address the large memory footprint of voxel grids at a cost of losing expressibility of certain features. We will measure the flexibility of these different approaches.

2.2. Style Transfer

Deep models for 2D image synthesis and manipulation have shown promising results in generating photo-realistic images (Karras et al., 2019). In other cases, deep neural networks separate and recombine content and style of arbitrary images to create artistic views (Gatys et al., 2016). Previous work has applied 2D style transfer to generated novel views from neural meshes to limited success (Kato et al., 2018a). Through propagating style loss into the neural mesh network, content loss increased and resulting novel views show deformations. We will follow previous methods of style transfer to synthesize novel views and quantitatively measure content loss vs style loss in resulting views.

Additionally, 2D style transfer has shown to be quite effective in being able to transfer arbitrary visual styles to novel content images without needing to train on any predefined styles (Li et al., 2017). We will explore applying style transfer without training on a limited style dataset.

2.3. Perceptual Metrics

Previous works have shown that features of deep neural networks trained for object classification are a better perceptual metric then classic metrics (Zhang et al., 2018). Classic perceptual metrics such as PSNR, SSIM, MSSIM, FSIM, and HDR-VDP (Lin & Kuo, 2011) rely on shallow, hand-crafted functions and fail to capture many nuances of perception.

For evaluation of our model, keeping the style-network, optimization algorithm and the number of optimizations steps fixed, we will compare the average style loss of synthesized renderings over multiple viewpoints. We will also compare models with respect to the number of optimization steps and time required to reach a threshold style loss.

3. Methodology

We extend neural meshes to include perceptual metrics as guide to efficacy. We first recreate the style transfer using neural meshes and apply quantitative metrics to novel views (Kato et al., 2018a). Then, we use our ShapeNet dataset to establish a baseline of comparable inputs and outputs (ShapeNet, 2019a). We then attempt to recreate the input dataset necessary for training the DeepVoxel network (Sitzmann et al., 2019a). Lastly, we attempt a number of approaches to apply style transfer upon DeepVoxels.

3.1. Data

We use four datasets. First, we use the partial dataset made available by Kato which consists of 2 3D objects, a teapot and a bunny (Kato et al., 2018b). Secondly, we use a group of 50 512x512 images for style transfer (Kato et al., 2018b). When working with DeepVoxels, we will reuse their dataset that consists of both 480 1920x1080 images captured with a DSLR and 480 synthetic 512x512 views of 3D objects (Sitzmann et al., 2018). Specifically, we focus on an armchair that has a similar ShapeNet counterpart useful for comparing both networks. All images have metadata associated with respect to perspective.

We extracted 15 objects from various categories from ShapeNetCore to use with both frameworks (ShapeNet, 2019a). ShapeNet has 3D models that can be converted into meshes as well as voxel grids. We use ShapeNet-Viewer to generate a series of 500 synthetic 512x512 2D images to be compatible with DeepVoxels (ShapeNet, 2019b).

In order to generate camera extrinsic and intrinsic metadata necessary for training, we use Colmap that learns attributes such as viewing angle and pose metadata (Schönberger et al., 2016). We will verify these poses and angles are sufficient for training via a DeepVoxels tool. By aiming to use the same dataset from ShapeNetCore in both architectures, results would be directly comparable.

4. Neural 3D Mesh Renderer

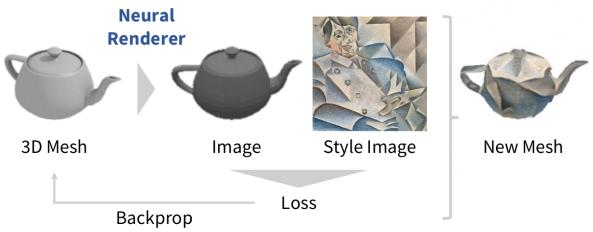

Figure 1. Overview of model components for 2D-to-3D style transfer using Neural Meshes (Kato et al., 2018a)

Using neural networks to generate novel views from meshes was not possible due to rasterization, a discrete non-differentiable operation that converts vectors into pixels. However, Neural 3D Mesh Renderer approximates rasterization gradients to develop a neural network capable of novel view synthesis (Kato et al., 2018a). Using this network allows for 3D mesh editing operations such as 2D-to-3D style transfer (see Figure 1). We use this network with trained meshes to backpropogate style loss onto the mesh and generate stylized novel views.

4.1. Style Loss Network

After training the 3D neural mesh, we then apply style transfer. First, we generate a set of novel views, compute a combined loss function using our style image as a reference, and backpropogate this loss back into the neural renderer. To ensure the underlying 3D mesh is not deformed during the style transfer, we use a weighted loss function consisting of content, style and regularization loss.

For calculating content loss, they assume that vertices-to-faces relationships $\{f_i\}$ are same for both meshes m and m_c (input mesh). Content loss is then calculated as $L_c(m|m^c) = \sum_{\{v_i, v_i^c \in (m,m^c)\}} |v_i - v_i^c|^2$. Similar to traditional image style transfer models, style loss is defined as $L_s(m|x^s, \phi) = |M(f_s(R(m,\phi))) - M(f_s(x_s))|_F^2$. Here $M(.)$ is gram matrix of feature vectors $f_s(.)$ from rendering $R(m,\phi)$ of mesh m at a particular camera position ϕ. $f_s(.)$, our style feature vector, is calculated using our style transfer network. A regularization loss of $L_t(m|\phi) = \sum_{\{\boldsymbol{p}_a, \boldsymbol{p}_b\} \in P} |\boldsymbol{p}_a - \boldsymbol{p}_b|_2^2$ is added to regularize adjacent pixel colors to be similar.

5. DeepVoxels with Style Transfer

We proposed applying a similar style loss network to a DeepVoxels approach of 3D representation (Sitzmann et al., 2019a). Given that voxels based image generation is differentiable, we do not have to approximate similar to neural meshes. DeepVoxels has demonstrated high quality view

synthesis that is encouraging towards applying style transfer with less deformation in the form of content loss.

5.1. Learning Transferred Style

Applying 2D-to-3D style transfer by generating novel views, applying style transfer, and backpropogating our loss function is not so straightforward. Following a method similar to Figure 1, we add a style transfer network that would generate a batch of randomized novel views and backpropograte a weighted loss against the style image we want to transfer. We generate random views by selecting arbitrary viewing angles of the object.

Overall loss is a combination of content loss and style loss. To measure content loss, we compare this rendered view with a non-style transferred network novel view from the same angle. Unlike Neural Meshes, we cannot directly measure the change to the underlying structure. While not a perfect content loss function when compared to the actual 3D object, this is sufficient to capture a delta to minimize loss specific to applying style transfer. To measure style loss, we reuse the function defined in section 4.1 in order to minimize differences between approaches.

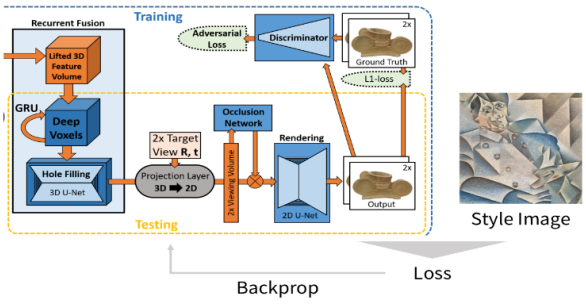

Figure 2. Overview of model components for 2D-to-3D style transfer using DeepVoxels (Sitzmann et al., 2019a)

As seen in Figure 2, there are a few more layers of complexity between the underlying 3D structure and synthesized novel views such as the occlusion network and projection layer. We explore applying style transfer to images prior to each of these steps and backpropogating our loss function.

5.2. Applying Style Transfer

We also apply style transfer to synthetic images prior to training the DeepVoxels network. Essentially, each image goes through a canonical 2D style transfer network using a white background mask. Afterwards, the underlying 3D structure has learned a stylized representation and is used to generate novel views that are multi-view consistent with respect to how style was transferred to the 2D images. We then used these results to compare to previous attempts.

In contrast to applying 2D style transfer and then running DeepVoxels, we also used a pretrained multi-layer Universal Style Network while rendering our novel views (Li et al., 2017). This uses a series of pretrained VGG-19 (Simonyan & Zisserman, 2014) convolutional neural networks and whitening and coloring transforms to apply the style features of an image onto the content of another. After rendering a novel view, we apply this network to then generate a stylized novel view as seen in Figure 3.

These stylized novel views can then be run through our combined content and style loss function and backpropogated to the DeepVoxels network. This is necessary to produce multi-view consistent style transferred novel views. However, this can also be used with translating style images based on camera angle and position.

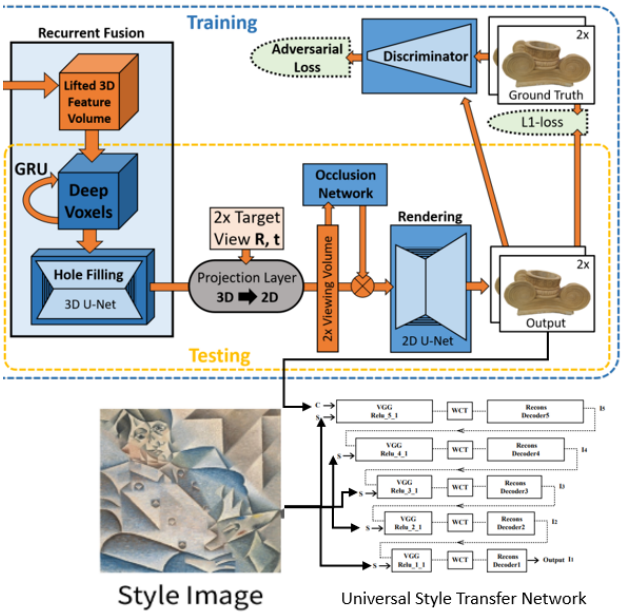

Figure 3. Applying a universal style transfer network (bottom) (Li et al., 2017) after training a DeepVoxel network (top) to generate stylized novel views. (Sitzmann et al., 2019a)

6. Results

We have run the Neural 3D Mesh Renderer with a style transfer network on the original neural meshes dataset as well as objects from various categories from ShapeNet. We have run the DeepVoxels network on their original dataset, however, synthesizing ShapeNet views necessary to train DeepVoxels is difficult. We then applied various methods of style transfer upon DeepVoxels with varied results.

Models are trained on a virtual machine hosted by Google Cloud Platform with 2 virtual CPUs, 13GB of memory, 250GB of space and a NVIDIA Tesla K80.

STYLE	OBJECT	CONTENT	STYLE	TIME
SIMPSONS	TEA POT	52.55	11.28×10^6	790
	BUNNY	42.81	70.79×10^6	1188
	URN	57.47	89.93×10^6	2498
FUNGAI	TEA POT	2.07	4.06×10^6	811
	BUNNY	1.45	3.02×10^6	1200
	URN	1.64	2.95×10^6	2480
PERRY	TEA POT	11.49	13.16×10^6	798
	BUNNY	5.29	7.78×10^6	1192
	URN	7.72	9.67×10^6	2467
NEW YORK	TEA POT	23.06	37.92×10^6	784
	BUNNY	12.35	24.91×10^6	1185
	URN	14.80	26.80×10^6	2488

Table 1. Content and Style loss for each style and object

6.1. Style Hardness

Not all style transfer is successful. We noticed, as can been seen in Table 1, that different styles have different difficulties. As style loss grows, content loss also follows given our linear loss function. We also see that time is more a factor of object than style.

Figure 4. Some styles are clearly harder than others. Here we use Simpsons (top) vs Bernardino Fungai's painting of Madonna and Child (bottom) to illustrate content loss and deformation of object during novel view synthesis.

We notice in Figure 4 that the urn is extremely deformed corresponding to a higher content loss. However, the urn is not as deformed when applying the fungai style corresponding to a lower content loss. Given this relationship, we can then grade our style dataset of varying difficulty levels for further research.

Styles that are more like textures similar to monet5 are much easier to apply when compared to styles like lichenstein1 that can be more compared to vector graphics. Figure 5 defines a ranking of difficulty to our style image dataset useful for measuring the efficacy of style transfer networks.

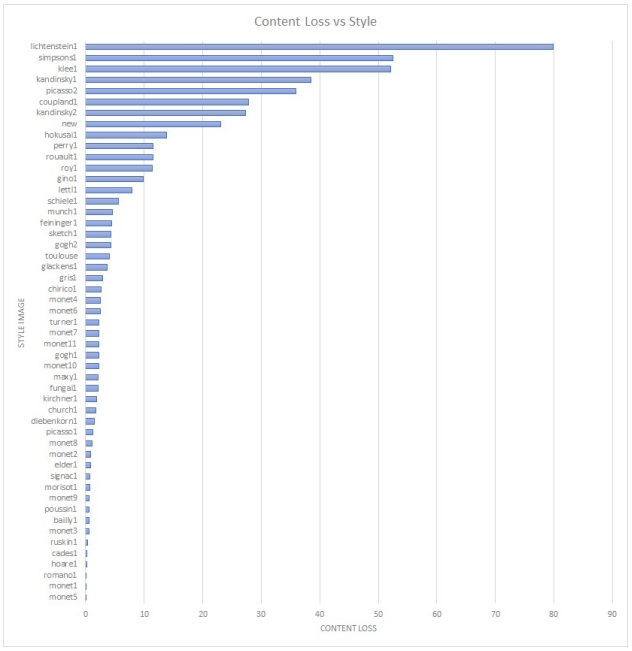

Figure 5. Hardness of style as measured by average content loss across objects. This includes all 50 styles in our dataset.

6.2. ShapeNet extension to Neural Meshes

When applying various new objects from ShapeNetCore such as the urn, relative loss was within range of previous examples (ShapeNet, 2019a) . We have successfully extended 3D style transfer using Neural Meshes to a much larger dataset than previously explored. By doing so, we can now directly compare results with other research. We see in Figure 6 that the ShapeNet urn yields comparable results to the original dataset's objects.

Figure 6. Applying style transfer to original dataset objects bunny (top) and teapot (bottom) vs ShapeNet dataset urn (middle) is comparable. Here we apply Grayson Perry's The Annunciation of the Virgin Deal to all 3 objects with similar results.

6.3. ShapeNet extension to DeepVoxels

Trying to extend DeepVoxels to new data is not as easily reproduceable. First, after selecting our 15 ShapeNet models, we synthesized over thousands screen shot images from randomized viewing angles using ShapeNet-Viewer (ShapeNet, 2019b). Afterwards, we ran Colmap to learn camera extrinsics and intrinsics such as viewing angle and distance (Schönberger et al., 2016). Then, we manually edit the poses and metadata information to conform and eliminate noise.

DeepVoxels also requires depth map images corresponding to each pose ColMap has identified that we can generate via python image libraries. At each step, there a multiple possible points of failure from Colmap not converging or identifying poses to manual errors in massaging the data. After a few iterations, we were unable to reproduce generating data that would yield results with DeepVoxels.

It is important to note that due to lack of generating new data, comparing DeepVoxels results with Neural Meshes is not direct. However, DeepVoxels codebase did have an armchair that matches our armchair we acquired from ShapeNet. To compare results, we use this armchair as a basis.

6.4. Object Hardness

Another relationship we noticed is that objects also have varying difficulty. For example, the tea pot consistently has higher content loss values regardless of style compared to other objects. Despite having higher content loss, manually checking the synthesized novel view does not show as large deformations when compared to the urn. This is likely due to the overall size of smooth surfaces on the object compared to distinct features.

We do see that objects do not all uniformly result in larger or smaller content losses. We can see in Figure 7 that there is a relationship between object and style with some styles performing better on objects. However, content losses are still within the same order of magnitude.

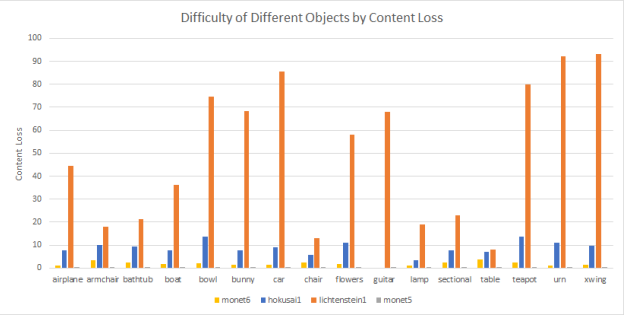

Figure 7. While content loss is within the same scale across objects, there is no clear hardest object out of our ShapeNet dataset.

6.5. Style Transfer using DeepVoxels

Our initial approach of applying a style transfer network similar to our set up with Neural Meshes did not yield good results. Although our style loss and content loss decreased as we trained, the output images did not take any style features. The relevant values can be seen in Figure 8.

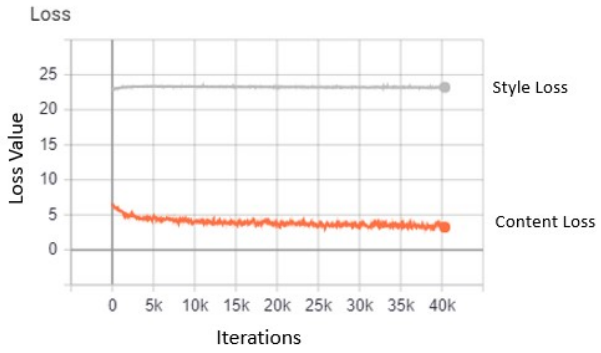

Figure 8. Overall loss reaches a steady state early, with 3.23 content loss (orange) and 23.21 style loss (grey).

Despite loss converging to a local minimum, this approach actually deformed the object as can be seen in Figure 9. This maybe due to using a simple L1 loss function to measure content loss similar to the DeepVoxels architecture and may demand a more concise approach. In contrast, Neural Meshes had the benefit of referencing distinct points representing the structural features of the object.

Figure 9. The novel view generated from backpropograting style loss to DeepVoxels from the same angle is heavily deformed for the armchair object, despite a lower content loss after training.

However, our approach using an Universal Style Transfer network when rendering novel views did show much more promising results (Li et al., 2017). One drawback is that style in subsequent novel views are not consistent as we can see in Figure 10. However, the content loss and deformation is smaller when compared to Neural Meshes. We believe that applying transformed style images based on camera angle would lead to a result that is multi-view consistent and leave that for future work.

Figure 10. Style Transfer using an universal style transfer network upon novel views generated by DeepVoxels without transforming style images yields results that are not multi-view consistent. Note the highlights in each novel view differ based on pose. Here we use Van Gogh's Starry Night (left) and Roy Lichtenstein's Bicentennial Print (right) on an armchair.

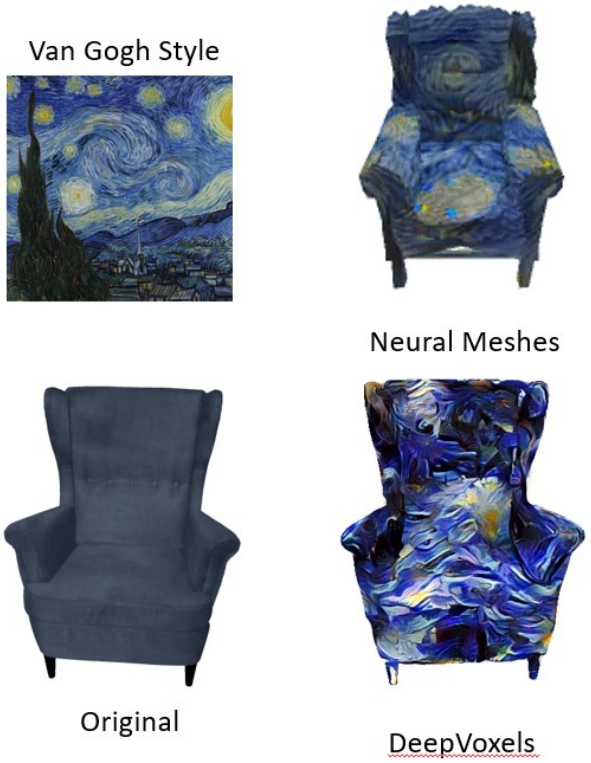

Figure 11. We can directly compare the style loss across both novel views rendered by each model. Neural Meshes resulted in 1.81e-2 style loss while DeepVoxels performed better with 7.25e-3 style loss. DeepVoxels (bottom right) outperforms Neural Meshes (top right) when applying Van Gogh's Starry Night (top left) to an armchair (bottom left).

7. Conclusions

Neural Meshes (Kato et al., 2018a) have shown promising results in terms of reusability, efficiency, and quality of results. DeepVoxels (Sitzmann et al., 2019a) have shown difficulty in terms of reusability, characteristically is not efficient, but does provide a lower content loss and therefore deformity when applying styles. DeepVoxels also results in a lower style loss when compared to Neural Meshes using the same function.

As we can see in Figure 11, both methodologies have produced stylized results while minimizing content loss. We do see deformation in both models, though, since we use different content loss functions, it is not directly comparable how well both models performed.

8. Contributions

We summarize 4 main contributions of our work. First, we defined a hardness scale for 3D Style Transfer for a dataset of 50 images. Then, we extended Neural Meshes to ShapeNet's objects for future standardized research. With that extension, we also quantified loss with 3D style transfer. Lastly, we extended our approach to DeepVoxels and demonstrated the difficulty in both applying the network to new datasets as well as learning style transfer.

All relevant code, gathered data, and results are accessible via https://github.com/najerama/3dstyletransfer.

9. Future Work

There are many directions future work can take with respect to 2D-to-3D style transfer. First would be to explore transforming style images based on camera angle to improve multi-view consistent stylized novel views in DeepVoxels.

We can also explore applying style transfer to SRNs (Sitzmann et al., 2019b) to compare with Neural Meshes (Kato et al., 2018a). Additionally, exploring alternative content loss functions may yield better results for style transfer using DeepVoxels as L1 error is prone to noise.

Lastly, exploring other voxel grid methods may prove promising given the better quantitative quality of stylized novel view synthesis with lower content and style loss.

References

Baumann, F. and Roller, D. Survey on additive manufacturing, cloud 3d printing and services. *arXiv*, 08 2017.

Butler, P. and Team, M.-C. Mars: Colour x-rays of people. 12 2019.

Gao, M., Wang, X., wu, K., Pradhana, A., Sifakis, E., Yuksel, C., and Jiang, C. Gpu optimization of material point methods. volume 37, pp. 1–12, 12 2018. doi: 10.1145/3272127.3275044.

Gatys, L. A., Ecker, A. S., and Bethge, M. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2414–2423, 2016.

Karras, T., Laine, S., and Aila, T. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4401–4410, 2019.

Kato, H., Ushiku, Y., and Harada, T. Neural 3d mesh renderer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3907–3916, 2018a.

Kato, H., Ushiku, Y., and Harada, T. Neural mesh data, 2018b. URL https://github.com/hiroharu-kato/style_transfer_3d.

Li, Y., Fang, C., Yang, J., Wang, Z., Lu, X., and Yang, M. Universal style transfer via feature transforms. *CoRR*, abs/1705.08086, 2017. URL http://arxiv.org/abs/1705.08086.

Lin, W. and Kuo, C.-C. J. Perceptual visual quality metrics: A survey. *Journal of visual communication and image representation*, 22(4):297–312, 2011.

Litjens, G. J. S., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., van der Laak, J. A. W. M., van Ginneken, B., and Sánchez, C. I. A survey on deep learning in medical image analysis. *CoRR*, abs/1702.05747, 2017. URL http://arxiv.org/abs/1702.05747.

Robinson, T. and Furneaux, W. Voxel printing using procedural art-directable technologies. In *ACM SIGGRAPH 2019 Posters*, SIGGRAPH '19, pp. 72:1–72:2, New York, NY, USA, 2019. ACM. ISBN 978-1-4503-6314-3. doi: 10.1145/3306214.3338555. URL http://doi.acm.org/10.1145/3306214.3338555.

Schönberger, J. L., Zheng, E., Pollefeys, M., and Frahm, J.-M. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision (ECCV)*, 2016.

ShapeNet. Shapenet core data, 2019a. URL https://www.shapenet.org/about#download.

ShapeNet. Shapenet model viewer, 2019b. URL https://github.com/ShapeNet/shapenet-viewer.

Simonyan, K. and Zisserman, A. Very deep convolutional networks for large-scale image recognition, 2014. URL http://arxiv.org/abs/1409.1556. cite arxiv:1409.1556.

Sitzmann, V., Thies, J., Heide, F., Nießner, M., Wetzstein, G., and Zollhofer, M. Deepvoxel dataset, 2018. URL https://vsitzmann.github.io/deepvoxels/.

Sitzmann, V., Thies, J., Heide, F., Nießner, M., Wetzstein, G., and Zollhofer, M. Deepvoxels: Learning persistent 3d feature embeddings. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2437–2446, 2019a.

Sitzmann, V., Zollhöfer, M., and Wetzstein, G. Scene representation networks: Continuous 3d-structure-aware neural scene representations. *arXiv preprint arXiv:1906.01618*, 2019b.

Zhang, R., Isola, P., Efros, A. A., Shechtman, E., and Wang, O. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 586–595, 2018.