Delft University of Technology

DeepMaterialInsights

A Web-based Framework Harnessing Deep Learning for Estimation, Visualization, and Export of Material Assets from Images

Sinha, Saptarshi Neil; Gorsclüter, Felix; Graf, Holger; Weinmann, Michael

DOI
10.1145/3665318.3677152

Publication date
2024

Document Version
Final published version

Published in
Proceedings - Web3D 2024

Citation (APA)
Sinha, S. N., Gorsclüter, F., Graf, H., & Weinmann, M. (2024). DeepMaterialInsights: A Web-based Framework Harnessing Deep Learning for Estimation, Visualization, and Export of Material Assets from Images. In S. N. Spencer (Ed.), *Proceedings - Web3D 2024: 29th International ACM Conference on 3D Web Technology* Article 24 (Proceedings - Web3D 2024: 29th International ACM Conference on 3D Web Technology). ACM. https://doi.org/10.1145/3665318.3677152

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright
Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy
Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

This work is downloaded from Delft University of Technology.
For technical reasons the number of authors shown on this cover page is limited to a maximum of 10.

DeepMaterialInsights: A Web-based Framework Harnessing Deep Learning for Estimation, Visualization, and Export of Material Assets from Images

Saptarshi Neil Sinha
Fraunhofer IGD
Darmstadt, Hessen, Germany
saptarshi.neil.sinha@igd.fraunhofer.de

Felix Gorsclüter
Fraunhofer IGD
Darmstadt, Hessen, Germany
felix.gorschlueter@igd.fraunhofer.de

Holger Graf
Fraunhofer IGD
Darmstadt, Hessen, Germany
holger.graf@igd.fraunhofer.de

Michael Weinmann
Delft University of Technology
Delft, Zuid-Holland, Netherlands
M.Weinmann@tudelft.nl

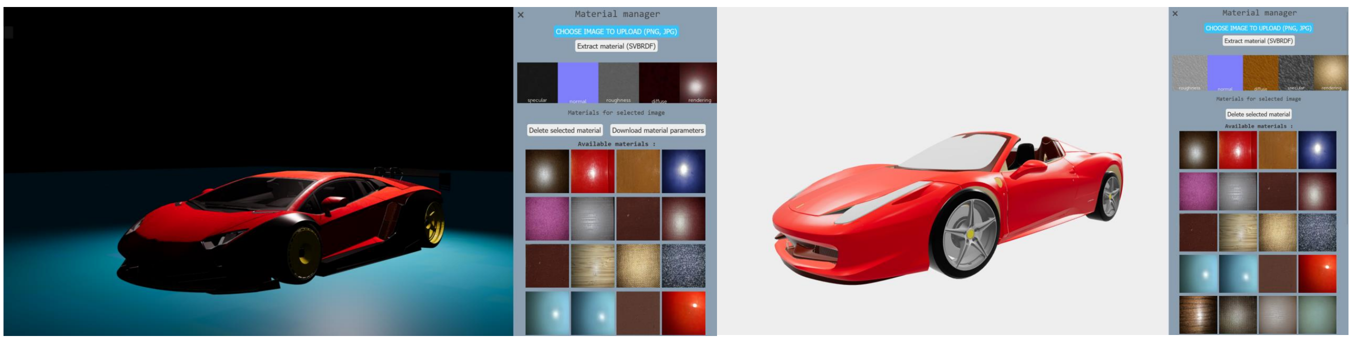

Figure 1: Snapshot of the client material manager with different estimated material assets applied to the above 3D models (Lamborghini [Performance 5 20] and Ferrari [vincent091036 2024])

ABSTRACT

Accurately replicating the appearance of real-world materials in computer graphics is a complex task due to the intricate interactions between light, reflectance, and geometry. In this paper we address the challenges of material representation, acquisition, and editing by leveraging the potential of deep learning algorithms our framework provide. To enable the visualization and generation of material assets from single or multi-view images, allowing for the estimation of materials from real world objects. Additionally, a material asset exporter, enabling the export of materials in widely used formats and facilitating easy editing using common content creator tools. The proposed framework enables designers to effectively collaborate and seamlessly integrate deep learning-based material estimation models into their design pipelines using traditional content creation tools. An analysis of the performance and memory usage of material assets at various texture resolutions shows that our framework can be used plausibly according to the needs of the end-user.

This work is licensed under a Creative Commons Attribution International 4.0 License.

WEB3D '24, September 25–27, 2024, Guimarães, Portugal
© 2024 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0689-9/24/09
https://doi.org/10.1145/3665318.3677152

CCS CONCEPTS

• **Computing methodologies** → **Reflectance modeling**; **Artificial intelligence**; **Image-based rendering**; • **Human-centered computing** → **Visualization systems and tools**.

KEYWORDS

3D graphics on the web, Deep image based BRDF estimation

ACM Reference Format:
Saptarshi Neil Sinha, Felix Gorsclüter, Holger Graf, and Michael Weinmann. 2024. DeepMaterialInsights: A Web-based Framework Harnessing Deep Learning for Estimation, Visualization, and Export of Material Assets from Images. In *The 29th International ACM Conference on 3D Web Technology (WEB3D '24), September 25–27, 2024, Guimarães, Portugal*. ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3665318.3677152

1 INTRODUCTION

Human perception of materials is affected by how light interacts with objects (reflection, scattering, and absorption), altering their appearance based on material properties like color and roughness. Therefore, photo-realistic rendering is crucial across various fields

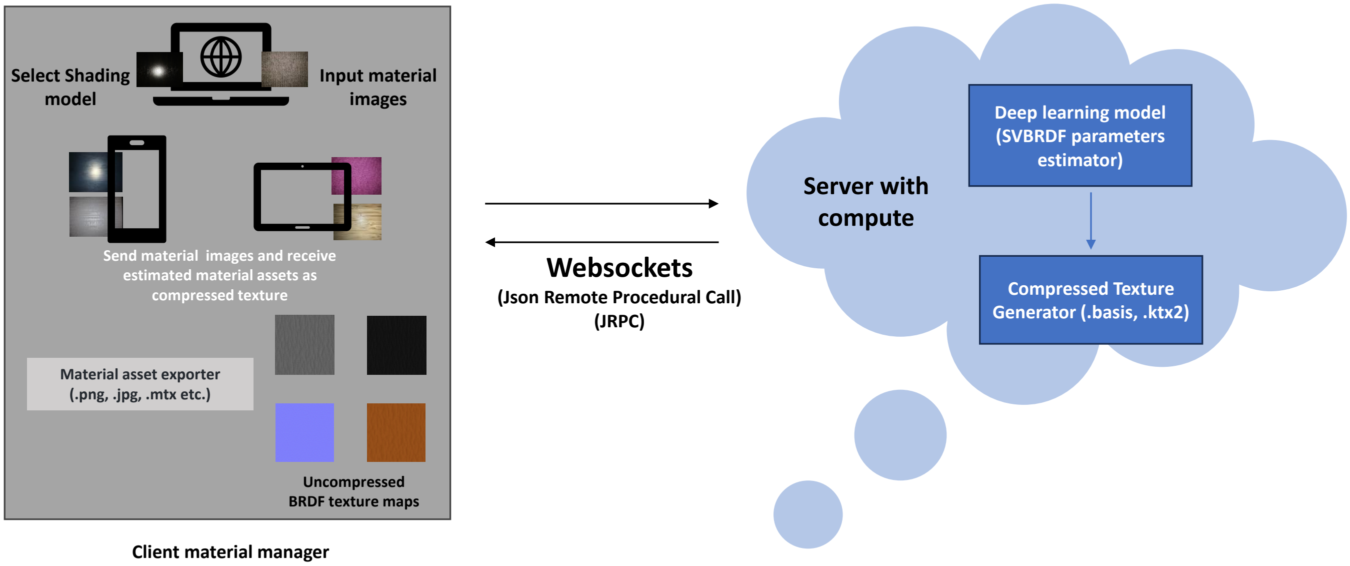

Figure 2: The architecture of our material predictor framework

such as visual effects, architectural modeling, cultural heritage, computer games, movies and automotive design. However, accurately replicating these material characteristics from real-world objects is challenging due to the lack of a universal material model. This results in significant issues in rendering systems, such as the need for uniform material representations, high costs, and additionally, material models are difficult to edit for artists and require substantial memory storage.

The appearance of real-world objects is a complex phenomenon resulting from intricate interactions between light, reflectance, and geometry. Lightweight appearance capture [Deschaintre et al. 2018; Han et al. 2023; Henzler et al. 2021] aims to extract reflectance functions from a limited number of photographs, which is a challenging task due to the ill-posed nature of the problem. This is because multiple reflectance properties can produce the same visual output in an image. Nevertheless, deep learning methods have shown promise in automatically learning effective priors from data, offering a new pathway to tackling this complex problem. Recent advancements in deep learning-based multi-view methods, such as Neural Radiance Fields (NeRF) [Mildenhall et al. 2020] and 3D Gaussians [Kerbl et al. 2023], have made significant progress in novel view synthesis, 3D reconstruction, and inverse rendering. Extensions to these methods [Boss et al. 2021; Jiang et al. 2023; Liu et al. 2023; Saito et al. 2024; Zarzar and Ghanem 2023] enable the decomposition of the estimated appearance into lighting and material properties, leading to more accurate and realistic rendering of scenes from different viewpoints. Single image or multi-view image deep learning based material asset estimation based on different shading models have opened up new opportunities for designers to generate material assets from real-world sources. However, single and multi-view material asset estimation methods like [Henzler et al. 2021; Jiang et al. 2023] have not been widely adopted in industrial applications. To bridge this gap, we propose a comprehensive framework that enables visualization, editing, and export of material assets. With this framework, designers can collaborate effectively, harnessing the advantages of these methods to perform their work more efficiently. The main contributions of this paper are:

- A framework that utilizes deep learning algorithms to visualize and generate material assets from either a single image or multiple view images. The framework allows for the editing of material assets generated from single images, which can be applied to various scenes. For multi-view material estimation, the generated atlases can be edited and applied specifically to the scene from which they were generated.
- Material asset exporter that offers the capability to export materials in widely used formats (.png, .jpg, and .mtx) and hence, the exported materials can be easily edited using common content creator tools.

2 RELATED WORK

Material acquisition is a critical aspect of creating realistic scenes [Guarnera et al. 2016]. Material management systems like Substance [Allegorithmic 2018] provide tools for organizing, accessing, and sharing materials, streamlining workflows for designers. Open standard formats like MaterialX [Foundation 4 24] have democratized the exchange process by enabling the representation, transfer, and management of materials across different content creation engines.

In the realm of single-image deep learning, methods such as [Aittala et al. 2016] have leveraged deep learning to extract material maps from 2D images. While single-image-based SVBRDF estimation like [Deschaintre et al. 2018] has shown promise, it is limited to low resolutions due to memory constraints. Advances in generative approaches, such as [Henzler et al. 2021], have allowed for the creation of high-resolution material assets, with recent works [Guo et al. 2023] aiming to further enhance quality.

Furthermore, multi-view image-based 3D scene reconstructions like Neural Radiance Fields (NeRF) [Mildenhall et al. 2020] and 3D Gaussian splatting (3DGS) [Kerbl et al. 2023] have made significant progress in generating realistic novel views. These advancements have facilitated multi-view material estimation using both implicit representations [Boss et al. 2021; Liu et al. 2023; Munkberg et al. 2022] and explicit representations [Jiang et al. 2023; Saito et al. 2024], demonstrating promising results for various surfaces, including shiny objects [Jiang et al. 2023; Liu et al. 2023].

3 MATERIAL PREDICTOR

Our framework estimates material assets (SVBRDF parameters of parameteric reflectance/shading models [Blinn 1977; Burley 2012; Cook and Torrance 1982; Guarnera et al. 2016; Karis 2013]) from single image or multi-view images leveraging deep learning algorithms. It can be used with various shading models, including those that handle anisotropic materials (only for multi-view images), as long as the underlying deep learning model supports this functionality. The framework is currently able to estimate materials from both single flash images (limited to stationary isotropic materials) and multi-view images. Additionally, for materials estimated from multi-view images, the framework also estimates image-based environment light, supporting both relighting and material editing.

3.1 Architecture and implementation

We propose a client-server architecture (Fig. 2) that utilizes JSON RPC over websockets for efficient data exchange. The client is a web application developed using ThreeJS and JavaScript, while the server hosts a deep learning model deployed using root-less Docker [Gomes et al. 2018] and Slurm [Yoo et al. 2003]. To ensure effective workload management and deployment, we containerize the model using Docker and configure Slurm for job scheduling on server nodes.

For single image material asset estimation, we employ a modified version of a generative model [Henzler et al. 2021], by changing the encoder network to a Vision transformer [Dosovitskiy et al. 2021] instead of ResNet50 [He et al. 2015] as the network needs less number of iterations during the fine-tuning step to generate qualitatively similar results. Additionally, our framework integrates different isotropic shading models, including the Phong [Blinn 1977], Isotropic Ward [Guarnera et al. 2016], Cook-Torrance [Cook and Torrance 1982], and Disney models [Burley 2012] which can be used for estimation of stationary isotropic materials.

For multi-view material asset estimation, we utilize a shading and lighting estimation network [Jiang et al. 2023] based on Gaussian splatting which uses the split-sum approach [Karis 2013].
The main features of our framework are the following:

- *Generating material assets from images:* The web-client allows users to choose a specific shading model and send image(s) to the server, which hosts deep learning models for estimating the SVBRDF (Spatially-Varying Bidirectional Reflectance Distribution Function) using various parametric shading models. The server processes the images and returns the corresponding material assets, such as diffuse and roughness, as compressed textures in the form of .basis files back to the client. The compression of these textures is performed using Basis Universal [Binomial 4 24], a GPU texture data interchange system that supports highly-compressed intermediate file formats (.basis or .KTX2). These compressed files can then be efficiently transcoded into various GPU compressed and uncompressed pixel formats, providing flexibility and optimization for rendering purposes. The performance and scalability of our framework is greatly enhanced using compressed textures without any percept ual loss in quality of the materials rendered.
- *Material asset visualization*: The material assets (generated from single flash images) can be applied to different 3D models, and the lighting conditions can be changed to obtain different perspectives and visualizations as shown in Figure 1. For multi-view material asset estimation, the generated atlases (Figure 4) can be optionally edited and applied to the same scene under different lighting conditions.
- *Material asset exporter:* In the client-side, the compressed material assets (.basis files) can be uncompressed and exported as texture-maps in .png or .jpg formats, allowing further editing in tools like Blender. Additionally, the assets can be exported in MaterialX [Foundation 4 24] format, facilitating integration into different rendering engines and 3D software for material definition exchange.

4 EVALUATION

In our analysis of the performance and memory footprint of material assets at different texture resolutions (Table: 1 and 2), we conducted the analysis on a server PC equipped with an RTX 3090 GPU and a client PC with an RTX 3070 GPU. The performance analysis assumed that the textures already exist on the server. However, it is important to note that for the initial conversion process of each material asset, when the material does not exist, it takes approximately 1 second on average for each material asset to be converted to the basis format.

For the performance analysis, we utilized a diverse set of material input images (Figure 3) to generate material assets with different resolutions using the Cook-Torrance BRDF model [Cook and Torrance 1982], Phong shading model [Blinn 1977], Ward isotropic model [Guarnera et al. 2016], and the Disney shading model [Burley 2012]. We reported the average time for the generation process.

The memory analysis was conducted on the same set of materials (Figure 3) using the Cook-Torrance shading model [Cook and Torrance 1982].

The analysis provides valuable insights into the trade-offs between

Figure 3: Material assets involved in the performance and memory footprint analysis

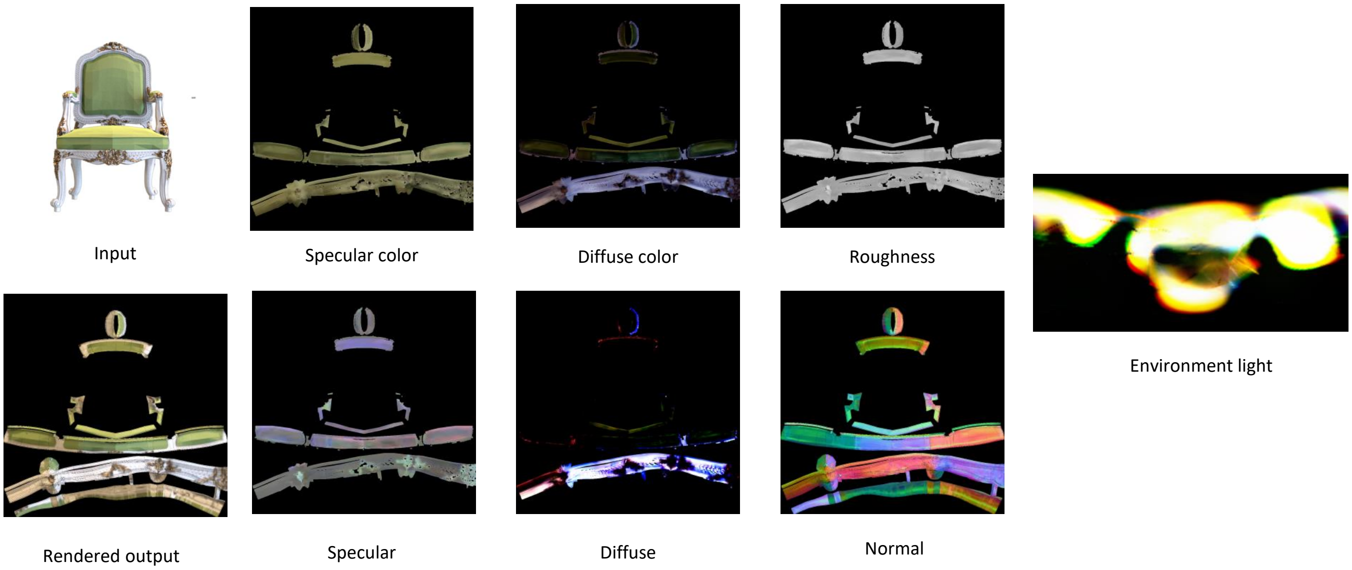

Figure 4: Material assets and environment light generated from multi-view images (top-left image represents one such view) using the Gaussian shader model [Jiang et al. 2023] which uses the split-sum approach [Karis 2013]. Our framework efficiently generates corresponding atlases for these material assets from different views, as shown above.

performance and memory usage when working with material assets at different texture resolutions. It assists in making informed decisions regarding texture compression and resolution selection to achieve a desired balance between rendering quality and resource consumption.

Table 1: Average response times for different texture resolutions (in pixels) considering different material assets and isotropic parametric shading models

Resolution	Avg. Time (in msec)	Texture compression
256	40	✓
	76	×
512	101	✓
	220	×
1024	185	✓
	701	×
2048	528	✓
	2518	×

Table 1 shows that the basis texture compression reduces the response time greatly, approximately it needs twenty-five percent of the time without compression at higher resolution. Moreover, we see this change as the memory footprint (Table 2) is reduced to one-tenth (on average) in case of using compressed textures considering all material assets of the Cook-Torrance shading model. Based on our evaluation of quality and performance, we recommend a resolution of 1024 pixels.

Table 2: Average memory footprint (in KB) of material assets at different resolution (in pixels)

Resolution	Material Assets				Texture Compression
	Diffuse	Roughness	Specular	Normal	
256	85	31	25	59	×
	7	7	6	2	✓
512	360	134	106	230	×
	29	31	25	3	✓
1024	1450	545	470	1056	×
	96	106	86	64	✓
2048	5755	885	703	3512	×
	386	12	118	45	✓

5 LIMITATION

Single image material asset estimation is limited to stationary isotropic materials. Anisotropic materials are important for many industrial and cultural heritage applications where our single image estimation will fail. The material asset estimation from multi-view images works better for a larger set of materials, including shiny objects, but is constrained to a particular scene (Figure 4). Users currently can only select from the implemented shading models to estimate material assets and they cannot customize any shading models and parameters to estimated themselves.

6 CONCLUSION AND FUTURE WORK

We have developed an architecture that utilizes a deep learning model to generate material assets from images. This architecture is versatile and can be applied to various 3D assets, with the ability to export them into different formats for further usage. Our framework allows for estimating material assets for different types of shading models, using both single and multi-view images. The multi-view material asset estimation also estimates the image-based environment light, which enables the relighting of the current scene. Along with that, users can edit the material asset atlases using 3D content editing tools, allowing customization and modifications to the material properties.

A valuable enhancement for the framework would be the capability to segment different material types within a scene. This segmentation can then be applied to other scenes using the multi-view material estimation models, providing greater versatility in material application. Furthermore, the architecture can be further enhanced to incorporate a custom shading model that the deep learning models can estimate. This would enable more accurate estimation of material assets and provide designers with greater flexibility in defining their desired shading characteristics. By integrating these features, the framework would facilitate efficient communication and real-time updates, enabling a dynamic and interactive workflow for material asset creation and management.

ACKNOWLEDGMENTS

The work presented in this paper has been funded by the European Commission during the project PERCEIVE under grant agreement 101061157.

REFERENCES

Miika Aittala, Timo Aila, and Jaakko Lehtinen. 2016. Reflectance modeling by neural texture synthesis. *ACM Trans. Graph.* 35, 4, Article 65 (jul 2016), 13 pages. https://doi.org/10.1145/2897824.2925917

Allegorithmic. 2018. Substance Share. https://share.allegorithmic.com/.

Binomial. Accessed: 2024-04-24. Basis Universal Supercompressed GPU Texture Codec. https://github.com/BinomialLLC/basis_universal.

James F. Blinn. 1977. Models of light reflection for computer synthesized pictures. In *Proceedings of the 4th Annual Conference on Computer Graphics and Interactive Techniques* (San Jose, California) *(SIGGRAPH '77)*. Association for Computing Machinery, New York, NY, USA, 192–198. https://doi.org/10.1145/563858.563893

Mark Boss, Raphael Braun, Varun Jampani, Jonathan T. Barron, Ce Liu, and Hendrik P.A. Lensch. 2021. NeRD: Neural Reflectance Decomposition from Image Collections. In *IEEE International Conference on Computer Vision (ICCV)*.

Brent Burley. 2012. Physically-Based Shading at Disney. https://api.semanticscholar.org/CorpusID:7260137

R. L. Cook and K. E. Torrance. 1982. A Reflectance Model for Computer Graphics. *ACM Trans. Graph.* 1, 1 (jan 1982), 7–24. https://doi.org/10.1145/357290.357293

Valentin Deschaintre, Miika Aittala, Fredo Durand, George Drettakis, and Adrien Bousseau. 2018. Single-image SVBRDF capture with a rendering-aware deep network. *ACM Transactions on Graphics* 37, 4 (July 2018), 1–15. https://doi.org/10.1145/3197517.3201378

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. arXiv:2010.11929 [cs.CV]

Academy Software Foundation. Accessed: 2024-04-24. MaterialX. https://materialx.org/.

Jorge Gomes, Emanuele Bagnaschi, Isabel Campos, Mario David, Luís Alves, João Martins, João Pina, Alvaro López-García, and Pablo Orviz. 2018. Enabling rootless Linux Containers in multi-user environments: The udocker tool. *Computer Physics Communications* 232 (2018), 84–97. https://doi.org/10.1016/j.cpc.2018.05.021

Dar'ya Guarnera, Giuseppe Claudio Guarnera, Abhijeet Ghosh, Cornelia Denk, and Mashhuda Glencross. 2016. BRDF Representation and Acquisition. *Computer Graphics Forum* 35 (2016). https://api.semanticscholar.org/CorpusID:13847665

Jie Guo, Shuichang Lai, Qinghao Tu, Chengzhi Tao, Changqing Zou, and Yanwen Guo. 2023. Ultra-High Resolution SVBRDF Recovery from a Single Image. *ACM Trans. Graph.* 42, 3, Article 33 (jun 2023), 14 pages. https://doi.org/10.1145/3593798

Yuxuan Han, Junfeng Lyu, and Feng Xu. 2023. High-Quality Facial Geometry and Appearance Capture at Home. arXiv:2312.03442 [cs.CV]

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Deep Residual Learning for Image Recognition. arXiv:1512.03385 [cs.CV]

Philipp Henzler, Valentin Deschaintre, Niloy J Mitra, and Tobias Ritschel. 2021. Generative Modelling of BRDF Textures from Flash Images. *ACM Trans Graph (Proc. SIGGRAPH Asia)* 40, 6 (2021).

Yingwenqi Jiang, Jiadong Tu, Yuan Liu, Xifeng Gao, Xiaoxiao Long, Wenping Wang, and Yuexin Ma. 2023. GaussianShader: 3D Gaussian Splatting with Shading Functions for Reflective Surfaces. *arXiv preprint arXiv:2311.17977* (2023).

Brian Karis. 2013. Real Shading in Unreal Engine 4. http://blog.selfshadow.com/publications/s2013-shading-course/karis/s2013_pbs_epic_notes_v2.pdf

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 2023. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Transactions on Graphics* 42, 4 (July 2023). https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/

Yuan Liu, Peng Wang, Cheng Lin, Xiaoxiao Long, Jiepeng Wang, Lingjie Liu, Taku Komura, and Wenping Wang. 2023. NeRO: Neural Geometry and BRDF Reconstruction of Reflective Objects from Multiview Images.

Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. 2020. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In *ECCV*.

Jacob Munkberg, Jon Hasselgren, Tianchang Shen, Jun Gao, Wenzheng Chen, Alex Evans, Thomas Müller, and Sanja Fidler. 2022. Extracting Triangular 3D Models, Materials, and Lighting From Images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 8280–8290.

SDC Performance. Accessed:2024-05-20. Lamborghini Aventador SVJ SDC (FREE). https://sketchfab.com/3d-models/lamborghini-aventador-svj-sdc-free-784e4656aca649cca55d6b18740a19b2.

Shunsuke Saito, Gabriel Schwartz, Tomas Simon, Junxuan Li, and Giljoo Nam. 2024. Relightable Gaussian Codec Avatars. In *CVPR*.

vincent091036. Accessed on : 31.05.2024. Ferrari 458. http://ivr-ahnu.cn/app/three/r107/examples/webgl_materials_cars.html.

Andy B. Yoo, Morris A. Jette, and Mark Grondona. 2003. SLURM: Simple Linux Utility for Resource Management. In *Job Scheduling Strategies for Parallel Processing*, Dror Feitelson, Larry Rudolph, and Uwe Schwiegelshohn (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 44–60.

Jesus Zarzar and Bernard Ghanem. 2023. SplitNeRF: Split Sum Approximation Neural Field for Joint Geometry, Illumination, and Material Estimation. arXiv:2311.16671 [cs.CV]