

# Texture Translation of PBR Materials Based on Pix2pix-Turbo

Jiamu Liu<sup>a</sup>

*School of Computer Science and Technology, Beijing Institute of Technology, Beijing, 100081, China*

**Keywords:** Physically Based Rendering, Texture Translation, Pix2pix-Turbo.

**Abstract:** Physically Based Rendering (PBR), a high-quality method in 3D model rendering, is widely used in modern games and 3D short films. However, generating corresponding PBR textures is relatively complex and challenging. This paper proposes a new task called PBR texture translation. The task involves generating corresponding texture maps such as height, normal, and roughness maps based on the base color image of a given PBR texture using an image-to-image translation model. Additionally, this paper improves the latest image translation model, pix2pix-turbo, by incorporating a classifier and expert models, and specifically adjusting the text-image alignment via a Text Prompt through experiments. After training on the MatSynth dataset, the model achieved a Minimum Mean Squared Error (MSE) of 1181.41 and a maximum Structural Similarity Index (SSIM) of 0.614 on the height texture of the test set, reducing MSE by 1,443.53 and improving SSIM by 0.13 compared to the original model. The contributions of this research include proposing the PBR texture translation task and improving the pix2pix-turbo model to make it more suitable for texture translation tasks.


## 1 INTRODUCTION

In the field of 3D modeling and game development, the quality and effectiveness of textures are critical factors that determine the outcome of the work. Specifically, high-quality, realistic texture maps can provide users with a more immersive, authentic, and aesthetically pleasing experience. Therefore, creating high-quality texture maps and appropriately applying them in 3D modeling software to ensure accurate and suitable shading on models is a top priority in current research in this field.

Physically Based Rendering (PBR) is an important technology for enhancing the realism and immersive experience of 3D modeling. This technology was introduced in 2004 by Matt Pharr (Pharr, Humphreys, 2004). At that time, computer rendering techniques were not very advanced, leading to "plastic-like" appearances for metallic objects in games and 3D short films. However, after over a decade of effort and collaboration from talents across various fields, modern 3D engines have seamlessly integrated PBR technology. This integration has eliminated the plastic-like appearance, bringing

objects closer to reality and sometimes achieving near-photorealism. In the field of offline rendering, the famous "Disney Principled Bidirectional Reflectance Distribution Function," introduced by Disney at SIGGRAPH 2012, significantly improved the usability of PBR. In the same year, Disney applied this technology to introduce the metallic workflow, which played a key role in the production of the critically acclaimed Wreck-It Ralph, marking a major leap in the depiction of metallic textures. In the realm of real-time rendering, various game developers shared their advancements in PBR technology at SIGGRAPH conferences. Notably, Brian Karis' talk Real Shading in Unreal Engine 4 at SIGGRAPH 2013 highlighted Unreal Engine 4 as the first game engine to use PBR technology, making it an indispensable tool in the game industry.

PBR technology requires eight different texture maps that collectively determine how the material in a 3D engine interacts with light to simulate real-world physical laws. Currently, the creation of PBR textures relies on professional artists, who go through a complex and tedious process of handling image content. The final quality of the PBR textures is

<sup>a</sup> <https://orcid.org/0009-0003-7982-380X>

highly dependent on the artist's experience and judgment. Independent game and film developers also struggle to find suitable PBR textures at the initial stages of production, which significantly increases both time and learning costs. Therefore, a key research area is how to generate high-quality PBR textures quickly and accurately, aligned with the user's expectations.

In the field of PBR texture generation, current research can generally be categorized into three main approaches. The first approach, such as the method proposed by Vecchio and Martin, focuses on automatically extracting corresponding textures from images. Vecchio and colleagues employed a diffusion model and introduced rolled diffusion and patched diffusion, achieving an SSIM of 0.729 and LPIPS of 0.184 (Vecchio, Martin, Roullier, et al., 2023; Martin, Roullier, Rouffet, et al., 2022). The second approach, as proposed by Guo and Hu, involves generating PBR textures based on various conditions and rules. Guo and colleagues built a MaterialGAN model using StyleGAN2, optimizing latent space representations to better generate target textures under constrained conditions, achieving the lowest LPIPS of 0.071, significantly outperforming previous models (Guo, Smith, Hašan, et al., 2020; Hu, Hašan, Guerrero, et al., 2022). The third approach involves more convenient methods like text-to-texture, as recently proposed by Vecchio and Siddiqui. Siddiqui and colleagues' Meta 3D AssetGen model used multi-view and symbolic distance functions to represent 3D shapes more reliably, improving Chamfer distance by 17% and LPIPS by 40% (Vecchio, 2024; Siddiqui, Monnier, Kokkinos, et al., 2024).

These methods have undoubtedly improved the convenience and speed of PBR material creation. However, since many of these models and applications rely on textual descriptions or various constraints to generate textures, the final results may not be as satisfactory to users as those created from handpicked or photographed textures. Additionally, since the results of these models are closely tied to the quality of the training dataset, previous models may struggle to generate high-quality PBR textures based on outdated training data. This paper will make corresponding adjustments and improvements to the pix2pix-turbo method, aiming to achieve PBR texture translation based on an improved pix2pix-turbo model. The goal is to enhance the quality of the generated images, enabling the rapid translation of consistent and high-quality PBR textures.

## 2 METHOD

### 2.1 MatSynth Dataset

The MatSynth dataset (Vecchio, Deschaintre, 2024) is a high-definition PBR texture dataset containing over 4,000 ultra-high-resolution textures. Curated and published by Giuseppe Vecchio and Valentin Deschaintre, the dataset focuses on a variety of materials under the CC0 and CC-BY licensing frameworks, sourced from AmbientCG, CGBookCase, PolyHeaven, ShareTexture, TextureCan, and part of artist Julio Sillet's materials released under the CC-BY license. The dataset covers 13 types of materials: ceramic, concrete, fabric, ground, leather, marble, metal, misc, plaster, plastic, stone, terracotta, and wood. Each material category contains over 200 sets of PBR textures, and each set includes Basecolor, Diffuse, Normal, Height, Roughness, Metallic, Specular, and Opacity maps.

In addition, the dataset's publishers visually inspected and filtered out low-quality and low-resolution PBR textures, and enhanced the original dataset using a method that blends semantic compatibility. The MatSynth dataset provides an important data source for the texture generation field, addressing the scarcity of high-quality datasets over the past six years, which were plagued by issues such as low resolution, copyright restrictions, and limited material variety. Figure 1 shows an example of a wood texture from the dataset, containing eight different texture maps.

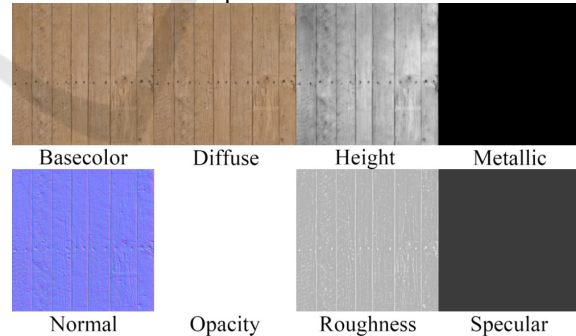


Figure 1: MatSynth dataset example (Photo/Picture credit: Original).

### 2.2 Pix2pix Principle

The pix2pix-turbo model is a successor to pix2pix (Isola, Zhu, Zhou, et al. 2017). Before the pix2pix method was introduced, image translation tasks were a significant and extensive branch of image processing. Many methods require different model

architectures and loss functions to adapt to the specific task at hand. Due to the diversity of tasks (e.g., facade reconstruction versus Monet-style translation), the resulting model architectures and loss functions varied greatly, making it difficult to standardize operations across tasks.

However, just as in the field of Natural Language Processing (NLP), where all NLP tasks can be generalized as question-answer tasks, the pix2pix method introduced a unified approach for image translation tasks. This method uses Conditional Generative Adversarial Networks (CGANs) for image translation, but unlike traditional CGANs, the discriminator in pix2pix operates on image pairs rather than single images. The generator uses a U-Net architecture to retain more details from the original image, ensuring that the generated image contains both high-level features (e.g., textures) and low-level features (edges, corners, contours, colors).

For instance, given an original image  $x$ , noise input  $z$ , and corresponding target image  $y$ , with the U-Net generator represented as  $G$  and the discriminator as  $D$ , the generated fake image would be  $G(x)$ . Instead of having the discriminator compare  $G(x)$  with  $y$ , it distinguishes between the pairs  $(x, G(x))$  and  $(x, y)$ . The discriminator does not directly assess whether the generated image is real or fake but rather determines whether the generated or target image forms a valid image pair with the original image. This strengthens the model's ability to maintain correspondence between the original and target images.

Based on this setup, the loss functions for the generator and discriminator are as follows:

$$\min_G \max_D V(D, G) = \mathbb{E}_{x,y} [\log D(x, y)] + \mathbb{E}_{x,z} [\log(1 - D(x, G(x, z)))] \quad (1)$$

Nevertheless, at this stage, the generated image and the original image still do not have a pixel-level correspondence. Since the cGAN generator inevitably requires noise data, the generated image may have slight shifts at the edges. To deceive the discriminator  $D$ , the generator may produce blurry edges to minimize the loss. However, having blurry edges is not ideal for a high-quality generated image.

To prevent the generator from producing blurry edges, an L1 loss is introduced, ensuring that the generated image closely matches the target image at the pixel level. L1 is chosen over L2 because L1 represents the median, whereas L2 represents the mean, and L2 tends to produce more blurriness compared to L1.

With the introduction of L1 loss, the final loss function is as follows:

$$\min_G \max_D L_{CGAN}(D, G) = \mathbb{E}_{x,y} [\log D(x, y)] + \mathbb{E}_{x,z} [\log(1 - D(x, G(x, z)))] \quad (2)$$

$$G^* = \min_G \max_D L_{CGAN}(D, G) + \lambda L_1(G) \quad (3)$$

## 2.3 Pix2pix-Turbo Principle

Although pix2pix can achieve good results in image translation tasks, it struggles with tasks that require precise or complex image descriptions, as it cannot effectively learn intricate patterns. Additionally, the original pix2pix requires training the generator from scratch, which incurs significant time costs, and the model's inference speed is not optimal.

To address these issues, Gaurav Parmar and colleagues (Parmar, Park, Narasimhan, et al., 2024). Introduced improvements by incorporating a Text Encoder and leveraging pre-trained diffusion models (such as SD-Turbo). Instead of training the generator from scratch, they fine-tuned it using LoRA (Low-Rank Adaptation) through text prompts and input-target image pairs. To align text with images, they used CLIP (Radford, 2021), a model for connecting natural language supervision with visual models.

Additionally, skip connections were introduced between the encoder and decoder of the generator network to balance detail loss caused by generator changes. Consequently, the model's loss function includes not only the original pix2pix generator and discriminator losses but also CLIP similarity loss and reconstruction loss (including L2 loss and LPIPS loss to measure differences between the generated and target images).

With these improvements, the pix2pix-turbo model allows fine-grained control over generated content using text prompts. It also achieves faster training and inference times compared to the original pix2pix, while producing images with superior overall quality and better detail retention.

## 2.4 Model Improvement

To apply the pix2pix-turbo method to the domain of PBR texture generation, several adjustments to the model are necessary. The original method was designed for one-to-one correspondence between an input image and a target image, whereas the task in this paper requires generating multiple texture maps—such as height, normal, and roughness—from a single base color image. This turns the task into a

one-to-many problem, necessitating a modification of the original model structure.

Additionally, since different materials exhibit distinct properties, their corresponding texture maps may vary significantly. For example, the metallic map for the ceramic category is mostly black, as ceramics do not exhibit metallic properties. Conversely, metal textures often contain large white areas, representing the presence of metallic shine. Therefore, to distinguish between different material categories and generate appropriate texture maps for each, the model needs to incorporate a classifier that can identify the input material type.

In this experiment, the yolov8m-cls model is employed for image classification, helping the system better recognize the material category of the input image.

At the same time, due to the overlapping characteristics of different types of PBR materials in the training dataset (for example, Ground textures may include small amounts of stone as embellishments), even though the input image can largely be classified into a specific category for the texture translation task, there needs to be a fallback mechanism. To ensure that the expert model assigned by the classifier can successfully process the input image, the system provides a universal expert model as a secondary option. This fallback guarantees that, in cases where the classifier makes an error, the input image won't be processed by an incorrect expert model and yield poor results. Instead, the universal expert model offers an alternative path, allowing the user to obtain a more reliable output.

Next is the Text Prompt design. Unlike traditional image translation tasks, where features are easier to describe, the specific requirements for PBR texture maps are more abstract. For example, in a standard

task, if you want to transform a daytime image into a nighttime one, the text prompt "night" suffices. Similarly, if you want a circle image to be filled with violet and have an orange background, a text prompt like "violet circle with orange background" would work. This is because CLIP, during training, has aligned abstract concepts like "night" and colors such as "violet" or "orange." However, when it comes to more technical terms like height map, it is uncertain whether CLIP can adequately align with these concepts. Experimental validation is needed to determine how well CLIP handles such specialized terms.

Additionally, it is crucial to assess the quality of the generated texture maps. This study uses MSE to evaluate the overall similarity between the generated and target images, while SSIM measures the structural similarity. A subjective evaluation of the rendered textures after model inference is also employed to assess the practical performance of the generated images.

Ultimately, the modified pix2pix-turbo architecture is shown in Figure 2.

## 2.5 Experimental Procedure

### 2.5.1 MatSynth Dataset Preprocessing

The MatSynth dataset is provided for download in Parquet format, with a total size of over 400 GB. To ensure the training process is both efficient and manageable, the preprocessing steps involved downloading the dataset and cropping the material images from 4096x4096 to 512x512. The images were then categorized by type and stored accordingly, compressing the dataset from over 400 GB to 8 GB for easier data transfer and training.

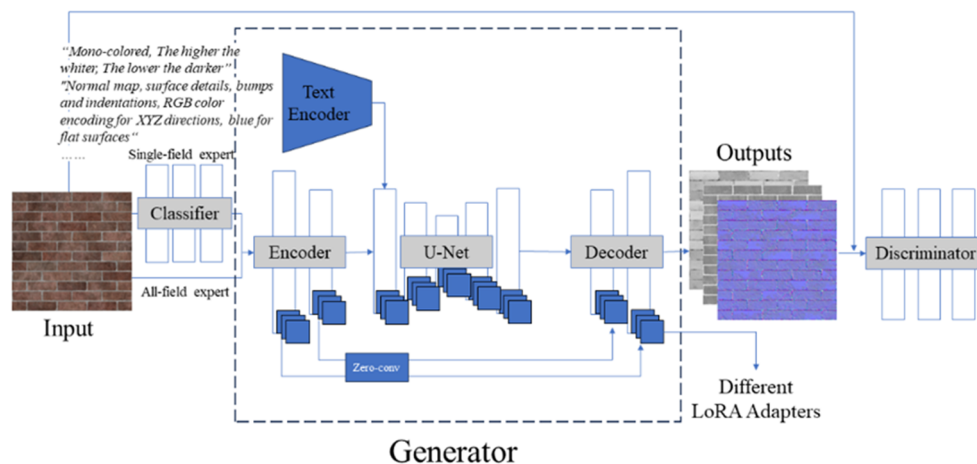


Figure 2: The architecture of the improved pix2pix-turbo model (Photo/Picture credit: Original).



Additionally, the image formats within the dataset were converted for consistency. To standardize the input and output formats, all single-channel grayscale images were converted to RGB three-channel images. Similarly, RGBA four-channel images were also converted to RGB three-channel images to maintain format uniformity, facilitating the model's processing of input and output.

### 2.5.2 Text Prompt Design

To select the best Text Prompt for fine-grained alignment of material representations, this study designed experiments focused on the Height map to investigate how different text prompts affect the generation results. For clarity, the experiment selected the most distinguishable height conditions from category 11, Terracotta. This category primarily consists of brick wall structures, making it suitable for evaluating the effectiveness of different prompts using both MSE and SSIM metrics, as well as subjective visual assessment.

The results of the experiment were obtained by training the model multiple times with different text prompts and averaging the evaluation metrics. One prompt simply required converting the base color to height, while another provided a detailed description of the height map characteristics and conversion requirements. Each set of experiments was run three times, and the average values for MSE and SSIM were compared to determine the effectiveness of the text prompts.

### 2.5.3 Model Training

First, the image classification task was trained using the yolov8m-cls model. Although yolov8n-cls is faster and yolov8x-cls is more accurate, the yolov8m-cls model was chosen for its balance between accuracy and time efficiency. The training dataset for the classification task is a subset of the training dataset for the texture conversion task, and it only includes base color images. The final training size was set to 512x512, with the number of epochs configured to 100.

There are 13 material categories in the training set, with each category providing only three types of textures for training: Height, Normal, and Roughness. Diffuse textures were excluded because they are nearly identical to the base color in most categories, leaving insufficient training samples. Specular and Opacity textures were also excluded as they generally consist of solid colors with minimal variation, making them less valuable for training. Metallic textures were only available for metal material categories, so they

were not used in training for every material type. Additionally, for categories with too few samples after splitting by type, a universal expert model was used as a substitute.

After preparing the text prompts for each material type, the model was trained using an RTX 4090 24GB GPU. Each texture was 512x512 in size, with a maximum of 10,000 training steps.

## 3 EXPERIMENTAL RESULTS

To test the conversion generation capability of the model, the improved pix2pix-turbo model was evaluated using the Mean Squared Error (MSE) and Structural Similarity Index (SSIM) metrics across 13 different material categories. MSE measures the mean squared error of the pixel differences between the target image and the converted image, while SSIM compares the images in higher-level dimensions such as brightness and contrast. Generally, for high-quality generated images, MSE should be lower and SSIM should be higher.

The experiment first compared the effects of different prompts on the model's performance. Two types of prompts were used: one that directly requested conversion and another that provided a detailed description of the conversion rules and material characteristics. For the Terracotta category (category 11), each prompt was used to train the model three times, and the final conversion results were averaged based on their performance on the test set. The results showed that the model trained with the first prompt had an average MSE of 2868.36 and an average SSIM of 0.49, while the model trained with the second prompt achieved an average MSE of 2672.22 and an average SSIM of 0.51. Both metrics were better for the second prompt, indicating that the quality of the generated images improved with more detailed and specific prompts.

This difference is evident from the height maps generated, as shown in the images below. The first prompt did not specify that the height map should be black and white, which led to a lower penalty from the CLIP model for non-black-and-white colors in the generated images. This resulted in some parts of the generated images not being black and white, directly affecting the MSE and SSIM scores. The final generated material effects are shown in Figure 3.

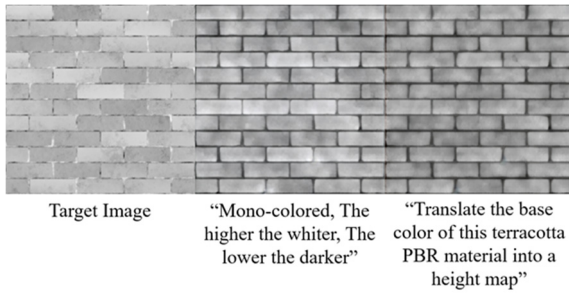


Figure 3 Comparison of Training Results with Different Text Prompts: The left image shows the target image, the middle image shows the converted image generated after a detailed description of conversion rules and characteristics, and the right image shows the converted image generated after providing a simple conversion instruction (Photo/Picture credit: Original).

In addition, the experiment was conducted on the unmodified pix2pix-turbo method to compare the image conversion results in the PBR material domain before and after the model improvement. In Figure 4, the upper image is a comparison of MSE metrics, with the x-axis representing material category numbers and the y-axis representing MSE. The blue line shows the MSE of the original model for Height, the orange line shows the MSE of the improved model for Height, and the green line shows the MSE of the improved model for Normal. Similarly, the lower image is a comparison of SSIM metrics, with the x-axis representing material category numbers and the y-axis representing SSIM. The blue line shows the SSIM of the original model for Height, the orange line shows the SSIM of the improved model

for Height, and the green line shows the SSIM of the improved model for Normal.

From Figure 4, the MSE comparison results for categories 5, 6, and 7 show that because the training set was divided by category, some categories did not have enough training data to effectively support the domain-specific expert models. As a result, the full-domain expert model was used as an alternative, which led to some material types performing on par with the original model, while others, such as categories 1, 3, and 8, achieved better results.

In some cases, the MSE for height maps is relatively poor while the SSIM is good. This is due to the fact that height information provides a relative estimate but cannot accurately determine the exact height difference, leading to larger pixel differences, while structural information can still be well transferred.

Additionally, it's worth noting that for the normal map of category 5, marble, the MSE is lower and the SSIM is exceptionally high. This is because the marble surface has fewer protrusions, which leads to better results in the comparison.

The subjective visual comparison is shown in Figure 5. Judging from the visual results, the details produced by the original pix2pix model are the poorest, with many artifacts and jagged edges, and the solid color areas are inadequately filled. The original pix2pix-turbo model, lacking expert and classification systems, failed to effectively represent the height variations, resulting in nearly grayscale outputs. In contrast, the improved pix2pix-turbo model generates clearer height maps.

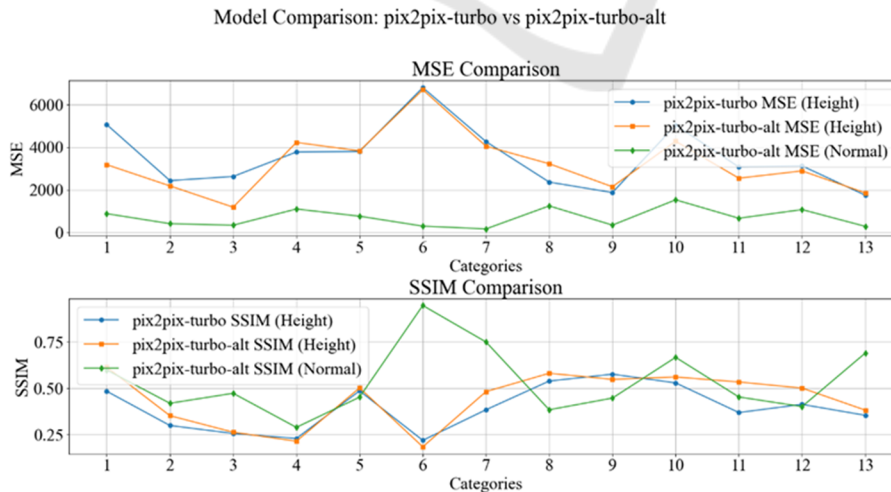


Figure 4: Comparison of MSE and SSIM Results between the Improved Model and the Original Model: The upper chart shows the MSE comparison and the lower chart shows the SSIM comparison. The blue line represents the original model, while the orange and green lines represent the improved model (Photo/Picture credit: Original).

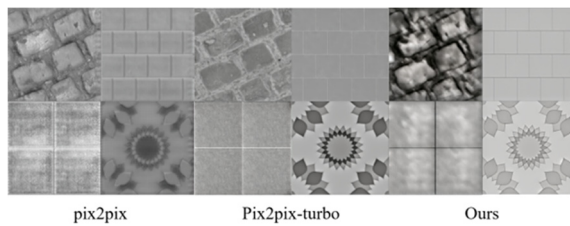


Figure 5: Comparison of height maps generated by different models. The left image shows the conversion result using the pix2pix model, the middle image shows the result from the original pix2pix-turbo model, and the right image displays the result from the improved pix2pix-turbo model (Photo/Picture credit: Original).

Additionally, the dataset includes many examples where the source and target images have weak correlations. For instance, in the Ceramic category, a significant portion of the materials are tiles. This means that even if the base color image has complex patterns, the height map might simply consist of straightforward square segments. These examples do not provide the model with meaningful variation patterns, which contributes to a decline in the final generated quality.

## 4 CONCLUSIONS

This paper improves the pix2pix-turbo model by incorporating multi-layer LoRA for material generation and introducing a classifier along with a combination of domain-specific expert models and a general expert model. The improved model achieved a minimum MSE of 1181.41 and a maximum SSIM of 0.614 on the MatSynth dataset's height maps, which represents a reduction in MSE by 1,443.53 and an increase in SSIM by 0.13 compared to the original model. The results demonstrate that the improved model has a strong capability for PBR material image conversion, allowing for the rapid generation of high-quality PBR material images from input basecolor images.

From the experimental results, it is evident that the modified pix2pix-turbo model for PBR material conversion performs better than the original model and the standard pix2pix model. Additionally, the CLIP text-image alignment tool shows that more precise input leads to better material generation results. However, CLIP may not fully understand certain terms. For example, the quality of images generated with the terms "rough" and "smooth" for roughness material does not match the quality achieved for height and normal maps.

Future research could focus on the semantic aspects of images, using other models to evaluate height, normal, and roughness features in specific areas. Combining models like pix2pix-turbo with advanced image semantic understanding could enhance the realism and accuracy of PBR material conversion effects, addressing the model's current limitations in roughness material conversion.

## REFERENCES

- Guo, Y., Smith, C., Hašan, M., Sunkavalli, K. and Zhao, S., 2020. *MaterialGAN: Reflectance capture using a generative SVBRDF model*. arXiv preprint arXiv:2010.00114.
- Hu, Y., Hašan, M., Guerrero, P., Rushmeier, H. and Deschaintre, V., 2022. *Controlling material appearance by examples*. Computer Graphics Forum, 41(4), pp.117-128.
- Isola, P., Zhu, J. Y., Zhou, T. and Efros, A. A., 2017. *Image-to-image translation with conditional adversarial networks*. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp.1125-1134.
- Martin, R., Roullier, A., Rouffet, R., Kaiser, A. and Boubekur, T., 2022. *MaterIA: Single image high-resolution material capture in the wild*. Computer Graphics Forum, 41(2), pp.163-177.
- Parmar, G., Park, T., Narasimhan, S. and Zhu, J. Y., 2024. *One-step image translation with text-to-image models*. arXiv preprint arXiv:2403.12036.
- Pharr, M. and Humphreys, G., 2004. Physically based rendering: From theory to implementation.
- Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S. and Sutskever, I., 2021. *Learning transferable visual models from natural language supervision*. In International Conference on Machine Learning, pp.8748-8763. PMLR.
- Siddiqui, Y., Monnier, T., Kokkinos, F., Kariya, M., Kleiman, Y., Garreau, E. and Novotny, D., 2024. *Meta 3D AssetGen: Text-to-mesh generation with high-quality geometry, texture, and PBR materials*. arXiv preprint arXiv:2407.02445.
- Vecchio, G., 2024. *StableMaterials: Enhancing diversity in material generation via semi-supervised learning*. arXiv preprint arXiv:2406.09293.
- Vecchio, G. and Deschaintre, V., 2024. *MatSynth: A modern PBR materials dataset*. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.22109-22118.
- Vecchio, G., Martin, R., Roullier, A., Kaiser, A., Rouffet, R., Deschaintre, V. and Boubekur, T., 2023. *Controlmat: A controlled generative approach to material capture*. ACM Transactions on Graphics.