Alchemist: Parametric Control of Material Properties with Diffusion Models

Prafull Sharma*,1,2 Varun Jampani†,2 Yuanzhen Li2 Xuhui Jia2
Dmitry Lagun2 Fredo Durand2 Bill Freeman1,2 Mark Matthews2
1MIT CSAIL 2Google Research

www.prafullsharma.net/alchemist

Figure 1. Overview. Our method, Alchemist, edits material properties of objects in input images by relative attribute strength $s$. Top: We set the strength $s = 1$, resulting in a beetle without specular highlights, a dark metallic dinosaur, and boot with gray albedo. Our model generates plausible transparency including the light tint, caustics, and hallucinated plausible details behind the object. Bottom: We demonstrate smooth edits for linearly chosen strength values.

Abstract

We propose a method to control material attributes of objects like roughness, metallic, albedo, and transparency in real images. Our method capitalizes on the generative prior of text-to-image models known for photorealism, employing a scalar value and instructions to alter low-level material properties. Addressing the lack of datasets with controlled material attributes, we generated an object-centric synthetic dataset with physically-based materials. Fine-tuning a modified pre-trained text-to-image model on this synthetic dataset enables us to edit material properties in real-world images while preserving all other attributes. We show the potential application of our model to material edited NeRFs.

1. Introduction

Achieving fine-grained control over material properties of objects in images is a complex task with wide commercial applications beyond computer graphics. This ability is particularly relevant in image editing, advertising, and image forensics. We propose a method for precise editing of material properties in images, harnessing the photorealistic generative prior of text-to-image models. We specifically
target four key material properties: roughness, metallic, albedo, and transparency. Our results illustrate that generative text-to-image models contain a strong understanding of light transport which can be leveraged for precise control of these material properties. The physics of light transport affects the appearance of the object. How we view the objects is an interplay of physical factors such as surface geometry, illumination sources, camera intrinsics, color science, sensor linearity, and tone-mapping. However, the most significant of these factors is material properties.

In computer graphics, Bidirectional Reflectance Distribution Functions (BRDFs) [11, 25, 26] define material properties which led to the development of principled and physically based BRDF models [3]. Prior methods typically employed an inverse rendering approach to disentangle and estimate complex scene attributes like geometry and illumination for material modification [36]. Recent work by Subias et al. proposed a GAN-based method trained on synthetic data for perceptual material edits, focusing on metallic and roughness parameters, necessitating the masking of the targeted real-world object [31]. Our approach uses the generative prior of text-to-image models. We directly modify real-world images in pixel space, eliminating the need for auxiliary information such as explicit 3D geometry or depth maps, environment maps, and material annotations, thereby bypassing the process of accurately estimating object and scene-level properties.

Manipulating material properties in pixel space using a pre-trained text-to-image model presents two main challenges. First, the scarcity of real-world datasets with precisely labeled material properties makes generalizing from supervised training difficult. Second, text-to-image models are trained with textual descriptions like "gold," "wood," or "plastic," which often lack fine-grained details about the material. This issue is compounded by the inherent disconnect between the discrete nature of words and the continuous nature of material parameters.

To overcome the first challenge, we render a synthetic dataset featuring physically-based materials and environment maps, thus addressing the need for fine-grained annotations of material properties. For the second challenge, we employ extra input channels to an off-the-shelf diffusion model, refining this model with an instruction-based process inspired by InstructPix2Pix [3]. Despite being trained on only 500 synthetic scenes comprising 100 unique 3D objects, our model effectively generalizes the control of material properties to real input images, offering a solution to the issue of continuous control.

To summarize, we present a method that utilizes a pre-trained text-to-image model to manipulate fine-grained material properties in images. Our approach offers an alternative to traditional rendering pipelines, eliminating the need for detailed auxiliary information. The key contributions of our method are as follows:

1. We introduce an image-to-image diffusion model for parametric control of low-level material properties, demonstrating smooth edits of roughness, metallic, albedo, and transparency.
2. We render a synthetic dataset of fine-grained material edits using 100 3D objects and randomized environment maps, cameras, and base materials.
3. Our proposed model generalizes to real images despite being trained on synthetic data.

2. Related Work

Diffusion models for image generation. Denoising Diffusion Probabalistic Models (DDPMs) have been an active focus of the research community [12, 14, 28–30, 33, 34, 79] for their excellent photorealistic image generation capabilities from text prompts [56, 65, 67, 71]. Image-to-image tasks are possible by modifying the denoising network to accept image inputs, allowing style-transfer [78], inpainting, uncropping, super-resolution, and JPEG compression [70]. Furthermore, the generative priors of 2D diffusion models have been utilized towards novel-view synthesis, 3D generation, and stylistic 3D editing [6, 8, 19, 23, 31, 43, 45, 62, 64, 72, 73, 76, 83, 87, 91, 93, 97]. Our image-to-image method leverages and further controls this learned prior of DDPMs.

Control in generative models. Controlling generative model output remains an active area of study with many works proposing text-based methods [1, 3, 5, 10, 21, 27, 35, 37, 44, 53, 61, 82, 84, 86]. Other works proposed alternative control inputs such as depth maps, sketches [85, 92], paint strokes [51], identity [48, 90], or photo collections [41, 68, 69, 75]. Prompt-to-Prompt [27], P+ [86], and Null-text inversion [53] present editing techniques based on reweighting of cross-attention maps. ControlNet [94] and T2I-Adapter [55] demonstrate control through spatial inputs defining mid-level information. Generated images from diffusion models can also incorporate new subjects from an image collection using a small number of exemplars [7, 20, 41, 68, 69, 75, 89]. While these works control high and mid-level information about objects, control of low-level properties such as materials remains a challenge for them, leading us to our present line of study.

Material understanding and editing. Editing materials in images is a significant challenge, requiring a strong understanding of image formation. Human vision research has extensively explored how attributes like albedo, roughness, illumination, and geometry affect object perception [16, 17, 18, 50, 54, 57–59, 80].

Image-based material editing was introduced by Khan et al. presenting simple material operations using depth estimates [36]. Subsequent works demonstrated disentangle-
3. Method

There is no existing object-centric dataset that precisely varies only single material attributes. Curating such a real-world dataset would be infeasible due to the difficulty of creating physical objects in this way with known parameters. Therefore, we opt to render a synthetic dataset, giving us full control of material attributes. Using this data, we propose a method to perform material attribute control given a context image, instruction, and a scalar value defining the desired relative attribute change. The method is based on latent diffusion model for text-to-image generation with modification that allows us to condition the network on the relative attribute strength.

3.1. Datasets

We render our dataset with the Cycles renderer from Blender [9], using publicly available 3D assets, physically-based materials, and environment maps. Each scene begins with one of 100 unique object meshes from polyhaven.com. Each of these is paired with five randomly chosen materials of the 1200 available from ambientcg.com, and illuminated with one of 400 environment maps. The material is a Principled BRDF shader node, the base shader in Blender. The base configuration of the material is kept as a control defined as 0 strength change for each of the attributes. This control serves as the context input image to the method against which relative changes in roughness, metallic, albedo, and transparency are applied, sampling 10 random relative values for each attribute, the details of which are described below. Finally, we render 15 images of each setup using different camera viewpoints and intrinsics. Samples from the rendered dataset are presented in Figure 3.

Roughness and Metallic. For both roughness and metallic properties, we operate in an additive framework. In the case when the material has the associated map for roughness or metallic, we use an additive node yielding a parametric control between [-1, 1]. For materials where either of these spatial maps are missing, we control the attribute...
control directly as a constant map, assuming the base 0.5 as the control state of the attribute. Note that these values are clamped between [0, 1] so in some cases, further increasing or decreasing the roughness does not result in any change in the rendered image. We account for this by under-sampling such images where the gradient of change is constant.

Reducing the roughness value results in a surface that reflects light more uniformly and sharply, giving it a glossy or shiny appearance. On the other hand, increasing the roughness value leads to a more diffused light reflection, making the surface appear matte or dull. Low metallic value results in the surface absorbing more of the incoming light, as in the case of plastic and wood. Increasing the metallic value leads to a more diffused light reflection, making the surface appear glossy or shiny. On the other hand, increasing the roughness value results in light being reflected more uniformly and sharply, giving it a more matte or dull appearance. We can achieve a mirror-like or a shading-only image when combined with roughness and metallic parameters to produce an effect of the albedo resulting in a fine tint on the object.

We introduce the ability to control transparency. This can be considered as detexturing the albedo and can be interesting when combined with roughness and metallic parameters to achieve a mirror-like or a shading-only image.

Albedo. We implement a color mixing between the original albedo map of the object and a spatially constant gray (RGB = 0.5) albedo map. The parametric controller operates between 0 and 1, where 0 corresponds to the original albedo, and 1 corresponds to completely gray albedo. This can be considered as detexturing the albedo and can be interesting when combined with roughness and metallic parameters to achieve a mirror-like or a shading-only image.

Transparency. We introduce the ability to control transparency by controlling the transmission value in the BSDF shader node. The attribute value is chosen to be in range [0, 1]. For a chosen transmission value t, we choose to reduce the roughness and metallic component in an additive manner by t, and also add a white overlay to the albedo to increase the intensity of the appeared color. For the value of 0, we expect the same opaque object and at 1, we would get a transparent version of the object, making it appear as if it was made of glass. Note that we made the choice to retain the effect of the albedo resulting in a fine tint on the object.

3.2. Parametric Control in Diffusion Models

The rendered synthetic data is used to finetune an image-to-image diffusion model conditioned on relative attribute strength and a generic texture instruction providing parametric control over material properties. We operate in latent space using Stable Diffusion 1.5, a widely adopted text-to-image latent diffusion model.

Diffusion models perform sequential denoising on noisy input samples, directing them towards the dataset distribution by maximizing a score function [77]. A noise process is defined over timesteps t ∈ T, resulting in a normal distribution at T. We operate in latent space by using a pretrained variational encoder E and decoder D [39], a potent aid to conditional image generation [67]. Training draws an image sample I from the dataset, encodes it into a latent z = E(I), then noises it at t as zt. A denoising network εθ predicts the added noise given the latent zt, diffusion time t, and conditioning variables.

Our image-to-image model is conditioned on an input image to be edited, provided as E(Ic) concatenated to the latent being denoised zt. Text conditioning is provided via cross-attention layers using a generic prompt, p = “Change the <attribute_name> of the <object_class>.” Since textual CLIP embeddings [63] do not encode fine-grained information well [60], prompt-only conditioning of s expressed textually (i.e. “Change the roughness of the apple by 0.57.”) yields inconsistent output. To facilitate relative attribute strength conditioning, we also concatenate a constant scalar grid of edit strength s.

We initialize the weights of our denoising network with the pre-trained checkpoint of InstructPix2Pix [3], providing an image editing prior and understanding of instructive prompts. During training (Fig. 2), we minimize the loss:

$$\mathcal{L} = \mathbb{E}_{E(I), E(Ic), s, p \sim \mathcal{N}(0, 1), t} \left[ \| \epsilon - \epsilon_\theta(z_t, t, E(I_c), s, p) \|_2^2 \right]$$

We always provide the s = 0 image as context Ic, and draw an edited image I at random for noising. Since we always render an s = 0 sample, and other s are chosen with stratified sampling, our distribution has a slight bias towards zero. Since many edit strengths may have little effect (i.e. we cannot lower the roughness of an object with 0 roughness), we find that providing too many of these examples biases the network towards inaction. We therefore down-weight such null examples, defined as \(\|I_c - I_e\|^2 < \tau\), by \(w_{null}\) via rejection sampling. In practice we set \(w_{null} = 0.80\), \(\tau = 0.05\). We train with fp16 precision for 10k steps using Adam [38] and learning rate of 5e-5. We use the text encoders and noise schedule from Stable Diffusion. We trained the model on a machine with 2 x A100 for 10k steps.

At test time we provide a held out image as context Ic, edit strength s, and prompt p for the object class of the input image. We denoise for 20 steps using the DPM-solver++ based noise scheduler [47].

Multi-attribute editing. We edit multiple attributes in a single diffusion pass by concatenating more than one edit strength, drawn from \(\{s_a, s_r, s_m\}\) giving us \[z_t[E(I_c)]|s\] as the final UNet input, where | is concatenation.

Classifier-free guidance. Ho et al. [30] proposed classifier-free guidance (CFG) to improve visual quality and faithfulness of images generated by diffusion models. We retain the same CFG setup as InstructPix2Pix for both image and prompt conditioning. We do not however impose CFG with respect to the relative attribute strengths s. We want the network to be faithful to edit strength and forced to reason about incoming material attributes. As s can be 0 by definition of the problem itself, and downweighted as described above, we did not find CFG on s necessary.

We will release the dataset generation pipeline, image renderings with metadata, and the training code.

24133
4. Results

We present qualitative analysis demonstrating the generalization capability of our model to real images despite being trained on synthetic data. Comparisons to baselines show the effectiveness of our model for fine-grained material editing, further supported by a user study. We extend the use of our model to NeRF material editing on the DTU dataset [32, 52].

4.1. Results on Real Images

We demonstrate the effectiveness of our technique through editing material attributes, one at a time, for real unseen images, in Figure 4. For each of the material attributes we use a separate model trained only on that attribute. We observe that the model outputs preserve geometry and take the global illumination into account.

**Roughness.** As the roughness is increased, the output shows removal of the specular highlights replaced by estimate of the base albedo. The highlights are amplified when the roughness is reduced as shown in the case of the pumpkin and statue.

**Metallic.** The increase in the metallic component of the car and pumpkin results in dampening of the base albedo and increase in the shine on the surface. The effect is reverse for the cat statue when the metallic strength was reduced. Our method shows similar behavior to the Principled BRDF shaders, which present perceptually subtle effects when tuning the metallic value.

**Albedo.** As the relative strength for the albedo is turned to 1, we observe the albedo of the Buddha statue, fish, and toy go to gray. This is not a trivial in-image desaturation operation as the network maintains the highlights, shadows, and the light tint from the plausible environment map.

**Transparency.** The transparent renditions of the binocular and the chair demonstrate the prior over 3D geometry of the objects, using which it generates the appropriately tinted glass-like appearance and in-paints background objects. With the edit of the banana, we can see the caustics underneath and the preservation of the specular highlights.

4.2. Baseline Comparisons

We compare our method, Alchemist, to the GAN-based in-image material editing of Subias et al. [81], Prompt-to-Prompt [27] with Null-text Inversion (NTI) [53], and InstructPix2Pix [3] in Figure 5. Furthermore, we fine-tuned the InstructPix2Pix prompt-based approach with our synthetic dataset. Subias et al.’s method results in exaggerated material changes as their objective is perceptual, not physically-based material edits. Null-text inversion and InstructPix2Pix change the global image information instead of only the object of interest: lighting changes for roughness and albedo edits, or a geometry change for metallicity edit. When InstructPix2Pix is trained on our dataset with a prompt-only approach, we observe the model exaggerating the intended effect, yielding artifacts on the panda for metallic and the water for the transparency case. The model also changes the albedo of the sand when only asked to make the change to the crab. Our method faithfully edits only the object of interest, introducing the specular highlights on the
Figure 5. **Qualitative comparison.** Comparison of Alchemist with baseline methods. We increase each of the attributes shown on the left.

<table>
<thead>
<tr>
<th></th>
<th>InstructPix2Pix w/ our data</th>
<th>Our Method</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PSNR↑</td>
<td>SSIM↑</td>
</tr>
<tr>
<td>Roughness</td>
<td>30.9</td>
<td>0.89</td>
</tr>
<tr>
<td>Metallic</td>
<td>31.0</td>
<td>0.89</td>
</tr>
<tr>
<td>Albedo</td>
<td>26.9</td>
<td>0.88</td>
</tr>
<tr>
<td>Transparency</td>
<td>26.9</td>
<td>0.85</td>
</tr>
</tbody>
</table>

Table 1. **Quantitative analysis.** Metrics for the prompt-only InstructPix2Pix trained on our data and our proposed method computing the PSNR, SSIM [88], and LPIPS [95] on a held-out unseen synthetic rendered dataset of 10 scenes with 15 cameras.

User study. To further the comparison between the baseline and our method, we conducted a user study presenting randomly selected N=14 users with pairs of edited images. For each image pair, the users were asked to choose between the two based on: (1) **Photorealism**, and (2) “Which edit do you prefer?”. For both questions, the users were presented with the instruction, i.e. for transparency, the instruction was stated as “the method should be able to output the same object as transparent retaining the tint of the object”. Each user was presented with a total of 12 image pairs (3 image results for each of the 4 attributes).

Our method was chosen as the one with more photorealistic edits (69.6% vs. 30.4%) and was strongly preferred overall (70.2% vs. 29.8%). This is likely due to the apparent exaggeration exhibited by InstructPix2Pix trained on our data with prompt-only approach, leading to saturated effects making it less photo-realistic.

**Smoothness in Parametric Control.**

We demonstrate that our model achieves fine grained
AlbedoRoughness
-1 +1

TransparencyMetallic
-1 +1 +10

Figure 6. **Slider results.** Alchemist produces edits smoothly with attribute strength. We note that the outputs for linear change in the input relative strength in InstructPix2Pix prompt-only trained on our data results in non-smooth transitions. Refer to the supplement videos.

"Change the material of the cat."
albedo: 2
roughness: -1
metallic: 1

"Change the transparency of the cup."
transparency: 1

Figure 7. **Spatial Localization.** Edit results (bottom) when the scalar strength input is masked by the shown segmentation (top). The image is only altered in the segmented region, becoming either shiny (cat), or semi-transparent (cup).

4.3. Specializations

**Spatial localization.** Attending to a specific instance of a class when multiple objects of the same class are present in the image is a difficult task using just language instruction. We explore the possibility of changing the material attribute of a specific instance of the class by only attributing the scalar value in the segmented region, assuming a known segmentation map from an instance segmentation method such as Segment Anything [40]. Though the network was not trained for this specific task, we find that the network does respect the localization of the relative attribute strength, though requires over-driving to values beyond 1. We observe that mask-based editing works in such cases, i.e. changing the material properties of specific instance of cat and cup, as shown in Figure 7.

**Multi-attribute changes.** To enable control over multiple attributes in a single diffusion pass, we train our network on two versions of the dataset to vary albedo \(s_a\), roughness \(s_r\), and metallic \(s_m\). In the *axis-only sampled* version, we keep the context image at the baseline, and vary a single attribute at a time, such that only one of \(\{s_a, s_r, s_m\}\) is non-zero for any given training target image. In the *volume sampled* version, \(\{s_a, s_r, s_m\}\) are all allowed to be non-zero, effectively sampling the 3D volume of material attributes. In both cases, we keep the same number of input images.

We present the qualitative analysis of the joint control
Figure 8. **Multi-Attribute Editing.** Comparison between an “axis-only sampled” model trained on images where only one of \(\{s_a, s_r, s_m\}\) is \(\neq 0\), vs. a “volume sampled” one where all \(\{s_a, s_r, s_m\}\) may be \(\neq 0\). We show edits with \((s_a, s_r, s_m) = (1, -1, 1)\). The former tends to edit only a single attribute, while the latter successfully achieves the desired “silver” appearance.

in Figure 8. We find that the axis-only sampled training fails to compose the three attributes, generally showing bias towards one of the attributes. The model trained on the volume of these attributes successfully generalizes, demonstrating ability to edit multiple attributes at once. We find this essential to producing a strong metallic appearance on objects, as the Principled BSDF shader requires a white, non-rough, and highly metallic surface to produce this look.

**Material editing of NeRFs.** We test the efficacy of per-frame editing using our method for two-step material control in neural radiance field (NeRF) reconstruction. We use a selection of scenes from the DTU MVS [32] dataset and edit them to have reduced albedo or higher specular reflections. We train a NeRF with the vanilla configuration based on [66] (complete details in the supplement).

In the results presented in Figure 9, we observe highly plausible renderings from held-out views showing 3D structure with the intended albedo, roughness, and metallic change. Please refer to the supplement for rendered video of NeRFs trained on edited data.

5. **Discussion**

Our model generalizes to editing fine-grained material properties in real images, despite being trained solely on synthetic data. We believe that our method could extend to a wide range of material alterations achievable with a shader. However, our approach does have limitations, such as producing minimal perceptual changes for roughness and metallic attributes, and occasionally yielding physically unrealistic transparency, as illustrated in Figure 10. The model lacks a complete 3D world model and is unable to maintain physical consistency as seen in the candy-cane example. As is typical with generative models, our method generates plausible interpretations that are true to the given instructions, but it does not necessarily replicate the exact outcomes of a traditional graphics renderer.

6. **Conclusion**

We present a method that allows precise in-image control of material properties, utilizing the advanced generative capabilities of text-to-image models. Our approach shows that even though the model is trained on synthetic data, it effectively edits real images, achieving seamless transitions as the relative strength of the desired attribute is varied. Beyond image editing, we demonstrate the applicability to NeRF allowing for editable materials in NeRFs. We believe that our work can further impact downstream applications and allow for improved control over low-level properties of objects.

7. **Acknowledgements**

This work was supported by NSF 2105819 and gifts from Google and Amazon. We would like to thank Forrester Cole, Charles Herrmann, Junhwa Hur, and Nataniel Ruiz for helpful discussions. Thanks to Shriya Kumar and Parimarjan Negi for proofreading the submission.
References


