Comparative Evaluation of Deep-Learning-Based Generative Methods for Material Generation in 3D Modelling Workflow

Nejc Hirci[1], Nejc Lešek[2],
Žiga Lesar[1][0000−0002−9850−0497], and Ciril Bohak[1][0000−0002−9015−2897]

[1] Faculty of Computer and Information Science, University of Ljubljana, Večna pot 113, 1000 Ljubljana, Slovenia
[2] Lightmass Dynamics, Badovinčeva ulica 20, 3270 Laško, Slovenia
ciril.bohak@fri.uni-lj.si

Abstract. This paper presents a comparative evaluation of two state-of-the-art deep-learning-based generative methods, *NeuralMaterial* and *MaterialGAN*, for material generation in 3D modeling application – Blender. For comparison, we introduce a baseline method developed using only tools available in Blender. A user study involving 20 participants, including both experts and lay users, was conducted to assess usability and effectiveness. Participants engaged in a material authoring task, evaluating all developed approaches. Usability was evaluated using the System Usability Scale (SUS) questionnaire, and material fidelity was validated by comparing user-created materials to target scene renders using Learned Perceptual Image Patch Similarity (LPIPS) metric. Users are able to achieve the desired results with a similar level of precision as with the existing tools but with higher editability. Results also show preference differences between lay and expert users on which approaches they prefer and why. Our research provides valuable insights into deep-learning-based material generation integrated into everyday workflows. It highlights the importance of considering user preferences and expertise levels when designing such workflows. By comparing performance and usability, we inform the development of more effective and accessible material generation techniques within the field.

Keywords: Procedural materials · 3D modeling integration · generative models · user study.

1 Introduction

Creating and editing materials in 3D modeling workflows poses significant challenges, as illustrated by several recent publications in the field. While numerous papers have proposed methods to support material acquisition and editing, they often overlook the crucial aspect of user experience. This paper investigates how modern deep generative models can enhance existing pipelines for material generation, focusing on improving usability in everyday 3D modeling workflows.

Photorealistic rendering is essential in various industries, including film, video games, and advertising, where convincing virtual scenes captivate audiences. Traditionally, physically accurate rendering methods have been developed to simulate light interactions with 3D models, ensuring realistic visual outcomes. However, acquiring accurate optical representations of materials can be labor-intensive and expensive, often requiring controlled photography setups or intricate measurement devices. In such cases, the Physically Based Rendering (PBR) [28,23] can be used to render the images (e.g., IKEA often uses such approach in their catalogues [13]).

To address these challenges, simpler representations of PBR materials have been introduced in 3D graphics software. Despite sharing a theoretical foundation, these systems differ significantly in their user interfaces and workflows. This research aims to bridge the gap between existing approaches and modern machine learning techniques to enhance material generation and editing processes.

A very precise but complex method for extracting the optical properties of materials needed for photorealistic rendering requires a large number of deliberately taken photographs with predetermined positions of the object and the camera in a controlled environment. There are datasets of carefully measured Bidirectional Reflectance Distribution Functions (BRDF) of materials, such as the MERL BRDF Database [21], which contains 100 different materials. Such datasets can be acquired using methods similar to one presented by Dupuy and Jakob [7], which uses a custom goniophotometer. This approach provides very high precision results much faster, which is crucial for the deeper study of light phenomena but is often meaningless for commercial applications. For this reason, many 3D applications support simpler representations of PBR materials, very often following Disney's [22], NVIDIA's [26] or Industrial Light & Magic's [34] implementations. Although many of these systems have a similar theoretical basis, the user interfaces for setting their parameters and how they affect the material properties can be very different.

In our research, we were interested in the shortcomings of the existing approaches and how they could be improved with the help of state-of-the-art machine learning methods. We focused on the field of Spatially Varying Bidirectional Reflectance Distribution Function (SVBRDF) capture, where researchers proposed neural network models to extract material textures directly from flash photos. To accurately compare two of the most promising machine learning approaches *MaterialGAN* [11] and *NeuralMaterial* [12], we implemented them as an add-on for Blender[3] and analyzed their usability through a user study. We also implemented a baseline approach using only the Blender built-in tools [3]. All three implementations offer a form of procedural texture generation from flash photos and a simple form of editing that requires no knowledge of PBR principles.

To assess the usability of these approaches, we conducted an extensive user study involving participants with varying levels of expertise in 3D modeling software. The study utilized a survey to capture users' prior knowledge and

[3] `https://www.blender.org`

experience alongside the widely accepted SUS questionnaire [1,18]. Participants were tasked with populating a given 3D scene with materials closely matching reference materials in a target scene generated by each approach. User performance, including task duration times, the accuracy of the results, and qualitative observations, were recorded and analyzed.

By evaluating the user experience and effectiveness of different material generation approaches, this research aims to advance the understanding of how machine learning models can enhance usability and efficiency in everyday 3D modeling workflows. The contributions presented in this paper are:

- New publicly available Blender add-on with a user-friendly interface for supporting deep-learning-based material extraction and editing workflows (see our GitHub repository[4]).
- Integration of two state-of-the-art generative material models in our add-on with additional editing options and additional deep-learning-based support tools (i.e., texture upscaling).
- A user study methodology for analyzing the usability of integrated approaches on a material editing task and a comprehensive user study for evaluating the usability of deep generative material models in practical applications.

2 Related Work

The problem of material acquisition is a small subset of the inverse rendering problem, which represents a 3D virtual scene in a way that allows to photo-realistically re-render novel views. Recent advances in deep learning methods, as outlined in a survey on intrinsic image decomposition [9], have demonstrated significant progress with the use of neural networks as universal scene approximations and as differentiable renderers. The latter is particularly noteworthy as it allows for end-to-end inverse estimation for a wide range of scene parameters based on PBR principles. This also leads to significant improvements in the material acquisition from flash photos. To solve a more complex problem, Prakash et al. [30] have recently presented an approach for estimating materials for the whole scene. To estimate the material specularity Jin et al. [16] integrate reflectance guidance and shadow/specular aware learning. Lately, scientists have tackled the complexities of reconstructing scenes from images using Neural Radiance Fields (NeRF) models [25,20]. These models have the capacity to modify different aspects of the extracted scene, such as materials using the learned latent space. The significance of these capabilities underscores the need for further investigation through user studies, aligning with the methodologies we design in our work.

2.1 User Studies in AI-supported design problems

Assessing the usability of machine-learning-supported tools in practical applications is a complex endeavor, requiring the integration of a neural model that

[4] `https://github.com/NejcHirci/material_addon`

generates high-quality textures at satisfactory image resolutions and a user-friendly graphical interface offering good functionalities. Some of the described work briefly included user study assessments to support their findings, but these assessments often received limited attention, resulting in an incomplete understanding of the applicability of AI-supported design tools. Our investigation emphasizes evaluating deep generative material models in practical workflows to show their most important practical contributions.

Considering work focused on high-level design problems, such as authoring PBR materials, Shimizu et al. [33] created a framework for building parameterized design tools in high-dimensional design spaces. The building blocks of their framework are design adjectives, which are machine learning models of user intent for a guided exploration of high-dimensional spaces. They evaluated the usability of their framework with a user study, where users performed three design tasks, two using their framework and one using a baseline configuration with sliders. Results showed that adjectives support the exploratory design and provide a better approach than the baseline approach. We partly followed their design in our user study, where we similarly mimicked two machine-learning-based approaches supported with a sequential gallery user interface. However, because our research focused on both experienced and inexperienced users, we had to adapt the baseline method with a slider interface to offer only simple interpolations instead of full parameter control. The idea behind our sequential gallery interface for latent space exploration is closely related to the proposed novel user-in-the-loop optimization method presented by Koyama et al. [17]. Their approach uses sequential plane search based on Bayesian optimization, where users can easily zoom into and out of the extracted latent space plane. It is also worth mentioning that both methods, while highly applicable to established parameter spaces, are not directly intended for the latent space exploration of learned machine learning models.

Comparatively, our work builds from the established ideas in AI-supported design interface tools. Unlike previous work, we comprehensively evaluate the usefulness of state-of-the-art material models in practical applications.

Most importantly, we perform a comprehensive user study, tracking users' performance and satisfaction when using each integrated pipeline on a simple material editing task. This has been only partially analyzed in some of the mentioned material-editing approaches like [41,40] and some of the material modeling papers [12,15], but without a proper comparison across different editing models and without comprehensively analyzing differences in usability for both novice and expert users.

3 Methodology

Our research aims to evaluate the practical user application of machine-learning approaches for material generation. Often, these approaches lack proper analysis in real-world scenarios, where user experience relies on seamless integration into graphical user interfaces. To address this, we implemented interfaces for three material generation methods, two using deep generative models and a baseline method. We designed these interfaces to minimize differences between the methods, allowing us to focus on variations in functionality between two machine-learning-supported approaches and the baseline approach. Our main objective was to compare the benefits and drawbacks of machine learning pipelines to a baseline. While designing the interface, we prioritized simplicity to observe variations in user experience between two distinct groups: lay users with no prior knowledge of PBR principles and industry experts. Our user study aims to offer valuable insights into the practical requirements of material artists and potentially uncover intriguing pathways for the future design of user experiences incorporating machine learning.

3.1 Designing material add-on workflows

We decided to keep the baseline approach simple because introducing any additional complexity would prevent us from including lay users in our study. After exploring some of the community-developed add-ons and tools, we found a fully integrated approach by Pathik [27], which uses only the baseline shader node systems. His very simple pipeline can extract material textures like albedo, specular, roughness, and others directly from photos with automatic image transformations and mathematical operations. For the editing functionality, we selected eight materials from different categories (wood, metal, etc.) designed by expert Blender artists, which the user could use for blending with the extracted material. The blending operation utilized the *Mix* node offered in Blender, which works as a simple pixel interpolation of the extracted and selected material. This meant that the user could modify an extracted material without understanding the underlying shaders.

The first selected machine-learning-based approach used the *MaterialGAN* [11] model and was shortly presented in the previous section. Again, the approach has considerable practical limitations and produces textures with a resolution of only 256×256 pixels. At the time of conducting the user study, the approaches [39,14] were not yet published, which is why we had to find an alternative to combat this limitation. After performing additional testing, we decided to use an additional resolution upscaling model [6], which focuses on preserving continuous details present in the low-resolution input image. This meant that we could generate higher-resolution textures with a reasonably low amount of artifacts between different material parameters. A considerably more important decision was how to utilize the learned latent space of the model for the editing options that we wanted to provide. Researchers recommended interpolating the selected material with its randomly sampled neighbors. This provided inconsistent results for the iterative editing approach we were aiming for because it limited the overall editing options present in the latent space. In light of that, Shen and Zhou [32] presented an interesting unsupervised approach, which can be effectively implemented on various models based on the Generative Adversarial Network architectures (GAN) architecture. Their approach decomposes the learned GAN weights in a method similar to principal component analysis and helps extract the most semantically relevant latent eigenvectors. We employ this approach to provide editing capabilities by allowing the user to navigate through the learned latent space of the model. This navigation occurs around the currently selected material, with options to move in any of the eight directions corresponding to the neighboring materials in the material gallery. The neighboring materials are generated by shifting the current material latent vector by a small step in the direction of one of the eight extracted vector directions. The step was fixed and chosen empirically so as not to confuse lay users with additional complex editing properties. It is important to note that the *MaterialGAN* model does not adequately account for variations in rotation among the input flash photos. Therefore, an additional preprocessing step is necessary to project the material in the photo onto a plane parallel to the camera. In the research paper, the authors addressed this issue by capturing flash photos using an A4 paper with a cutout in the middle for the target material and fiducial marks surrounding the cutout, which could then be used to compute the necessary projection matrix. Initially, they utilized the AprilTags [36] fiducial markers, but we found the available Python implementation to be subpar. As a result, we opted for the approach presented by Garrido-Jurado et al. [10], which also uses fiducial markers.

For comparison, we implemented another machine learning approach *NeuralMaterial* [12]. This model is based on a more general encoder and decoder architecture and supports generalization across spatial positions of the material, which means that it can generate textures of arbitrary resolution. More importantly, the authors present learned latent space interpolations examples, showing more consistent color reproductions than the MaterialGAN model. Because the model is learned by integrating a random infinite noise field, it can also generate different variations of the same material using different seeds, which is another editing option that we integrated into our workflow. At the time when the user study was conducted, this proved to be one of the most versatile generative models in material generation research.

3.2 Blender add-on implementation

We implemented the three approaches described above within Blender due to its popularity and open-sourced nature. Our add-on was developed in Python using Blender's Python API [3], which allows for the seamless integration of custom user interface components inside Blender. This grants access to most of Blender's functionalities through convenient function wrappers. Throughout the development process, we adhered to the add-on development guidelines provided by the Blender Community, which encompassed recommendations on code style, utilization of the integrated Python environment, and add-on compatibility.

We created a unified interface (Figure 1 shows the MaterialGAN implementation) where the user can select the approach from a drop-down menu and is then provided with the appropriate material extraction and editing options. Following the ideas of papers on design interfaces [33,17], we decided to implement our iterative editing approaches as a sequential material gallery interface. The baseline approach does not support a similar interpolation style. Blender Python API restricts executing the shader pipeline in a separate thread, preventing us from creating eight neighboring materials in the add-on without blocking the program's user interface. Because of this limitation, we had to change the interface for the baseline approach to a simple slider with a drop-down selection of interpolation materials. For both of the machine learning approaches, the material is extracted from a flash input photo, and the user can select a material from a gallery of eight generated neighboring interpolated materials. On each traversal step, new neighboring materials are generated, which offers the user infinite editing options, regardless of the selected approach.

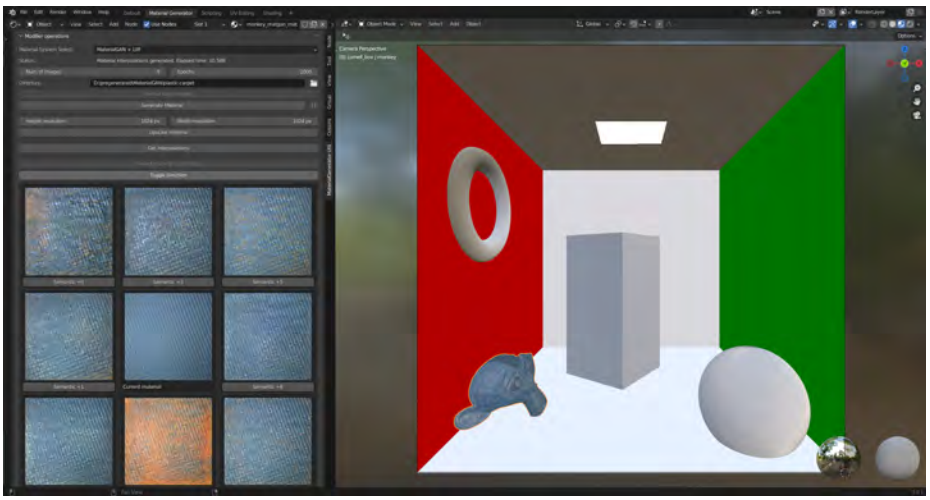


Fig. 1: User interface of the Blender add-on. On the left are the parameters of the selected approach and the neighboring materials image gallery. On the right is a scene the user should populate with generated materials which should be as close to the target ones as possible.

3.3 User study methodology

Our study focused on evaluating the efficacy of machine learning techniques in material extraction and editing tasks within real-world workflows compared to a baseline approach. To achieve this, we conducted an extensive user study that replicated practical material design scenarios. We also analyzed the correlation between participants' previous experience in physically-based computer graphics tasks and their utilization of all three available workflows. The study consisted of three material editing tasks in which participants used a specific workflow in our Blender add-on to fill a Cornell box scene, seen in Figure 2, containing four objects with materials that matched a target scene as closely as possible.

Since material extraction varies from approach to approach and takes a long time, we presented participants with a list of 20 pre-generated materials. Then participants were asked to edit the scene using each implemented approach. We measured task completion time and accuracy using LPIPS [38] and obtained feedback on the usability of each approach using the SUS. We also gathered participants' demographic data and information about prior experience with 3D computer graphics, material creation, and editing tasks. We formulated several hypotheses to guide our research:

- H1: All of the implemented approaches will be suitable for use for material design tasks (SUS score will be equal or better than "OK").
- H2: Participants prefer machine learning approaches because of more editing options.
- H3: Participants will rank the baseline approach with the highest usability.

Fig. 2: The provided initial Cornell's Box scene with target objects without materials.

Participants We recruited a diverse group of 20 participants with varying experience levels in 3D computer graphics and material design tasks. The group consisted of 17 male and 3 female participants. Of the 20 participants, 10 were between 18 and 24 years old, 2 were between 25 and 34 years old, 4 were between 35 and 44 years old, and 4 were between 45 and 54 years old. The participants also had different levels of education: 10 participants had graduated from high school, 3 participants had completed professional higher education, 2 participants had a bachelor's degree, 1 participant had a master's degree, and 4 participants had a doctoral degree. Participants' educational backgrounds included various qualifications, working in higher education, game development, and graphical design studios, which ensured a well-rounded perspective during the study. Concerning prior knowledge of PBR principles and experience with material design tasks, 10 participants were classified as experienced users and 10 as inexperienced.

Task Implementation We performed our user study during quarantine and held online sessions with individual participants. Both the moderator and the participant connected to a remote computer with a pre-prepared environment. To better understand users' workflow, we used screen recording during each online session. First, participants had to fill out a short list of questions regarding their demographic background, completed level of education, prior experience with 3D computer graphics, and material design tasks. We prepared the target scenes for the material editing tasks by utilizing only the tools offered by each of our implemented approaches to ensure that the participants could achieve the provided target.

After completing the initial survey, participants proceeded to the three design tasks. Each required them to watch a brief video demonstration showcasing the processing options. This video aimed to give participants a visual understanding of the available functionalities and parameters and how they could be used. Afterward, participants were asked to provide feedback on the demonstration's clarity and task instructions. The order of tested workflows was selected at random.

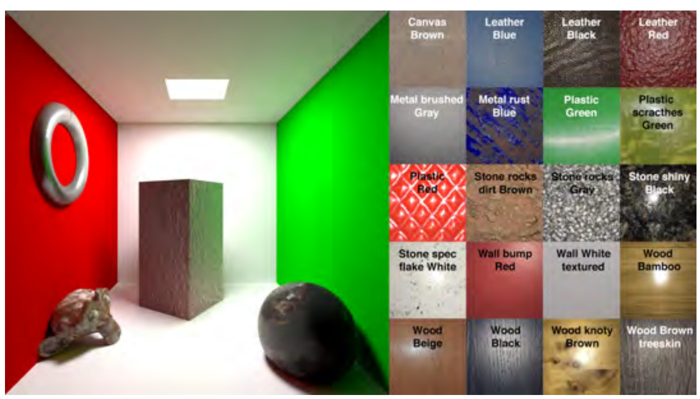


Fig. 3: List of pre-generated materials and target scene for the baseline method.

Next, they started working on processing materials for the empty scene. They had an image of the target scene they were trying to recreate and a list of pre-made materials they could choose from, shown in Figure 3.

During editing, the 3D scene was updated in real-time as participants applied and edited new materials. In the case of the machine learning approaches, participants were also offered previews of each selected material in the material gallery as renders of flat exemplars under flat flash lighting.

After completing the given design task, participants were asked to complete a SUS questionnaire to measure the suitability and usefulness of the provided workflow. After completing all three tasks, participants were also asked to rate the usability and to provide additional comments for each approach. A detailed User Study Plan and the instructions provided to the participant can be found in the Supplemental material.

A moderator was present throughout the user study to assist participants with further questions. In addition, because the study sessions were conducted online and with each participant individually, participants were asked to think aloud while completing the tasks, which gave us additional insight into their experience.

3.4 User study design

The independent variable in the user study was the implemented approach used for the task. The three tested approaches described in Section 3.1 were the baseline approach *Algorithmic*, and the machine learning approaches, *MaterialGAN* and *NeuralMaterial*.

The measured dependent variables were: task completion time, task accuracy measured as the perceptual similarity between the target and edited scene rendering, and the SUS score on a scale from $0 - 100$.

Completion time was measured as the time required to complete each design task, from the point at which the user opened the Blender program to the point at which the scene was saved. In a few cases, functional errors occurred, which we accounted for by subtracting the time it took to restart the program from the recorded screen captures.

Task accuracy was measured using LPIPS [38], a measure of the perceptual difference between two images that focuses on the similarity relevant to humans. Existing metrics that compare pixels, such as Euclidean distance and Peak Signal-to-Noise Ratio (PSNR) [37], assume pixel independence and cannot capture important structural features in images (i.e. blurring causes big Euclidean, but small perceptual difference). Researchers found the learned internal representation of convolutional neural models can represent the best approximation of human perceptual loss. Following this, we used a perceptual loss (LPIPS) as the main metric for comparing the user-produced rendered image of the scene to the rendered image of the target scene.

Usability, the most important aspect in our study, was measured using SUS score, which relies on the verified translation of the original SUS questionnaire [4,18] to Slovene language [2]. Despite being first introduced in 1986 by John Brooke, it is still widely used due to its simplicity. It consists of 10 questions on a 5-point Likert scale, which alternate between positive and negative sentiment. Later, researchers presented an extended plausible analysis of the SUS questionnaire results [19]. They identified four key dimensions of usability related to the SUS questionnaire: perceived complexity, ease of use, consistency, confidence in use, and learnability, which could be analyzed by comparing averages of specific statements to the expected averages given the final score of a system. With this we could analyze the deviation of each system along the four identified dimensions gaining further insights into differences in usability between the approaches.

Lastly, we added a set of general questions to obtain participants' demographic information and information on their prior experience with PBR material systems. For each approach, we also added two short questions where users could give feedback on the clarity of the provided video demonstration and task instructions. At the end of the session, the participants had to rate all three workflows based on ease of use and could give any additional comments and feedback. The full questionnaire is presented in the Supplemental material.

4 Results

We analyzed the user study results from several aspects presented in the following subsections. Firstly, we checked users' prior knowledge. Secondly, we evaluated performance and usability. Thirdly, we analyzed user observations and feedback. Finally, we analyzed the times it took users to complete the tasks for each workflow.

4.1 Questionnaire on prior experience and task clarity

Table 1 shows responses to questions about experience and background knowledge, measured on a 5-point Likert scale. The participants were divided into two groups: experienced and lay users, based on their scores in the table. To determine the separation between the two groups, we used a combination of cutoff scores, particularly focusing on the questions of prior experience in creating and editing materials, familiarity with PBR principles, and understanding of essential material parameters. In the second statement, 8 participants completely agreed to have previously worked on material design tasks, with two partially agreeing and the rest disagreeing to varying degrees. An even clearer distinction can be seen in the third statement, where 10 participants completely agreed to be familiar with the PBR principles, similarly, for the fourth statement, where the number was even a little bit higher (11).

By analyzing the results, we discovered the most important statements separating experienced users from lay users. As seen in Table 1, two visible peaks are especially visible in the third and fourth statements, which explicitly refer to their prior knowledge and understanding. Based on a combined cutoff value of 9 for these two statements, we could already divide the participants into groups of experienced and lay users, each containing 10 participants. Even though we lowered the focus on prior experience with 3D computer graphics compared to understanding material design, only 2 participants showed a large difference, with one having a lot of prior experience but little specific knowledge and the other having a lot of prior knowledge with little experience. We decided to keep the first participant with the lay users and the second one with the experienced users, as we considered the understanding of material design to be a much more important aspect of our study.

Table 1: Responses to questions about experience and background knowledge on a 5-point Likert scale.

Statement	1	2	3	4	5
I have worked with one of the 3D computer graphics programs (Blender, Maya, 3ds, etc.)	3	3	3	1	10
I have used one of the mentioned programs for creating and editing materials	6	2	2	2	8
I am familiar with established PBR principles in computer graphics	3	5	1	1	10
I know and understand material parameters such as normals, roughness, specular, and albedo	1	2	4	2	11
I have created textures for materials from images or photographs	7	1	3	2	7

Additionally, after viewing the video demonstration and before beginning each task, participants were asked to rate the clarity of the task instructions and video demonstration. Most participants fully agreed that the task instructions and video demonstration were understandable: 19 for *NeuralMaterial*, 19 for *MaterialGAN*, and 18 for *Algorithmic* approach. The rest of the participants felt that the video demonstration and task instructions were clear enough.

4.2 Performance and usability evaluation

We used a general linear model to assess the possible statistically significant differences between the implemented workflows using Analysis of variance (ANOVA) in terms of the previously defined dependent variables. We calculate average values (AVG), standard deviations (STD), minimal (MIN), and maximal (MAX) values for each workflow over responses from all users.

All three workflows yielded a very similar SUS score ($p = 0.79669$) and very similar minimum and maximum scores, as can be seen in Table 2 and the corresponding error plots shown in Figure 4a. This is also true when looking at the differences between experienced ($p = 0.50210$) and inexperienced ($p = 0.67335$) users, with the differences between approaches proving just slightly more significant in the experienced group. We computed the SUS questionnaire scores for each user for each workflow based on the presented methodology [1,18]. The interpretation of the SUS score results is based on the guidelines provided in the methodology.

The task completion time for each task ($p = 0.00181$) was determined as the time from the start of the task (starting the prepared Blender environment) and the end of the task (saving the final scene). The results in Table 3 and the corresponding error plots in Figure 4b show the baseline approach was consistently faster than the machine learning-based approaches, with *MaterialGAN* being the second fastest and *NeuralMaterial* the slowest. Differences in task completion

Table 2: Results showing the average (AVG), minimum (MIN), maximum (MAX) SUS score for all workflows, their grade, and their standard deviations (STD).

Workflow	↑ AVG	STD	MIN	MAX	↑ Grade
Algoritmic (Baseline)	80.00	16.72	45.0	97.5	A-
MaterialGAN	80.16	15.09	47.5	100	A-
NeuralMaterial	**83.00**	15.80	40.0	100	**A**
Exp. *Algoritmic*	77.25	18.46	48.0	98	A-
Exp. *MaterialGAN*	83.50	18.34	48.0	100	A
Exp. *NeuralMaterial*	**86.00**	13.60	58.0	100	**A+**
Lay *Algoritmic*	**82.75**	15.25	45.0	98	**A**
Lay *MaterialGAN*	76.75	10.93	63.0	100	A-
Lay *NeuralMaterial*	80.00	17.95	30.0	100	A-

times between the three approaches remain significant among lay users ($p = 0.00164$). This is not true for experienced users ($p = 0.19600$), where there are no significant differences between the provided approaches.

Table 3: Results showing average (AVG), minimum (MIN), maximum (MAX) task completion times for all workflows, and their standard deviations (STD).

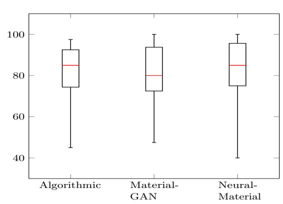
Workflow	↓ AVG (s)	STD (s)	MIN (s)	MAX (s)
Algoritmic (Baseline)	**947.45**	399.53	304	1532
MaterialGAN	1126.3	390.46	584	1938
NeuralMaterial	1430.5	440.65	857	2568
Exp. *Algoritmic*	**901.2**	469.09	304	1532
Exp. *MaterialGAN*	926.2	277.33	584	1449
Exp. *NeuralMaterial*	1165.4	267.41	857	1660
Lay *Algoritmic*	**993.7**	334.95	555	1466
Lay *MaterialGAN*	1326.4	394.93	683	1938
Lay *NeuralMaterial*	1695.6	426.90	1157	2568

Task completion accuracy ($p = 0.05618$) was measured as the LPIPS between the rendered target scene and rendered user scene for each task and each user. The target scene that the users had to create in each instance was produced with the corresponding implemented workflow. The results presented in Table 4 and the corresponding error plots in Figure 4c proved to be very similar between all three approaches, with the difference being slightly more pronounced in the lay user group ($p = 0.00676$) and slightly less pronounced in the experienced user group ($p = 0.54353$).

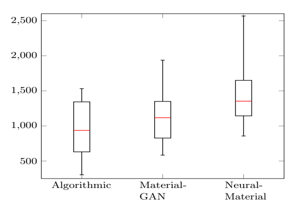
By utilizing the calculated averages for each statement in the SUS questionnaire, we have observed disparities between experienced and lay users regarding the usability dimensions mentioned earlier: perceived complexity, ease of use,

Table 4: Results showing average (AVG), minimum (MIN), and maximum (MAX) LPIPS values for all workflows.

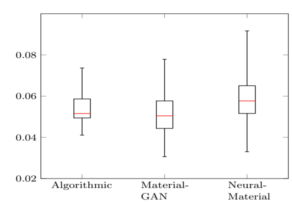
Workflow	↓ AVG	STD	MIN	MAX
Algorithmic (Baseline)	0.05374	0.00762	0.04108	0.07365
MaterialGAN	**0.05065**	0.01138	0.03060	0.07790
NeuralMaterial	0.05896	0.01273	0.03301	0.09167
Exp. *Algorithmic*	**0.05440**	0.00909	0.04246	0.07365
Exp. *MaterialGAN*	0.05518	0.01274	0.03060	0.07790
Exp. *NeuralMaterial*	0.06047	0.01671	0.03301	0.09167
Lay *Algorithmic*	0.05308	0.00626	0.04108	0.06239
Lay *MaterialGAN*	**0.04610**	0.00808	0.03696	0.06362
Lay *NeuralMaterial*	0.05745	0.00761	0.04342	0.06599



(a) Distributions of SUS scores for each workflow.



(b) Distributions of task completion times for each workflow.



(c) Distributions of task completion accuracy for each workflow.

Fig. 4: Error plots for SUS score in 4a, task completion times 4b, and LPIPS values for all workflows 4c.

consistency, confidence in use, and learnability. This metric provides a basis for comparing which usability dimensions a particular system ranked lower or higher than expected based on its final SUS score. In the case of experienced users, their ratings for the *Algorithmic* approach were lower than the overall average of 77.25 in terms of perceived ease of use (68), consistency (68), and confidence in use (68), with only perceived learnability (80) and complexity (80) being closer to the average. However, for lay users, their rankings for the *Algorithmic* approach aligned with the SUS score of 82.75 in most areas, except for confidence in use, which was slightly lower (68). For the *MaterialGAN* approach, the most notable discrepancies between lay and experienced users were observed regarding perceived consistency, ease of use, and confidence in use. Lay users consistently ranked the system lower (68) than expected in all three dimensions, whereas experienced users rated it in line (80) with the computed SUS score of 83.5. As for the *NeuralMaterial* workflow, both groups ranked the approach lower (68) than expected in terms of perceived consistency, despite a SUS score of 83.00. Also, lay users rated the system lower in learnability (68) than anticipated. Experienced users ranked *NeuralMaterial* more in line with the computed average SUS score of 86.00 in all of the remaining dimensions of usability. A detailed overview of the calculated thresholds for each statement can be found in the Supplemental material.

4.3 User observation and feedback

We carried out our user study in private online sessions and analyzed participants' comments as they worked on study tasks to gain a deeper understanding of their user experience.

When using the first baseline approach, several users (6) believed that the options for material editing were extremely limited, but some (3) praised its simple and fast editing. Two users also commented that they relied heavily on the provided names of each material when solving the provided task.

While the iterative editing approach in *MaterialGAN* received positive feedback from many users (8), some (5) raised concerns about the ambiguous editing directions. Additionally, three users found *MaterialGAN* to be more precise but also more challenging to use than the baseline approach. A few users (3) also mentioned that they would be more inclined to use this approach if they had more control over the granularity of the interpolation step. Furthermore, a few users (4) expressed dissatisfaction with the generated previews displayed in the sequential gallery, as they believed they didn't effectively aid in envisioning the material's appearance in a new scene. Two participants noted that a faster generation of new interpolations after each editing step would be necessary for practical use in an industrial environment.

Concerning the third approach, several participants (7) preferred the enhanced level of detail provided in the materials. However, several participants (5) preferred the editing options available in the sequential gallery using the *MaterialGAN* workflow. Additionally, several participants (7) positively commented on the inclusion of the ability to resize the material's level of detail according to the resolution, as well as the option to generate many variations of the same material. On the other hand, some participants (5) remarked that the editing process was too slow, hindering an efficient iterative workflow.

During the evaluation phase, participants were actively engaged in providing valuable suggestions aimed at enhancing the workflows and improving the implemented user interface. A prevailing consensus among participants was the desire for reduced processing times between each editing step, as generating new materials in the sequential gallery proved to be quite time-consuming. Furthermore, a commonly shared suggestion involved incorporating the ability to interpolate established PBR parameters, including roughness, albedo, and specularity, within the sequential gallery interface. Additionally, participants expressed a need for finer control and granularity in the scale of interpolation carried out during each editing step within the sequential gallery.

4.4 Time analysis of each workflow

We felt it was important to assess task completion times in our study. However, these correlated strongly with the actual runtime performance of each implemented approach. For this reason, we decided to measure the time for material creation and the time required for a processing step with each of the implemented approaches. We measured performance on the computer system on which the user study was remotely conducted. The system had an AMD EPYC 7v12 CPU, 56 GB RAM, and an Nvidia Tesla T4 GPU with 16 GB of VRAM.

Our measured results are showcased in Table 5, illustrating the average duration of 10 measured iterations for each step in the material workflows. In the case of machine learning-based workflows, we assessed the extraction time for a single input exemplar, employing 1000 epochs of optimization on each integrated model. Additionally, when measuring the duration of resolution upscaling or material reseeding with *NeuralMaterial*, we conducted a comparison targeting a resolution of 1024 × 1024 pixels.

The duration of individual workflow steps is crucial for its usability and was also pointed out by several participants in the user study. This step, however, will be faster with newer hardware and will hopefully positively affect usability.

Table 5: Measurements of average time performance for each step in each of the implemented workflows (*n.a.* stands for not applicable).

Workflow	Image formatting (s)	Material extraction (s)	Material editing (s)	upscaling or reseeding (s)
Algorithmic (Baseline)	*n.a.*	28.013	<1.00	*n.a.*
MaterialGAN	34.706	383.459	10.430	15.584
NeuralMaterial	*n.a.*	392.525	32.073	7.369

5 Discussion

The study results show that modern deep-learning-supported material extraction and editing approaches can be integrated into the existing 3D modeling tools as workflows with high acceptance and usability. Users can achieve the desired results with similar precision as with the existing tools. This study shows that there are slight preference differences between lay and expert users on which approaches they prefer and why.

The study results have confirmed our hypotheses. All the implemented approaches are suitable for achieving the desired tasks achieving 80.00 (A-) SUS score on average for the worst performing workflow (*Algorithmic*), ranking our implementations as an *"above-average system"* (**confirming H1**). The users prefer the deep-learning-based approaches over the baseline, which not only results in their higher SUS scores but also their comments, where they praised the material editing options provided in the sequential gallery and the inclusion of the ability to resize the material's level of detail by the *NeuralMaterial* approach, (**confirming H2**). The users, on average, rated *NeuralMaterial* with the highest SUS score – 86.00. This rejects our hypothesis H3. However, while examining results within lay and expert users, the lay users graded the baseline workflow *Algorithmic* with the highest SUS score – 82.75 (**partially confirming H3**). Additionally, we observed that preference of the workflow differs between lay and expert users, the former favoring the baseline approach (*Algorithmic*), and the latter favoring the deep-learning-based one (*NeuralMaterial*). This finding is something we did not anticipate before conducting the study.

There are still some limitations to our work. Integration of several material extraction/editing approaches into existing tools limited their functionality to a set of features that could be directly compared and allowed integration within comparable workflows. While we took extreme care to make most of their functionalities available, there were still some that could not be implemented. One such limitation concerns the *Algorithmic* approach, where the sequential gallery option could not be implemented due to limitations of the Blender Python API. Another limitation of the add-on was the lack of ability to offer previews in the sequential gallery that would better match the given Blender scene. The differentiable renderer included in both machine learning approaches provides the ability to change lighting settings but cannot account for more complex geometries. Therefore, we had to design the material preview to only show materials rendered on a plane, which some users found problematic when trying to envision the appearance of the material on an object in the user study scene.

The presented study shows that integrating modern deep-learning-based material extraction/editing approaches into the existing 3D modeling workflows is possible and meaningful. With further development of technology, optimization of deep model inference speeds, and explainability of their results, integrating such approaches will be even more meaningful and welcome to ease and accelerate the 3D modeling pipelines.

While previous studies compare the results with other selected studies, they mostly, if at all, only perform user study that explores the usability of the presented approach as a stand-alone tool and compare it with a certain baseline method. To the best of our knowledge, this is the first user study that compares the integration of workflows implementing state-of-the-art deep-learning-based material extraction and editing approaches as an add-on for a popular and broadly used open-source 3D modeling software, Blender, and compares them with a baseline implementation which only uses the functionalities already available in Blender. This presents a significant novelty and demonstrates how a comparison of new material acquisition/editing approaches should be made.

6 Conclusion

In conclusion, this paper presents a comparative evaluation of deep-learning-based generative methods for material extraction and editing in everyday 3D modeling workflows. The research contributions include the development of a user-friendly

Blender add-on that integrates two state-of-the-art generative material models and additional editing options, the development of a baseline approach using Blender functionalities, and a user study for evaluating their usability.

Through a user study involving experts and lay users, the usability and effectiveness of the implemented approaches were assessed using the SUS questionnaire extended with additional questions. The results indicate that all implemented approaches are suitable for achieving the desired tasks, with one of the deep learning-based approaches (*NeuralMaterial*) ranking the best. The findings demonstrate the feasibility and usability of integrating deep learning-based material generation into 3D modeling workflows. The study emphasizes the importance of considering user preferences and expertise levels in designing and implementing these approaches. The results provide valuable insights for future research and development in the field of material generation, paving the way for more user-friendly and efficient workflows in 3D modeling software.

Similar studies should also be performed to evaluate the option of integrating deep-learning approaches in other steps of the 3D modeling pipeline, such as mesh generation from images or NeRFs [35,31], texture generation from text Text2Tex [5] or even model mesh generation from text, such as Text2Mesh [24], Text-2-3D [29], or Get3D [8].

References

1. Aaron Bangor, P.T.K., Miller, J.T.: An empirical evaluation of the system usability scale. International Journal of Human–Computer Interaction **24**(6), 574–594 (2008). https://doi.org/10.1080/10447310802205776, https://doi.org/10.1080/10447310802205776
2. Blažica, B., Lewis, J.R.: A slovene translation of the system usability scale: The sus-si. International Journal of Human–Computer Interaction **31**(2), 112–117 (2015). https://doi.org/10.1080/10447318.2014.986634, https://doi.org/10.1080/10447318.2014.986634
3. Blender Community: Blender 3.5 Reference Manual (2023), https://docs.blender.org/manual/en/latest/
4. Brooke, J.: SUS: A 'Quick and Dirty' Usability Scale, p. 207–212. CRC Press, London, UK (Jun 1996). https://doi.org/10.1201/9781498710411-35, http://dx.doi.org/10.1201/9781498710411-35
5. Chen, D.Z., Siddiqui, Y., Lee, H.Y., Tulyakov, S., Nießner, M.: Text2Tex: Text-driven Texture Synthesis via Diffusion Models (2023)
6. Chen, Y., Liu, S., Wang, X.: Learning Continuous Image Representation with Local Implicit Image Function. 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 8624–8634 (2021). https://doi.org/10.1109/CVPR46437.2021.00852
7. Dupuy, J., Jakob, W.: An Adaptive Parameterization for Efficient Material Acquisition and Rendering. ACM Trans. Graph. **37**(6) (December 2018). https://doi.org/10.1145/3272127.3275059, https://doi.org/10.1145/3272127.3275059
8. Gao, J., Shen, T., Wang, Z., Chen, W., Yin, K., Li, D., Litany, O., Gojcic, Z., Fidler, S.: GET3D: A Generative Model of High Quality 3D Textured Shapes Learned from Images. Advances In Neural Information Processing Systems (2022)

9. Garces, E., Rodriguez-Pardo, C., Casas, D., Lopez-Moreno, J.: A Survey on Intrinsic Images: Delving Deep into Lambert and Beyond. International Journal of Computer Vision **130**(3), 836–868 (March 2022). https://doi.org/10.1007/s11263-021-01563-8
10. Garrido-Jurado, S., Muñoz-Salinas, R., Madrid-Cuevas, F., Marín-Jiménez, M.: Automatic generation and detection of highly reliable fiducial markers under occlusion. Pattern Recognition **47**(6), 2280–2292 (2014). https://doi.org/https://doi.org/10.1016/j.patcog.2014.01.005, https://www.sciencedirect.com/science/article/pii/S0031320314000235
11. Guo, Y., Smith, C., Hašan, M., Sunkavalli, K., Zhao, S.: MaterialGAN: Reflectance Capture Using a Generative SVBRDF Model. ACM Trans. Graph. **39**(6) (November 2020). https://doi.org/10.1145/3414685.3417779
12. Henzler, P., Deschaintre, V., Mitra, N.J., Ritschel, T.: Generative Modelling of BRDF Textures from Flash Images. ACM Trans. Graph. **40**(6) (December 2021). https://doi.org/10.1145/3478513.3480507
13. Hobbs, J.: Why IKEA uses 3D renders vs. photography for their furniture catalog (January 2022), https://www.cadcrowd.com/blog/why-ikea-uses-3d-renders-vs-photography-for-their-furniture-catalog/
14. Hu, Y., Hašan, M., Guerrero, P., Rushmeier, H.E., Deschaintre, V.: Controlling Material Appearance by Examples. Computer Graphics Forum **41**, 117–128 (2022)
15. Hu, Y., He, C., Deschaintre, V., Dorsey, J., Rushmeier, H.: An Inverse Procedural Modeling Pipeline for SVBRDF Maps. ACM Trans. Graph. **41**(2) (January 2022). https://doi.org/10.1145/3502431, https://doi.org/10.1145/3502431
16. Jin, Y., Li, R., Yang, W., Tan, R.T.: Estimating reflectance layer from a single image: Integrating reflectance guidance and shadow/specular aware learning. Proceedings of the AAAI Conference on Artificial Intelligence **37**(1), 1069–1077 (Jun 2023). https://doi.org/10.1609/aaai.v37i1.25188, http://dx.doi.org/10.1609/aaai.v37i1.25188
17. Koyama, Y., Sato, I., Goto, M.: Sequential Gallery for Interactive Visual Design Optimization. ACM Trans. Graph. **39**(4) (August 2020). https://doi.org/10.1145/3386569.3392444, https://doi.org/10.1145/3386569.3392444
18. Lewis, J.R.: The system usability scale: Past, present, and future. International Journal of Human–Computer Interaction **34**(7), 577–590 (2018). https://doi.org/https://doi.org/10.1080/10447318.2018.1455307, https://doi.org/10.1080/10447318.2018.1455307
19. Lewis, J.R., Sauro, J.: Item Benchmarks for the System Usability Scale. J. Usability Studies **13**(3), 158–167 (May 2018). https://doi.org/10.5555/3294033.3294037
20. Martin-Brualla, R., Radwan, N., Sajjadi, M.S.M., Barron, J.T., Dosovitskiy, A., Duckworth, D.: Nerf in the wild: Neural radiance fields for unconstrained photo collections. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 7206–7215 (Jun 2021). https://doi.org/10.1109/cvpr46437.2021.00713, http://dx.doi.org/10.1109/cvpr46437.2021.00713
21. Matusik, W., Pfister, H., Brand, M., McMillan, L.: A data-driven reflectance model. ACM Trans. Graph. **22**(3), 759–769 (jul 2003). https://doi.org/10.1145/882262.882343, https://doi.org/10.1145/882262.882343
22. McAuley, S., Hill, S., Hoffman, N., Gotanda, Y., Smits, B., Burley, B., Martinez, A.: Practical physically-based shading in film and game production. ACM SIGGRAPH 2012 Courses pp. 1–7 (2012). https://doi.org/10.1145/2343483.2343493, https://doi.org/10.1145/2343483.2343493
23. McDermott, W.: The PBR Guide: A Handbook for Physically Based Rendering. Clermont-Ferrand, France: Allegorithmic, 2018, New York, NY, USA (2018)

24. Michel, O., Bar-On, R., Liu, R., Benaim, S., Hanocka, R.: Text2mesh: Text-driven neural stylization for meshes. 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (Jun 2022). https://doi.org/10.1109/cvpr52688.2022.01313, http://dx.doi.org/10.1109/cvpr52688.2022.01313
25. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. Commun. ACM **65**(1), 99–106 (dec 2021). https://doi.org/https://doi.org/10.1145/3503250, https://doi.org/10.1145/3503250
26. NVIDIA: Material Definition Language Handbook (May 2021), http://mdlhandbook.com/mdl%5Fhandbook/index.html
27. Pathik, F.H.: photo-to-pbr-texture-blender (2020), https://github.com/cgvirus/photo-to-pbr-texture-blender
28. Pharr, M., Jakob, W., Humphreys, G.: Physically Based Rendering: From Theory to Implementation. Elsevier Science, Cambridge, MA, USA, 3rd edn. (2017). https://doi.org/10.1016/b978-0-12-800645-0.50029-4
29. Poole, B., Jain, A., Barron, J.T., Mildenhall, B.: DreamFusion: Text-to-3D using 2D Diffusion (2022)
30. Prakash, S., Rainer, G., Bousseau, A., Drettakis, G.: Deep scene-scale material estimation from multi-view indoor captures. Computers & Graphics **109**, 15–29 (2022). https://doi.org/https://doi.org/10.1016/j.cag.2022.09.010, https://www.sciencedirect.com/science/article/pii/S0097849322001789
31. Rakotosaona, M.J., Manhardt, F., Arroyo, D.M., Niemeyer, M., Kundu, A., Tombari, F.: NeRFMeshing: Distilling Neural Radiance Fields into Geometrically-Accurate 3D Meshes (2023)
32. Shen, Y., Zhou, B.: Closed-Form Factorization of Latent Semantics in GANs. 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 1532–1540 (2021). https://doi.org/10.1109/CVPR46437.2021.00158
33. Shimizu, E., Fisher, M., Paris, S., McCann, J., Fatahalian, K.: Design Adjectives: A Framework for Interactive Model-Guided Exploration of Parameterized Design Spaces. Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology p. 261–278 (2020). https://doi.org/10.1145/3379337.3415866, https://doi.org/10.1145/3379337.3415866
34. Smythe, D., Stone, J.: MaterialX: An Open Standard for Network-Based CG Object Looks (March 2021), https://www.materialx.org/assets/MaterialX.v1.38.Spec.pdf
35. Tang, J., Zhou, H., Chen, X., Hu, T., Ding, E., Wang, J., Zeng, G.: Delicate Textured Mesh Recovery from NeRF via Adaptive Surface Refinement (2022)
36. Wang, J., Olson, E.: AprilTag 2: Efficient and robust fiducial detection. 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) pp. 4193–4198 (2016). https://doi.org/10.1109/IROS.2016.7759617
37. Yuanji, W., Jianhua, L., Yi, L., Yao, F., Qinzhong, J.: Image quality evaluation based on image weighted separating block peak signal to noise ratio. Proceedings of the 2003 International Conference on Neural Networks and Signal Processing **2**, 994–997 Vol.2 (2003). https://doi.org/10.1109/ICNNSP.2003.1281036
38. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition pp. 586–595 (2018). https://doi.org/10.1109/CVPR.2018.00068
39. Zhou, X., Hasan, M., Deschaintre, V., Guerrero, P., Sunkavalli, K., Kalantari, N.K.: Tilegen: Tileable, controllable material generation and capture. SIGGRAPH

Asia 2022 Conference Papers pp. 1–9 (2022). https://doi.org/10.1145/3550469.3555403, http://dx.doi.org/10.1145/3550469.3555403
40. Zsolnai-Fehér, K., Wonka, P., Wimmer, M.: Gaussian material synthesis. ACM Transactions on Graphics (TOG) **37**(4), 1–14 (2018)
41. Zsolnai-Fehér, K., Wonka, P., Wimmer, M.: Photorealistic material editing through direct image manipulation. Computer Graphics Forum **39**(4), 107–120 (Jul 2020). https://doi.org/10.1111/cgf.14057, http://dx.doi.org/10.1111/cgf.14057

Comparative Evaluation of Deep-Learning-Based Generative Methods for Material Generation in 3D Modelling Workflow
— *Suplemental Material* —

Nejc Hirci[1], Nejc Lešek[2],
Žiga Lesar[1][0000−0002−9850−0497], and Ciril Bohak[1][0000−0002−9015−2897]

[1] Faculty of Computer and Information Science, University of Ljubljana, Večna pot 113, 1000 Ljubljana, Slovenia
[2] Lightmass Dynamics, Badovinčeva ulica 20, 3270 Laško, Slovenia
ciril.bohak@fri.uni-lj.si

1 Additional related work

1.1 Generative models for material acquisition

Most machine learning methods for material acquisition are designed to predict a Spatially Varying Bidirectional Reflectance Distribution Function (SVBRDF) for an exemplar in a set of images. Because of the ill-posed nature of the problem, due to many different reflectances leading to the same observed image, researchers focused on introducing a strong prior assumption about the space of plausible solutions. Many such solutions focused not on a general or semantically meaningful prior but on using a strong prior learned on synthetic data with additional optimization for each exemplar [1,2]. These methods often proved reasonably successful in extracting material textures from a photo but were poorly applicable to further utilize the learned material latent space for editing. Furthermore, researchers commented on problematic residual artifacts that appear in the predicted maps with sharp and saturated highlights in the input flash photos. Initially, they were also often limited by the resolution of the textures that the model could learn to output, but this was improved in the follow-up work [4].

In contrast to the previous models, the *MaterialGAN* [8] uses Generative Adversarial Network architectures (GAN) model based on the StyleGAN2 [15] and is designed with the latent space optimization in mind. Researchers show their latent space proves to be more robust and semantically meaningful, offering interpolations of materials that could benefit established material editing workflows. The assumption of flat exemplars leads to limitations of the general SVBRDF estimation problem, where flat samples will produce blurring and ghosting effects in the extracted textures. It also limits output textures to 256×256 pixels, but this is also true of the previously described approaches, which is often an accepted trade-off due to the under-constrained nature of the problem. Regardless of its limitations the model offers reliable interpolation of materials, so we choose to assess it in our study. Limitations of low resolution and the problem of seamless materials of *MaterialGAN* were later addressed by Frühstück et al.in the *TileGAN* [3] model, which combines outputs of GANs to synthesize plausible large-scale high-resolution texture maps with no visible boundary artifacts. However, their approach required the model to be trained separately for different material categories (i.e., brick). Further advancements of the tileable seamless texture map generation are presented in [21,24].

A later study [9] managed to greatly progress the SVBRDF extraction from flash photos, presenting an encoder-decoder-based model. For generalization across the material's spatial positions, the decoder adds a random noise image as input, which means the model can generate different variations of the same material. The learned latent space also provides a more consistent color reproduction during interpolation and can output arbitrary-resolution textures. A recent work by Rodriguez-Pardo et al.presented the *UMat* [20], which showed the limitations of the previous model in generating strongly regular patterns (i.e., bathroom tiles). At the time of our conducted user study, the previous model was still the most versatile for material acquisition and generation research while showing significant conceptual differences and editing options to the *MaterialGAN* model, which is why we chose to include it.

A recent noteworthy improvement is presented by Zhou et al. [26] with PhotoMat, a model trained exclusively on real photos of material samples. During training, researchers did not rely on supervision but instead trained a neural material representation model. Because of this, their final model can fit any analytical material model, which is also tileable by using techniques similar to TileGAN. A considerable limitation of all described approaches is that we cannot extract materials from outdoor surfaces lit only by natural lighting. To address this, Martin et al. [19] propose a hybrid method for extracting SVBRDF textures from photos lit only by natural light. They accept the trade-off of poor reproduction of specular material such as metal.

1.2 Procedural material modeling

Procedural material modeling has gained significant interest in most commercial 3D graphics programs, largely due to its flexibility, compactness, and semantically meaningful editing options. For this reason, much of the state-of-the-art research has focused on solving the differentiable rendering [16] problem concerning estimating the procedural material parameters from photographs.

The problem was initially approached as a model optimization, as was shown for fabrics rendering [25,18]. Guo et al. [7] propose a more general solution using a Bayesian framework, precisely defining the posterior distribution of the parameters given the captured data but using only a small subset of authored procedural graphs. While this offers better editability, it requires the user to select an appropriate initial graph. A similar approach was extended with automatic graph selection by Hu et al. [11]. A more extensive approach by Shi et al.uses a differentiable procedural material method MATch [23], built on top of a new model library named DiffMat, which can reproduce the compositing graph system with auto-differentiation support. The approach provides an automatic material selection model that identifies procedural graphs that best represent the input photo from a given set of 88 graphs. Focusing on the limited set of graphs in previous approaches Guerro et al.present MatFormer [6], which produces simpler graphs created from scratch with unconditional material generation and offers light-guided authoring from initially selected nodes.

Several methods showed the benefits of developing a semi-procedural approach, which partially relies on data-driven SVBRDF texture extraction in combination with a simpler generated procedural graph. Guehl et al.present a semi-automatic approach [5] that defines a generic Point Process Texture Basis Function (PPTBF) designed to encapsulate a large span of common spatial stochastic structures. While this approach does not support a complete material extraction from flash photos, it offers many editing options, such as cross-material interpolating structural patterns. Zhou et al. [12] introduce a semi-procedural generation model resembling classical generative methods more closely. It combines procedural properties from input noise and patterns with learnable convolutional filters, making it exceptionally fast and robust for material optimization. Although it lacks fine-grained procedural graph editing, it supports easy adjustments to input noises and guide maps for material properties with minimal fine-tuning.

1.3 Material Editability

Modern material authoring workflows can be improved with machine learning models on the material editability level. Procedural models can inherently offer easier editability as long as the generated graphs can be easily understood. Utilizing the ideas of Guehl et al. [5], Hu et al. [14] present a pipeline for inverse procedural modeling of SVBRDF maps. They add a system for decomposing SVBRDF maps into multiple easily editable sub-materials, which rely on PPTBF masks and show better results than the MATch approach. They still cause some failure cases with semantically meaningful structures (i.e., bathroom tiles).

Zsolnai-Fehér et al.consider editability with a user learning-based system [27] for rapid-scale material synthesis. Learning is implemented using the Gaussian Process Regression (GPR), which leverages scores the user assigns to each material in a gallery user interface. Additionally, users can fine-tune a material with latent space variant generation. A short user study showed big improvements in the time required for both novice and expert users to select and generate 100 materials. Hu et al.present DeepBRDF [10], a deep-learning-based representation of measured Bidirectional Reflectance Distribution Functions (BRDF)s with high recovery. The authors designed another deep neural network to automatically estimate BRDF parameters from an input image. Like the previous approach, DeepBRDF allows exploring a low-dimensional manifold with a smooth transition of the BRDFs defined by its parameters.

The previous methods rely on iterative design steps, but image editing techniques can inspire a more efficient approach to material editing. Zsolnai-Fehér et al.introduce a material editing technique [28] using intuitive image transforms, such as colorization and image in-painting. This method improves editing times for both novice and expert users but shows limitations with strongly localized edits and limited integration with popular SVBRDF extraction methods.

Recently Hu et al. [12] proposed an interesting extension of the unconditional procedural graph generation model MatFormer. It retains unconditional graph generation but allows generation conditioned by text or image prompts. This would be particularly useful for novice users with less experience in visual design by relying on material naming conventions during the generation process.

The tileable version of the *MaterialGAN*, *TileGAN*, was a great inspiration for Hu et al.to adapt the model for additional editing options [13]. Their approach supports transferring the micro and mesoscale texture details from two target materials to the input material. Because of the unconstrained nature of possible edits, the appearance transfer may fail when used on materials with very different parameters (i.e., gray rock and shiny metal).

Editing options developed for our Blender add-on have the most in common with the last by-example editing approach. We wanted to examine different generative material models, so we offered a common user interface for both integrated pipelines as a sequential gallery editing interface. The two previous papers on material editing are either not directly applicable [27] to our approach or too complex for novice users [28].

1.4 Representing material appearance

Building material appearance similarity measures is crucial for efficiently training deep learning models for material properties and defining meaningful and user-friendly editing. Lagunas et al. [17] trained a model for measuring material similarity on a large crowd-sourced experiment to extract human material similarity judgments. This learned feature space correlates strongly with the perceived appearance similarity, which offers many applications such as clustering materials or providing material suggestions for the output of SVBRDF extraction models. This was further analyzed by Serrano et al. [22], who collected a large dataset of perceptual ratings of appearance attributes and not only their direct similarity. The participants had to rate attributes such as glossiness, sharpness of reflections, contrast of reflections, metallicness, lightness, and anisotropy on each image, focusing on the main object. Both papers offer important insights into understanding different components contributing to human understanding of material appearance. Most importantly, they show inherent differences between popular established parameters of Physically Based Rendering (PBR) models and their correlations with our visual understanding of materials.

1.5 User Studies in AI-supported design problems

Assessing the usability of machine-learning-supported tools in practical applications is a complex endeavor, requiring the integration of a neural model that generates high-quality textures at satisfactory image resolutions and a user-friendly graphical interface offering good functionalities. Some of the described work briefly included user study assessments to support their findings, but these assessments often received limited attention, resulting in an incomplete understanding of the applicability of AI-supported design tools. Our investigation emphasizes evaluating deep generative material models in practical workflows to show their most important practical contributions.

2 Additional results for the computed thresholds of the SUS questionnaire

Table 1: Computed thresholds for each statement of the SUS questionnaire.

Statement	Exp. Algorithmic	Exp. MaterialGAN	Exp. NeuralMaterial	Lay Algorithmic	Lay MaterialGAN	Lay NeuralMaterial	Thresholds for an above average system (80)	Thresholds for an average system (68)
I think that I would like to use this system frequently.	3.70	**4.70**	4.50	3.30	3.80	3.90	≥3.80	≥3.39
I found the system unnecessarily complex.	1.80	1.80	1.50	**1.40**	1.80	1.60	≤1.85	≤2.44
I thought the system was easy to use.	4.20	**4.30**	**4.30**	**4.30**	4.20	**4.30**	≥4.24	≥3.67
I think that I would need the support of a technical person to be able to use this system.	1.70	1.80	1.50	**1.10**	1.60	1.30	≤1.51	≤1.85
I found the various functions in this system were well integrated.	3.90	4.50	**4.60**	4.20	3.90	3.80	≥3.96	≥3.55
I thought there was too much inconsistency in this system.	2.00	1.70	1.80	**1.20**	2.00	2.00	≤1.77	≤2.20
I would imagine that most people would learn to use this system very quickly.	4.30	**4.50**	4.40	**4.50**	4.40	4.10	≥4.19	≥3.71
I found the system very cumbersome to use.	2.00	2.30	**1.70**	2.20	2.60	2.30	≤1.66	≤2.25
I felt very confident using the system.	3.80	4.30	**4.40**	4.00	3.90	4.30	≥4.25	≥3.72
I needed to learn a lot of things before I could get going with this system.	1.50	1.30	1.30	1.30	1.50	**1.20**	≤1.65	≤2.09

3 User Study Plan

Exploring the usability of different implemented approaches for creating and editing materials in Blender

Overview

In the following document, We will summarise the plan to conduct the research user study on the usability of different approaches for creating and manipulating physically-based materials in Blender. In it, we want to investigate whether new methods for creating materials based on flash photos ease the established design workflows and discover what advantages and disadvantages they bring to the creation pipeline of new materials.

The methodology

At least 7 users are expected to participate in the study, and the testing will take the form of web conferences with each user individually. The moderator and the user will be connected to a remote computer on which the necessary software will be available. Information will be collected on the testing process, and user satisfaction during use, and users will also provide demographic information, educational level, and previous experience of working with 3D modeling software before starting to solve the user study tasks.

Participants

We will select a small group of participants between 7 and 25. We want to include users with different levels of prior experience working with 3D graphics software. The interface and tasks in the study are designed in a way that does not require in-depth knowledge of using Blender.

Testing procedure

The user study will be conducted in the form of a web conference, where a moderator and a user will connect to a remote computer with a prepared Blender instance with the installed material add-on. The user will be able to consult the moderator in case of confusion about the task or problems with the software.

The moderator will record comments and observe the user's work as the testing progresses.

The testing process will be recorded using screen capture. The user will be made aware that the following recordings will not be made public and will only be used to facilitate analysis of the testing results. Before testing, the user will therefore be asked to sign a document agreeing to the screen recording.

Before starting the tasks, each participant will complete a short questionnaire covering demographic questions and questions on background and previous experience working in 3D modeling software (Blender, Maya, etc.). The task sets will be defined for the individual approach presented in the addon. On the remote computer's desktop, a PDF document will contain the task descriptions for each set and pre-prepared photographs and some materials in separate folders for each approach.

Testing will consist of three sets for each of the implemented approaches, with the user familiarising themselves with each approach through a short 2-minute video before starting to solve the provided task. With the moderator, the user will have 3 minutes to read the instruction for the prepared task. Testing will then begin, where the individual will face one longer task utilizing the provided approach. We will track the task completion times for future analysis. The order of tested approaches will be selected at random.

At the end of each session, the participant will also complete a short questionnaire concerning their observations and satisfaction with using each tested approach. This questionnaire will be a standardized System usability scale questionnaire. After completing all three tasks, a final questionnaire will be administered to the participants on the overall process of testing and using the Blender add-on.

Roles

User

Participates in a user study according to the moderator's instructions. While completing the provided tasks, they try to give feedback through a 'think aloud' approach. After the testing is completed, they are encouraged to express any comments about the usage of a given workflow. They are interested in computer graphics but do not require prior knowledge of 3D computer graphics programs.

Moderator

Before starting, the moderator explains the purpose and objectives of the user study to the participant. Before each set of tasks, they review the instructions with the user and clarify any potential uncertainties. During the testing process, they record the user's work and comments and assist to a reasonable extent in case of major difficulties.

Tasks

The testing will be divided into three sets for each approach (MaterialGAN, Algorithmic, and NeuralMaterial). The user will be tasked with selecting and editing materials in a given Cornell's box scene.

Each of the three sets is expected to cover a task lasting up to 20 minutes. The task instructions will be provided in a desktop file available on the remote computer, where the testing will take place. In the tasks, the user will attempt to fill a scene of 4 objects without materials (default white material): torus, block, sphere, and Suzanne (the signature Blender model of monkey head) with materials as close as possible to the target depiction of the scene. The time taken to complete each task will be recorded. The tasks will have an estimated completion time, but the user will be allowed to exceed it if necessary. The moderator will compare the user's work at the end of the testing by comparing the 3D scene renderings on which the user worked with the depiction of the target scene provided in each set. One of the formal approaches for image comparison (e.g., PSNR, SSIM, LPIPS) will be used to analyze the accuracy of the user's work.

After each set is completed, the user will complete the System Usability Scale (SUS) questionnaire, which provides a simple way to evaluate the usability of user interfaces. It consists of ten statements where the user marks the degree of agreement on a 5-point Likert scale.

After the testing is completed, the participant will be able to share any additional comments on the usage of different workflows and provide possible recommendations to the addon user interface.

Metrics

The following metrics will be used to assess the performance of individual workflows:

- Accuracy of the user’s reproduced scene compared to the target depiction.
- Duration of completing individual tasks.
- User's rating of usability and appropriateness of the interface.

Addressing critical implementation failures

During the problem-solving process, various issues related to the use of the developed add-on for Blender may arise. These issues will be recorded and addressed based on the following factors:

- **Frequency of occurrence of issues:**
    - *High:* more than 30% of users have encountered the issue,
    - *Medium:* 11%-29% of users have encountered the issue,
    - *Low:* 10% or fewer users have encountered the issue.

- **Consequences of critical issues:**
  Based on their impact on task resolution and interface usage:
    - *High*: The user was unable to complete the task due to the problem.
    - *Medium*: The user's work was significantly slowed down by the error.
    - *Low:* Task resolution was not hindered by the issue.

Review of results

The research paper will include a detailed record and analysis of the research results. It will encompass a comprehensive review of the surveys as well as the participants' performance in solving tasks. In addition to the obtained research conclusions, opinions on the advantages and disadvantages of the approaches, as well as recommendations for improving the interface implementation, will be expressed. Furthermore, a part of the reflection will be devoted to new emerging approaches in the field.

4 User Study Instructions

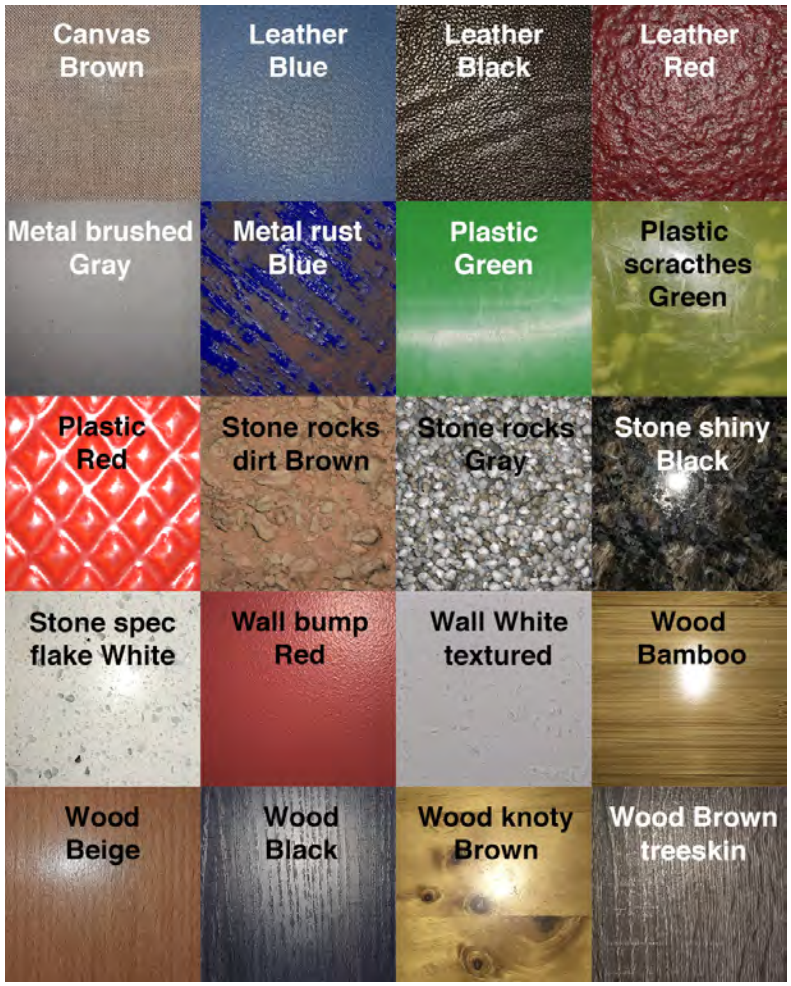
Instructions for User Study of Physically Based Material Systems

Introduction

In our research, we focused on comparing various established approaches for creating and editing physically based rendering (PBR) materials. For the purpose of the research, we also developed an add-on for the Blender program that includes three different approaches for working with PBR materials. Two approaches rely on machine learning and allow users to extract textures required for physically-based rendering from flash photos. The add-on also provides an alternative approach to material editing, where users can choose materials from a gallery with different sets of parameters instead of manually changing textures or adjusting parameters. The third approach is based on a simple algorithm for extracting textures from photographs and interpolating them with certain materials in Blender.

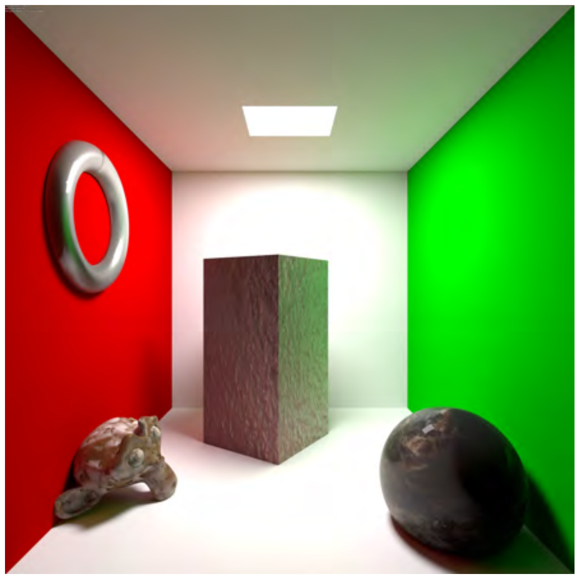
Before starting the tasks, please fill out the first page of the attached questionnaire, which includes some demographic questions and questions related to your prior experience with 3D computer graphics programs.

In the following tasks, you will be using 20 pre-prepared materials. You can see their appearance in the image below, arranged in alphabetical order corresponding to the folders in which they are located on the computer.



Canvas Brown
Leather Blue
Leather Black
Leather Red
Metal brushed Gray
Metal rust Blue
Plastic Green
Plastic scracthes Green
Plastic Red
Stone rocks dirt Brown
Stone rocks Gray
Stone shiny Black
Stone spec flake White
Wall bump Red
Wall White textured
Wood Bamboo
Wood Beige
Wood Black
Wood knoty Brown
Wood Brown treeskin

Algorithmic Workflow



To begin, you will watch a short video demonstrating all the steps for working with the Algorithmic approach. The video is located in the **Algorithmic** folder on the desktop. After watching it, please read the instructions of the task and a summary of workflow steps below, and answer the questions in the survey section titled **Pre-Testing**. Once you have completed the task, please fill out the survey questions in the **Post-Testing** section.

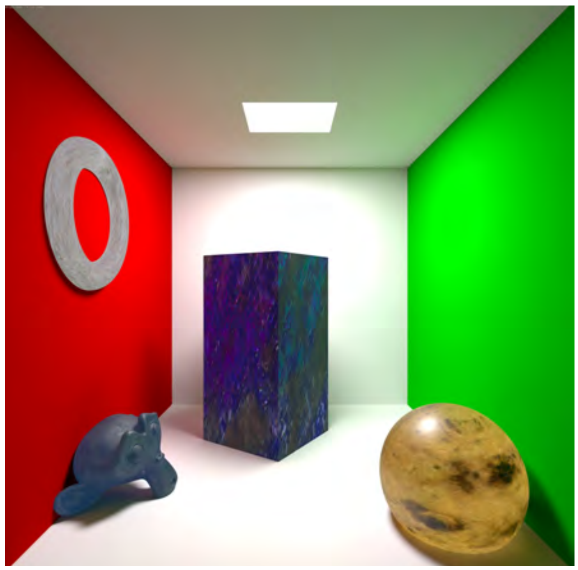
Summary of workflow steps:

1. Select an object in the scene using the camera view on the right side of the screen.
2. Choose the **Algorithmic** generation approach from the **Material System Select** dropdown menu in the addon interface on the left.
3. Open the browser by clicking on the 📁 icon and navigate to the D:\materials\Algorithmic folder.
4. Select one of the pre-prepared materials from the folder by clicking on the preview image in the browser on the **Select folder with data.**
5. Below the **Generate** button, you can choose from 8 different materials to mix with the currently selected material by adjusting the value in the **Mix** value field.

Task Instructions

On the desktop, you will find a folder named **Algorithmic**. Inside it, there is a Blender file with a prepared environment. Open the file to start working. The goal of the task is to use the selected approach to try to recreate the scene shown in the image above by selecting and editing the provided materials. Finally, save the changes in the same Blender file.

MaterialGAN Workflow



To begin, you will watch a short video demonstrating all the steps for working with the Algorithmic approach. The video is located in the **MaterialGAN** folder on the desktop. After watching it, please read the instructions of the task and a summary of workflow steps below, and answer the questions in the survey section titled **Pre-Testing**. Once you have completed the task, please fill out the survey questions in the **Post-Testing** section.

Summary of workflow steps:

1. Select an object in the scene using the camera view on the right side of the screen.
2. Choose the **MaterialGAN + LIIF** generation approach from the **Material System Select** dropdown menu in the addon interface on the left.
3. Open the browser by clicking on the 📁 icon and navigate to the D:\materials\MaterialGAN folder.
4. Select one of the pre-prepared materials from the folder by clicking on the preview image in the browser on the **Select folder with data.**
5. To start editing, click the **Get interpolations** button to generate a gallery of materials. The generation process takes approximately 15 seconds.
6. You can then select one of the 8 materials from the gallery interface to replace the current one by clicking the button just below the desired material. These materials are generated as interpolations in the material space. You can change the desired interpolation direction by clicking the **Toggle direction** button. When selecting a new material, the generation process will again take approximately 15 seconds.

7. After finishing the editing, you are encouraged to increase the texture resolution by entering your desired values in the **Height resolution** and **Width resolution** fields and clicking the **Upscale material** button. Depending on the chosen resolution, the process will take between 15 to 90 seconds. The recommended resolution is already selected by default.

Task Instructions

On the desktop, you will find a folder named **MaterialGAN**. Inside it, there is a Blender file with a prepared environment. Open the file to start working. The goal of the task is to use the selected approach to try to recreate the scene shown in the image above by selecting and editing the provided materials. Finally, save the changes in the same Blender file.

NeuralMaterial Workflow

To begin, you will watch a short video demonstrating all the steps for working with the **NeuralMaterial** approach. The video is located in the **NeuralMaterial** folder on the desktop. After watching it, please read the instructions of the task and a summary of workflow steps below, and answer the questions in the survey section titled **Pre-Testing**. Once you have completed the task, please fill out the survey questions in the **Post-Testing** section.

Summary of workflow steps:

1. Select an object in the scene using the camera view on the right side of the screen.
2. Choose the **NeuralMaterial** generation approach from the **Material System Select** dropdown menu in the addon interface on the left.
3. Open the browser by clicking on the 📁 icon and navigate to the D:\materials\NeuralMaterial folder.
4. To start editing, click the **Get interpolations** button, which will generate a gallery of materials. The generation process takes approximately 30 seconds.
5. You can then select one of the 8 materials from the gallery interface to replace the current one by clicking the button just below the desired material. These materials are generated as interpolations in the material space with 8 manually selected directions. You can change the desired interpolation direction by clicking the **Toggle**

direction button. When selecting a new material, the generation process will again take approximately 30 seconds.

6. During editing, you can also generate variations of the current material at various resolutions by pressing the **Upscale material** button.
7. After finishing the editing, you may also change the texture resolution and detail scale in the material by entering your desired values in the **Height resolution** and **Width resolution** fields and clicking the previously mentioned **Upscale material** button. Depending on the chosen resolution, the process will take between 15 to 90 seconds. The recommended resolution is already selected by default.

Task Instructions

On the desktop, you will find a folder named **NeuralMaterial**. Inside it, there is a Blender file with a prepared environment. Open the file to start working. The goal of the task is to use the selected approach to try to recreate the scene shown in the image above by selecting and editing the provided materials. Finally, save the changes in the same Blender file.

5 Full Questionnaire

Study of Physically Based Material Systems

*Required

1. Terms of agreement*

I have read and agree with the use of screen recording and video conferencing software for the purpose of later analysis of my own interactions with the developed user interface.

Mark only one oval.

- I agree.
- I disagree.

2. Gender *

Mark only one oval.

- male
- female
- nonbinary

3. Age *

Mark only one oval.

- 18-24
- 25-34
- 35-44
- 45-54
- 55-64
- more than 65

4. Level of education achieved *

Mark only one oval.

- trade school
- high school
- professional higher education
- bachelor's degree
- master's degree
- doctoral degree
- Other: ____________

5. Knowledge and experience *

For each statement, indicate your level of agreement on a scale from 1 - 'strongly disagree' to 5 - 'strongly agree,' where 3 represents 'neither agree nor disagree'.

Mark only one oval per row.

	1	2	3	4	5
I have worked with one of the 3D computer graphics programs (Blender, Maya, 3ds, etc.)					
I have used one of the previously mentioned programs for creating and editing materials.					
I am familiar with established physically-based rendering principles in computer graphics.					
I know and understand material parameters such as normal, roughness, specular, and albedo.					
I have previously created textures for materials from images or photographs.					

Pre-Testing: Algortihmic

6. General questions about the prepared approach *

For each statement, please indicate your level of agreement on a scale from 1 - "strongly disagree" to 5 - "strongly agree," where 3 - "neither agree nor disagree." '

Mark only one oval per row.

	1	2	3	4	5
I am familiar with the concept of tasks that I will need to solve.	◯	◯	◯	◯	◯
I had enough time to prepare for the use of the approach.	◯	◯	◯	◯	◯

Post-Testing: Algorithmic

7. SUS Questionnaire *

For each statement, please indicate your level of agreement on a scale from 1 - "strongly disagree" to 5 - "strongly agree," where 3 - "neither agree nor disagree."

Mark only one oval per row.

	1	2	3	4	5
I think that I would like to use this system frequently.					
I found the system unnecessarily complex.					
I thought the system was easy to use.					
I think that I would need the support of a technical person to be able to use this system.					
I found the various functions in this system were well integrated.					
I thought there was too much inconsistency in this system.					
I would imagine that most people would learn to use this system very quickly.					
I found the system very cumbersome to use.					
I felt very confident using the system.					
I needed to learn a lot of things before I could get going with this system.					

Pre-Testing: MaterialGAN

8. General questions about the prepared approach *

For each statement, please indicate your level of agreement on a scale from 1 - "strongly disagree" to 5 - "strongly agree," where 3 - "neither agree nor disagree."

Mark only one oval per row.

	1	2	3	4	5
I am familiar with the concept of tasks that I will need to solve.	◯	◯	◯	◯	◯
I had enough time to prepare for the use of the approach.	◯	◯	◯	◯	◯

Post-Testing: MaterialGAN

9. SUS Questionnaire *

For each statement, please indicate your level of agreement on a scale from 1 - "strongly disagree" to 5 - "strongly agree," where 3 - "neither agree nor disagree."

Mark only one oval per row.

	1	2	3	4	5
I think that I would like to use this system frequently.	◯	◯	◯	◯	◯
I found the system unnecessarily complex.	◯	◯	◯	◯	◯
I thought the system was easy to use.	◯	◯	◯	◯	◯
I think that I would need the support of a technical person to be able to use this system.	◯	◯	◯	◯	◯
I found the various functions in this system were well integrated.	◯	◯	◯	◯	◯
I thought there was too much inconsistency in this system.	◯	◯	◯	◯	◯
I would imagine that most people would learn to use this system very quickly.	◯	◯	◯	◯	◯
I found the system very cumbersome to use.	◯	◯	◯	◯	◯
I felt very confident using the system.	◯	◯	◯	◯	◯
I needed to learn a lot of things before I could get going with this system.	◯	◯	◯	◯	◯

Pre-Testing: NeuralMaterial

10. General questions about the prepared approach *

For each statement, please indicate your level of agreement on a scale from 1 - "strongly disagree" to 5 - "strongly agree," where 3 - "neither agree nor disagree."

Mark only one oval per row.

	1	2	3	4	5
I am familiar with the concept of tasks that I will need to solve.	⬭	⬭	⬭	⬭	⬭
I had enough time to prepare for the use of the approach.	⬭	⬭	⬭	⬭	⬭

Post-Testing: NeuralMaterial

11. SUS Questionnaire *

For each statement, please indicate your level of agreement on a scale from 1 - "strongly disagree" to 5 - "strongly agree," where 3 - "neither agree nor disagree."

Mark only one oval per row.

	1	2	3	4	5
I think that I would like to use this system frequently.	⬭	⬭	⬭	⬭	⬭
I found the system unnecessarily complex.	⬭	⬭	⬭	⬭	⬭
I thought the system was easy to use.	⬭	⬭	⬭	⬭	⬭
I think that I would need the support of a technical person to be able to use this system.	⬭	⬭	⬭	⬭	⬭
I found the various functions in this system were well integrated.	⬭	⬭	⬭	⬭	⬭
I thought there was too much inconsistency in this system.	⬭	⬭	⬭	⬭	⬭
I would imagine that most people would learn to use this system very quickly.	⬭	⬭	⬭	⬭	⬭
I found the system very cumbersome to use.	⬭	⬭	⬭	⬭	⬭
I felt very confident using the system.	⬭	⬭	⬭	⬭	⬭
I needed to learn a lot of things before I could get going with this system.	⬭	⬭	⬭	⬭	⬭

Additional feedback

The testing is over, but you can now provide additional feedback and recommendations.

12. Comments and recommendations

If you encountered any ideas during the testing on how the implemented approaches could be adjusted for even greater ease of use, you can write your notes and comments in the field below.

References

1. Deschaintre, V., Aittala, M., Durand, F., Drettakis, G., Bousseau, A.: Single-Image SVBRDF Capture with a Rendering-Aware Deep Network. ACM Trans. Graph. **37**(4) (July 2018). https://doi.org/10.1145/3197517.3201378, https://doi.org/10.1145/3197517.3201378
2. Deschaintre, V., Aittala, M., Durand, F., Drettakis, G., Bousseau, A.: Flexible svbrdf capture with a multi-image deep network. Computer Graphics Forum **38**(4), 1–13 (2019). https://doi.org/https://doi.org/10.1111/cgf.13765, https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.13765
3. Frühstück, A., Alhashim, I., Wonka, P.: TileGAN: Synthesis of Large-Scale Non-Homogeneous Textures. ACM Trans. Graph. **38**(4) (July 2019). https://doi.org/10.1145/3306346.3322993, https://doi.org/10.1145/3306346.3322993
4. GAO, D., Li, X., Dong, Y., Peers, P., Xu, K., Tong, X.: Deep inverse rendering for high-resolution svbrdf estimation from an arbitrary number of images. ACM Trans. Graph. **38**(4) (jul 2019). https://doi.org/10.1145/3306346.3323042, https://doi.org/10.1145/3306346.3323042
5. Guehl, P., Allègre, R., Dischler, J., Benes, B., Galin, E.: Semi-procedural textures using point process texture basis functions. Computer Graphics Forum **39**(4), 159–171 (Jul 2020). https://doi.org/10.1111/cgf.14061, http://dx.doi.org/10.1111/cgf.14061
6. Guerrero, P., Hašan, M., Sunkavalli, K., Měch, R., Boubekeur, T., Mitra, N.J.: Matformer: A generative model for procedural materials. ACM Trans. Graph. **41**(4) (jul 2022). https://doi.org/10.1145/3528223.3530173, https://doi.org/10.1145/3528223.3530173
7. Guo, Y., Hašan, M., Yan, L., Zhao, S.: A bayesian inference framework for procedural material parameter estimation. Computer Graphics Forum **39**(7), 255–266 (Oct 2020). https://doi.org/10.1111/cgf.14142, http://dx.doi.org/10.1111/cgf.14142
8. Guo, Y., Smith, C., Hašan, M., Sunkavalli, K., Zhao, S.: MaterialGAN: Reflectance Capture Using a Generative SVBRDF Model. ACM Trans. Graph. **39**(6) (November 2020). https://doi.org/10.1145/3414685.3417779
9. Henzler, P., Deschaintre, V., Mitra, N.J., Ritschel, T.: Generative Modelling of BRDF Textures from Flash Images. ACM Trans. Graph. **40**(6) (December 2021). https://doi.org/10.1145/3478513.3480507
10. Hu, B., Guo, J., Chen, Y., Li, M., Guo, Y.: DeepBRDF: A Deep Representation for Manipulating Measured BRDF. Computer Graphics Forum **39**(2), 157–166 (2020). https://doi.org/https://doi.org/10.1111/cgf.13920, https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.13920
11. Hu, Y., Dorsey, J., Rushmeier, H.: A novel framework for inverse procedural texture modeling. ACM Trans. Graph. **38**(6) (nov 2019). https://doi.org/10.1145/3355089.3356516, https://doi.org/10.1145/3355089.3356516
12. Hu, Y., Guerrero, P., Hasan, M., Rushmeier, H., Deschaintre, V.: Generating procedural materials from text or image prompts. ACM SIGGRAPH 2023 Conference Proceedings (2023). https://doi.org/10.1145/3588432.3591520, https://doi.org/10.1145/3588432.3591520
13. Hu, Y., Hašan, M., Guerrero, P., Rushmeier, H.E., Deschaintre, V.: Controlling Material Appearance by Examples. Computer Graphics Forum **41**, 117–128 (2022)
14. Hu, Y., He, C., Deschaintre, V., Dorsey, J., Rushmeier, H.: An Inverse Procedural Modeling Pipeline for SVBRDF Maps. ACM Trans. Graph. **41**(2) (January 2022). https://doi.org/10.1145/3502431, https://doi.org/10.1145/3502431

15. Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., Aila, T.: Analyzing and improving the image quality of stylegan. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (Jun 2020). https://doi.org/10.1109/cvpr42600.2020.00813, http://dx.doi.org/10.1109/cvpr42600.2020.00813
16. Kato, H., Beker, D., Morariu, M., Ando, T., Matsuoka, T., Kehl, W., Gaidon, A.: Differentiable rendering: A survey (2020)
17. Lagunas, M., Malpica, S., Serrano, A., Garces, E., Gutierrez, D., Masia, B.: A Similarity Measure for Material Appearance. ACM Trans. Graph. **38**(4) (July 2019). https://doi.org/10.1145/3306346.3323036, https://doi.org/10.1145/3306346.3323036
18. Leaf, J., Wu, R., Schweickart, E., James, D.L., Marschner, S.: Interactive design of periodic yarn-level cloth patterns. ACM Trans. Graph. **37**(6) (dec 2018). https://doi.org/10.1145/3272127.3275105, https://doi.org/10.1145/3272127.3275105
19. Martin, R., Roullier, A., Rouffet, R., Kaiser, A., Boubekeur, T.: MaterIA: Single Image High-Resolution Material Capture in the Wild. Computer Graphics Forum **41**(2), 163–177 (2022). https://doi.org/10.1111/cgf.14466
20. Rodriguez-Pardo, C., Dominguez-Elvira, H., Pascual-Hernandez, D., Garces, E.: Umat: Uncertainty-aware single image high resolution material capture. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5764–5774 (2023). https://doi.org/10.1109/CVPR52729.2023.00558
21. Rodriguez-Pardo, C., Garces, E.: SeamlessGAN: Self-Supervised Synthesis of Tileable Texture Maps. IEEE Transactions on Visualization and Computer Graphics **29**(6), 2914–2925 (2023). https://doi.org/10.1109/TVCG.2022.3143615
22. Serrano, A., Chen, B., Wang, C., Piovarči, M., Seidel, H.P., Didyk, P., Myszkowski, K.: The Effect of Shape and Illumination on Material Perception: Model and Applications. ACM Trans. Graph. **40**(4) (July 2021). https://doi.org/10.1145/3450626.3459813, https://doi.org/10.1145/3450626.3459813
23. Shi, L., Li, B., Hašan, M., Sunkavalli, K., Boubekeur, T., Mech, R., Matusik, W.: Match: Differentiable material graphs for procedural material capture. ACM Trans. Graph. **39**(6) (nov 2020). https://doi.org/10.1145/3414685.3417781, https://doi.org/10.1145/3414685.3417781
24. Toulatzis, V., Fudos, I.: Deep tiling: Texture tile synthesis using a constant space deep learning approach. Advances in Visual Computing pp. 414–426 (2021). https://doi.org/10.1007/978-3-030-90439-5_33, http://dx.doi.org/10.1007/978-3-030-90439-5_33
25. Zhao, S., Jakob, W., Marschner, S., Bala, K.: Building volumetric appearance models of fabric using micro ct imaging (2011). https://doi.org/10.1145/1964921.1964939, https://doi.org/10.1145/1964921.1964939
26. Zhou, X., Hasan, M., Deschaintre, V., Guerrero, P., Hold-Geoffroy, Y., Sunkavalli, K., Kalantari, N.K.: Photomat: A material generator learned from single flash photos. ACM SIGGRAPH 2023 Conference Proceedings pp. 1–11 (2023). https://doi.org/10.1145/3588432.3591535, https://doi.org/10.1145/3588432.3591535
27. Zsolnai-Fehér, K., Wonka, P., Wimmer, M.: Gaussian material synthesis. ACM Transactions on Graphics (TOG) **37**(4), 1–14 (2018)
28. Zsolnai-Fehér, K., Wonka, P., Wimmer, M.: Photorealistic material editing through direct image manipulation. Computer Graphics Forum **39**(4), 107–120 (Jul 2020). https://doi.org/10.1111/cgf.14057, http://dx.doi.org/10.1111/cgf.14057