

SP3F-GAN: SEAMLESS TEXTURE MAPS GENERATION BY ADVERSARIAL EXTENSION FOR FASHION

Honghong He^{1,2}, Zhengwentai Sun¹, Jintu Fan^{1,2} and Pik Yin Mok^{1,2}

¹*School of Fashion and Textiles, The Hong Kong Polytechnic University, Hunghom, Hong Kong*

²*Laboratory for Artificial Intelligence in Design, Hong Kong Special Administrative Region, Hong Kong*

ABSTRACT

Creating seamless textures is essential for achieving realistic virtual objects and environments. This paper proposes a method for automatic seamless texture synthesis by designing a tiled latent space in the generator of an adversarial expansion network, resulting in a continuous texture output at the seam intersection area. In addition, considering the different properties of virtual environment rendering materials, the spatially varying BRDFs (albedo, normal, roughness and displacement map) are all designed in the proposed model to handle multi-layer texture representation. Qualitative and quantitative experiments on the describable textures dataset (DTD) show that the generated texture maps are not only seamlessly tiled but also exhibit superior visual quality in preserving details compared to previous deep texture synthesis methods.

KEYWORDS

Generative Adversarial Network (GAN), Seamless Texture Synthesis, BRDFs

1. INTRODUCTION

Seamless textures are an essential component of many digital art and 3D applications, as they allow artists and designers to create realistic or stylized surfaces with ease. These textures are designed to be applied multiple times without any noticeable repetition or breaks in the pattern, resulting in a continuous and visually appealing surface. However, creating seamless textures is a challenging task, as it requires careful planning and attention to detail to ensure that the edges of the texture match up perfectly when tiled. Due to this complexity, the creation of such textures is usually done by skilled artists using specialized software to capture (Guo et al., 2020; Li et al., 2018) and reconstruct the details of real-world materials (e.g., leather, human skin), or they create their own patterns from scratch. Recent advances in convolutional neural networks (CNNs) and generative adversarial networks (GANs) have been applied to texture synthesis problems showing unprecedented levels of realism and quality (Hertz et al., 2020; Liu et al., 2020; Mardani et al., 2020). However, the output of these methods is not always seamless when it is tiled. This means that while these methods can generate visually appealing textures, they may not be suitable for use in certain applications where seamless tiling is necessary. While some recent techniques have attempted to address the problem of tileable texture synthesis (Bergmann et al., 2017; Frühstück et al., 2019), they either require a certain degree of regularity or produce textures that lack visual fidelity. Furthermore, many existing methods for synthesizing textures focus on generating a single RGB image. Accurately simulating the appearance of real-world materials in virtual environments requires a spatially varying BRDF (Montes & Ureña, 2012; Zhou et al., 2022) to represent complex materials that exhibit different properties at different locations on the surface. The BRDFs typically consists of multiple images, such as albedo, normal, roughness etc., which are used collectively to accurately represent the properties of the material. Therefore, a single RGB image is usually insufficient to capture the complexity of the material's appearance.

In this paper, we take these 4 types of texture maps into account (albedo, normal, roughness, and displacement map) and propose a **GAN-based method** for synthesizing **Seamless Patterns for 3D Fashion (SP3F-GAN)** from any arbitrary content input. We train a generative network using an adversarial extension model SP3F-GAN, which differs from the model (Zhou et al., 2018) in that SP3F-GAN involves tiling feature maps obtained from decoded input textures. The tiled feature map is then cropped in the central

region and decoded to produce a seamless output image. The discriminator is trained to distinguish between actual blocks from examples and blocks generated by the generator. The discriminator is trained to classify whether a $2k \times 2k$ texture block is real (a crop from the input exemplar) or fake (synthesized by generator). In addition, to allow texture maps synthesis, we design 4 decoders to cope with the generation of different texture maps and demonstrate the texture maps synthesis of 4 maps. Our method is shown to outperform state-of-the-art solutions for seamless texture synthesis. Unlike prior work on deep texture synthesis, our model is designed and optimized not only to generate visually pleasing texture effects but also to enable the generation of infinite texture patterns suitable for printing on various surfaces such as floors, clothing, walls, and more. This capability is demonstrated in Figure 1.

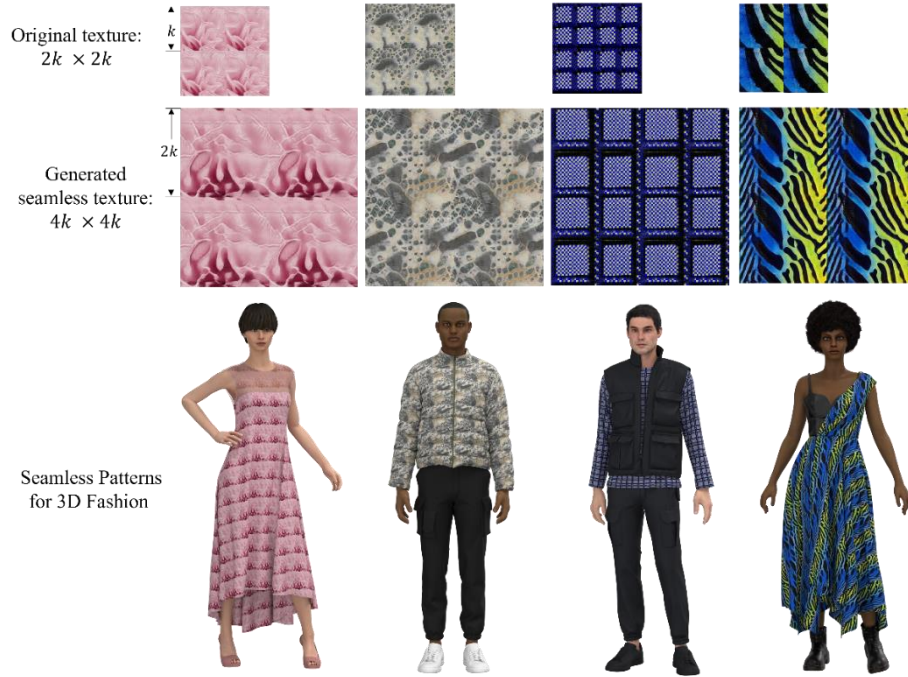


Figure 1. Tiling the original texture creates a discontinuity at the intersection of the seams, as shown in the top row. Our method can generate seamless textures, which are able to apply to fabric pattern swatches, 3D garments, etc.

2. RELATED WORK

Texture synthesis: Traditional texture synthesis algorithms are methods used to generate textures by analyzing a given input texture and synthesizing new textures with similar properties, mainly including texture mapping (Madaras & Đuriković, 2012; Piponi & Borshukov, 2000) and patch-based texture synthesis technology (Barnes et al., 2009; Cohen et al., 2003; Kaspar et al., 2015; Levin et al., 2004; Liang et al., 2001). These algorithms, while successful at generating a wide range of textures, often struggle to generate high-quality and realistic textures, especially for complex materials such as hair, fur, and natural landscapes. As a result, deep generative models have gained popularity due to its superior generation capability. Specifically, in terms of texture synthesis, they have been shown to successfully generate high-quality and diverse texture samples from images or image datasets that are visually similar to the input texture or dataset (Bergmann et al., 2017; Frühstück et al., 2019; Hertz et al., 2020; Zhou et al., 2018).

Seamless texture synthesis: Several recent works have explicitly explored the tileability of texture synthesis. The Periodic Spatial GAN proposed by Bergmann et al. (2017) can create textures from a set of random numbers that represent the expected general image, local variation, and periodic repetition of learning in generated images. GANosaic (Jetchev et al., 2017) extends such method to generate textures by optimizing the latent noise space to produce textures that match the overall content of a given guidance image. While these methods produce smoother transitions between texture patches, they still suffer from issues relating to texture

scalability and variability. Rodriguez-Pardo and Garces (2022) solves the scalability problem by completing textures into tiles through tile search and feature repetition. However, their method requires training each texture synthesis network from scratch, resulting in high computational costs. Inspired by (Zhou et al., 2018) in non-stationary texture synthesis task, in which we modified its network structure and extended it to BRDFs for generating tileable texture maps.

3. METHOD

Our ultimate goal is to generate seamless texture maps, which can then be used for downstream tasks, such as 3D garments. These seamless texture maps are perceptually similar to smaller input texture samples being tiled. However, by default, the input texture samples are tiled with significant seams. Our main idea is to eliminate the seams by automatically learning from the network to generate tileable texture maps from arbitrary texture input examples. The GAN (Goodfellow et al., 2014) is used in our method to learn the implicit representation of textures in two neural modules, generator G and discriminator D . Given a $k \times k$ source block S_A cropped from the input sample, the generator is trained to generate a new $2k \times 2k$ output $\hat{T}_A$. Among them, the other maps of the input sample are normal map, roughness map and displacement map, respectively denoted as S_N , S_R and S_D , which corresponding to the generated new texture maps represented as $\hat{T}_N$, $\hat{T}_R$ and $\hat{T}_D$. Discriminator D is trained to distinguish between these $4 \times 2k \times 2k$ texture maps are real (cropped from input samples) or fake (synthesized by G). Synthetic seamless texture maps can be applied to 3D garments for visualization and presentation. The diagram in Figure 2 depicts our proposed method of SP3F-GAN.

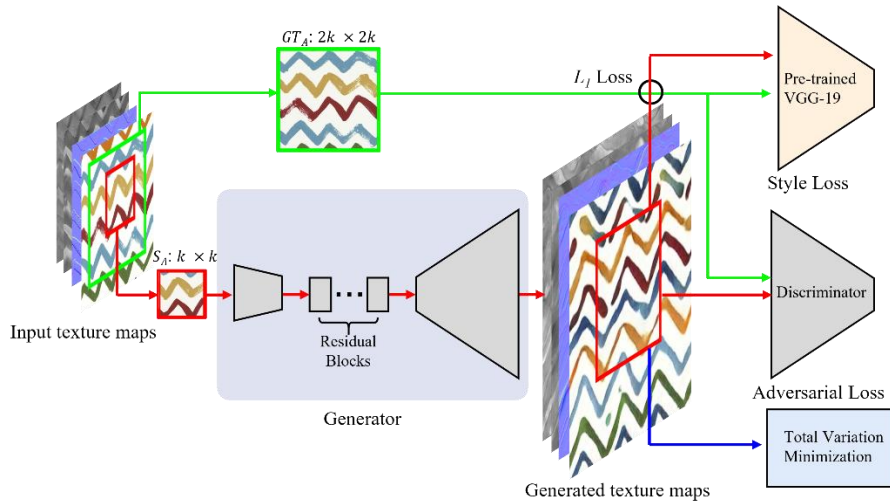


Figure 2. Illustration of the proposed method for seamless texture maps generation

Network architecture: The network architecture of SP3F-GAN is adopted from (Rodriguez-Pardo & Garces, 2022; Zhou et al., 2018), including a residual connected encoder-decoder, convolution generator G , and convolution discriminator D . However, their methods mainly focus on 2D textures. To achieve 3D seamless textures, we extend their methods with two major aspects. First, instead of synthesizing a single RGB image, we extend Generator's outputs to 4 texture maps (albedo, normal, roughness and displacement map). Second, to keep the continuity in the seam region, we applied Total Variation Loss as a regularization term.

Generator: The proposed generator consists of three main components: (i) the encoder, (ii) residual blocks and (iii) the decoder. First, the encoder encodes the input texture maps, taking the albedo map S_A as an example, to obtain feature map x_0 (the spatial resolution is $1/4$ of the input albedo map S_A). The feature map x_0 is first tiled (2×2) to obtain the tiled feature map x' (the spatial resolution is $1/2$ of the input albedo map S_A). The tiled feature map x' is then cropped in the central region to obtain x'_c . Then, a set of residual blocks learn a compact representation of the input texture. Finally, a bunch of decoders decode the output of the last residual block to the output texture $\hat{T}_A$. Synthesizing multiple texture maps with a single generative model presents additional challenges for the case of a single texture map because different texture maps produce different visual

expressions. Inspired by (Rodriguez-Pardo & Garces, 2022), we extend the decoder to four decoding texture maps for each of the four different data types.

Discriminator: Since we are not performing a pixel-by-pixel estimation of the whole image, the discriminator is designed as a PatchGAN architecture, allowing local evaluation of the synthetic texture quality for better local texture detail and global structural consistency. The discriminator has not been altered in this task, (Isola et al., 2017; Zhou et al., 2018; Zhu et al., 2017) can find more analysis on the patchGAN-based discriminator.

Loss Function: We train the network according to the standard GAN framework (Goodfellow et al., 2014), and iterate between the training D and the real texture maps GT_A , GT_N , GT_R and GT_D and the generated texture maps $\hat{T}_A$, $\hat{T}_N$, $\hat{T}_R$ and $\hat{T}_D$. In addition to the standard adversarial loss function L_{adv} , we used three additional loss terms: L_1 loss, style loss L_{style} (Gatys et al., 2015) and total variation loss L_{tv} (Larsen Greiner et al., 2022).

$$L_{tv}(x) = \sum_{i,j} \left((x_{i,j+1} - x_{ij})^2 + (x_{i+1,j} - x_{ij})^2 \right)^{\frac{1}{2}} \quad (1)$$

where x stands for the synthesized image, x_{ij} is a pixel from image x at location (i, j) . The L_1 loss is extended to 4 texture maps: L_1^a , L_1^n , L_1^r and L_1^d . The L_{style} and L_{tv} are calculated as the sum of the perceptual losses and total variation losses between the target and the generated albedo, normal, roughness and displacement texture maps. We weight the total style loss by weighting different layers in the same way described in (Gatys et al., 2016; Zhou et al., 2018). Our total loss function is defined as follow:

$$L = \lambda_{adv} L_{adv} + \lambda L_1^a + \lambda L_1^n + \lambda L_1^r + \lambda L_1^d + \lambda_{style} L_{style} + \lambda_{tv} L_{tv} \quad (2)$$

4. EXPERIMENT

In this section, we evaluate our two main contributions. First, we examine the impact of the design choices of Generator G on multilayer texture maps (Section 4.1), and second, we compare our SP3F-GAN with the SOTA method (Section 4.2).

4.1 Consistency of Texture Maps

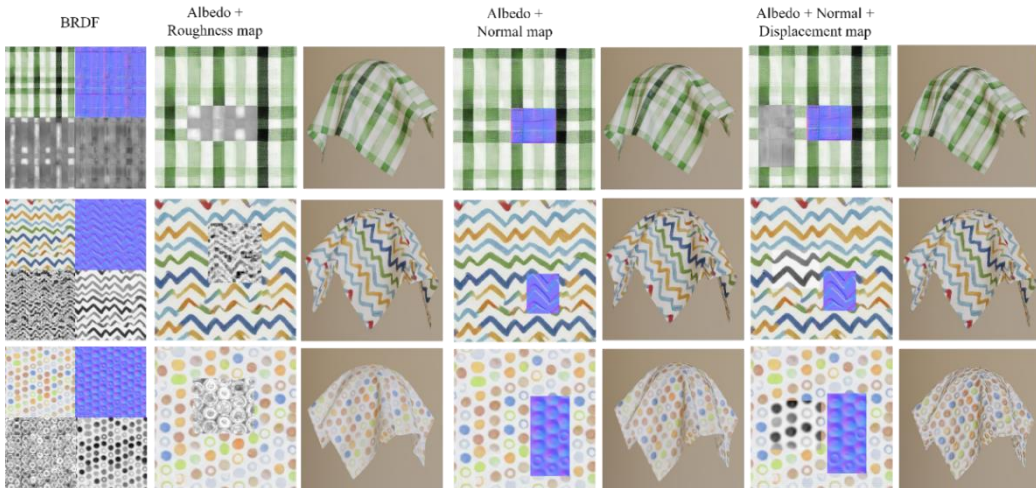


Figure 3. Illustration of consistency of synthesized texture maps

Our generator G can generate high-quality seamless texture maps but cannot guarantee that the pattern of these texture maps is consistently at a specific location. Gradients may not be shared in texture maps because L_{style} and L_{tv} are calculated independently for each map. In this section, we demonstrate the usefulness of our generator through a practical application. For the four generated seamless texture maps, we randomly select

two or three maps (e.g., albedo + roughness map, albedo + normal map, and albedo + normal + displacement map) to overlay on the fabric swatches for 3D garments and observe whether the pixel-level consistency between the maps is preserved. Figure 3 shows the consistency of synthesized texture maps.

4.2 Comparison Experiment

4.2.1 Qualitative Comparison

We compare with the SOTA texture synthesis method proposed by Bergmann et al. (2017)'s PSGAN, using samples from the describable textures dataset (DTD) (Cimpoi et al., 2014). 300 samples were randomly selected from DTD to compare with the PSGAN method. The Figure 4 shows some representative texture seamless generation results.¹ In the line, dot and marbled pattern examples, due to the relatively simple texture structure, both methods work convincingly, capturing different repeating patterns. In the second row of chequered pattern and the first row of grid pattern, PSGAN generates images with obvious artifacts, and the details of the texture are not well preserved (e.g., generated distorted chequered pattern). PSGAN performs reasonably when texture structure changes can be achieved locally at the seams but fails when the required changes are global (meaning that the effect is not good for 2*2 tiling). Textures can be applied to fabric pattern samples or 3D garments where seamless tiling is a minimum expectation. Our proposed SP3F-GAN can not only generate tractable textures but also preserve the high-level semantic structure, thereby yielding clear and realistic synthetic textures. Particularly, SP3F-GAN provides high-quality output for textures with non-uniform illumination (e.g., the second row of chequered pattern), irregular textures (e.g., striped and bubbly pattern) and gradient color textures (e.g., grid and smeared pattern).

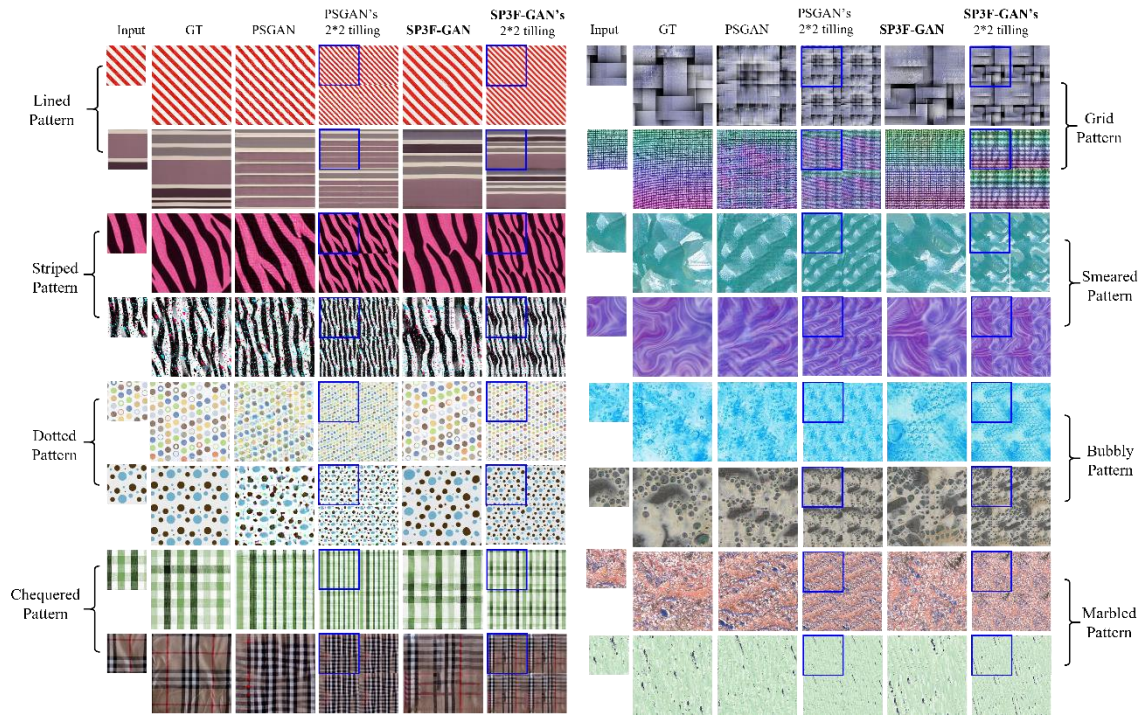


Figure 4. Qualitative comparison with SOTA method. On the left column, we show the input textures cropped from the ground truth. From left to right, the synthesized results of PSGAN (Bergmann et al., 2017) and the proposed SP3F-GAN. Outputs are tiled a similar number of times (at least twice in each dimension) for better visualization. Our method generally captures better the overall structure of the texture, while providing seamless and semantically coherent borders, for enhancing tileability.

¹More qualitative results can be obtained through e-mail: tracy.mok@polyu.edu.hk

4.2.1 Quantitative Comparison

In the quantitative evaluation, we compare the performance of the generated texture with the ground truth texture (GT) using standard metrics: SSIM and LPIPS. The Structural Similarity Index Metric (SSIM) (Wang et al., 2004) is a perceptual-aware metric that acts on pixel space to measure the similarity of structural information and may be suitable for evaluating synthesized textures. In addition, we use learning perceptual image block similarity (LPIPS) (Zhang et al., 2018) as perceptual distance metric in deep image space. This metric is widely used to evaluate generative models (Huang et al., 2018; Karras et al., 2020; Zhan et al., 2022). Experiments were conducted in a fair comparison between describable textures dataset (DTD) and the deep learning-based method PSGAN (Bergmann et al., 2017). We first compared the composite image and GT. Our method obtained better SSIM and LPIPS scores, as shown in Table 1. Then, we also compared the 2*2 tiled generated textures and GT. However, the metrics used for this experiment require both to have the same resolution. To achieve this, a 2*2 tiled generated image is cropped in the center so that it has the same resolution as the corresponding GT. Similarly, scores better than PSGAN are both in SSIM and LPIPS. This shows that our method can better preserve the stylistic and semantic content of the generated texture by combining various loss functions. In addition, our potential spatial manipulation algorithm allows seamless boundaries between tiling, superior to PSGAN in terms of structural similarity.

Table 1. Quantitative comparison of PSGAN and the proposed SP3F-GAN. Higher is better for SSIM, while lower is better for LPIPS

	PSGAN (Bergmann et al., 2017)	Center cropped PSGAN's 2*2 tiling	SP3F-GAN	Center cropped SP3F-GAN's 2*2 tiling
↑SSIM	0.1763	0.1896	0.2354	0.24566
↓LPIPS	0.5957	0.5236	0.4327	0.5063

5. CONCLUSION

In this paper, an automatic seamless texture synthesis network architecture is proposed to process multi-layer texture maps. Extensive experiments on a describable texture dataset (DTD) show that the proposed wearable GAN can generate seamless texture maps with superior visual quality in terms of detail preservation compared to previous deep texture synthesis methods.

ACKNOWLEDGEMENT

The work described in this paper was funded in part by a grant from the Laboratory for Artificial Intelligence in Design (Project Code: RP1-1) under the InnoHK Research Clusters, Hong Kong Special Administrative Region Government and by the Innovation and Technology Commission of Hong Kong, under grant ITP/028/21TP.

REFERENCES

- Barnes, C., Shechtman, E., Finkelstein, A., & Goldman, D. B. (2009). PatchMatch: A randomized correspondence algorithm for structural image editing. *ACM Trans. Graph.*, 28(3), 24.
- Bergmann, U., Jetchev, N., & Vollgraf, R. (2017). Learning texture manifolds with the periodic spatial GAN. *arXiv preprint arXiv:1705.06566*.
- Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., & Vedaldi, A. (2014). Describing textures in the wild. Proceedings of the IEEE conference on computer vision and pattern recognition.
- Cohen, M. F., Shade, J., Hiller, S., & Deussen, O. (2003). Wang tiles for image and texture generation. *ACM transactions on graphics (TOG)*, 22(3), 287-294.
- Frühstück, A., Alhashim, I., & Wonka, P. (2019). Tilegan: synthesis of large-scale non-homogeneous textures. *ACM transactions on graphics (TOG)*, 38(4), 1-11.

- Gatys, L., Ecker, A. S., & Bethge, M. (2015). Texture synthesis using convolutional neural networks. *Advances in neural information processing systems*, 28.
- Gatys, L. A., Ecker, A. S., & Bethge, M. (2016). Image style transfer using convolutional neural networks. Proceedings of the IEEE conference on computer vision and pattern recognition,
- Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative adversarial networks (2014). *arXiv preprint arXiv:1406.2661*, 1406.
- Guo, Y., Smith, C., Hašan, M., Sunkavalli, K., & Zhao, S. (2020). Materialgan: reflectance capture using a generative svbrdf model. *arXiv preprint arXiv:2010.00114*.
- Hertz, A., Hanocka, R., Giryas, R., & Cohen-Or, D. (2020). Deep geometric texture synthesis. *arXiv preprint arXiv:2007.00074*.
- Huang, X., Liu, M.-Y., Belongie, S., & Kautz, J. (2018). Multimodal unsupervised image-to-image translation. Proceedings of the European conference on computer vision (ECCV),
- Isola, P., Zhu, J.-Y., Zhou, T., & Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. Proceedings of the IEEE conference on computer vision and pattern recognition,
- Jetchev, N., Bergmann, U., & Seward, C. (2017). GANosaic: Mosaic creation with generative texture manifolds. *arXiv preprint arXiv:1712.00269*.
- Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., & Aila, T. (2020). Analyzing and improving the image quality of stylegan. Proceedings of the IEEE/CVF conference on computer vision and pattern recognition,
- Kaspar, A., Neubert, B., Lischinski, D., Pauly, M., & Kopf, J. (2015). Self tuning texture optimization. Computer Graphics Forum,
- Larsen Greiner, T. A., Lie, J. E., Kolbjørnsen, O., Kjelsrud Evensen, A., Harris Nilsen, E., Zhao, H., Demyanov, V., & Gelius, L.-J. (2022). Unsupervised deep learning with higher-order total-variation regularization for multidimensional seismic data reconstruction. *Geophysics*, 87(2), V59-V73.
- Levin, A., Zomet, A., Peleg, S., & Weiss, Y. (2004). Seamless image stitching in the gradient domain. Computer Vision-ECCV 2004: 8th European Conference on Computer Vision, Prague, Czech Republic, May 11-14, 2004. Proceedings, Part IV 8,
- Li, Z., Sunkavalli, K., & Chandraker, M. (2018). Materials for masses: SVBRDF acquisition with a single mobile phone image. Proceedings of the European conference on computer vision (ECCV),
- Liang, L., Liu, C., Xu, Y.-Q., Guo, B., & Shum, H.-Y. (2001). Real-time texture synthesis by patch-based sampling. *ACM transactions on graphics (TOG)*, 20(3), 127-150.
- Liu, G., Taori, R., Wang, T.-C., Yu, Z., Liu, S., Reda, F. A., Sapra, K., Tao, A., & Catanzaro, B. (2020). Transposer: Universal texture synthesis using feature maps as transposed convolution filter. *arXiv preprint arXiv:2007.07243*.
- Madaras, M., & Đuriković, R. (2012). Skeleton texture mapping. Proceedings of the 28th Spring Conference on computer Graphics,
- Mardani, M., Liu, G., Dundar, A., Liu, S., Tao, A., & Catanzaro, B. (2020). Neural ffts for universal texture image synthesis. *Advances in neural information processing systems*, 33, 14081-14092.
- Montes, R., & Ureña, C. (2012). An overview of BRDF models. *University of Grenada, Technical Report LSI-2012-001*.
- Piponi, D., & Borshukov, G. (2000). Seamless texture mapping of subdivision surfaces by model pelting and texture blending. Proceedings of the 27th annual conference on Computer graphics and interactive techniques,
- Rodriguez-Pardo, C., & Garces, E. (2022). SeamlessGAN: Self-Supervised Synthesis of Tileable Texture Maps. *arXiv preprint arXiv:2201.05120*.
- Wang, Z., Bovik, A. C., Sheikh, H. R., & Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4), 600-612.
- Zhan, F., Yu, Y., Wu, R., Zhang, J., Lu, S., & Zhang, C. (2022). Marginal contrastive correspondence for guided image generation. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition,
- Zhang, R., Isola, P., Efros, A. A., Shechtman, E., & Wang, O. (2018). The unreasonable effectiveness of deep features as a perceptual metric. Proceedings of the IEEE conference on computer vision and pattern recognition,
- Zhou, X., Hasan, M., Deschaintre, V., Guerrero, P., Sunkavalli, K., & Kalantari, N. K. (2022). TileGen: Tileable, Controllable Material Generation and Capture. SIGGRAPH Asia 2022 Conference Papers,
- Zhou, Y., Zhu, Z., Bai, X., Lischinski, D., Cohen-Or, D., & Huang, H. (2018). Non-stationary texture synthesis by adversarial expansion. *arXiv preprint arXiv:1805.04487*.
- Zhu, J.-Y., Park, T., Isola, P., & Efros, A. A. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. Proceedings of the IEEE international conference on computer vision