This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

PixHt-Lab: Pixel Height Based Light Effect Generation for Image Compositing

Yichen Sheng
Purdue University

Jianming Zhang
Adobe Inc.

Julien Philip
Adobe Inc.

Yannick Hold-Geoffroy
Adobe Inc.

Xin Sun
Adobe Inc.

He Zhang
Adobe Inc.

Lu Ling
Purdue University

Bedrich Benes
Purdue University

(a) cutout (b) SSN [48] (c) SSG [47] (d) PixHt-Lab (ours)

Figure 1. **PixHt-Lab** renders realistic reflection and soft shadows on general shadow receivers for 2D cutouts. (a) shows the object cutout and composition background. (b) SSN [48] cannot render soft shadows on walls due to its ground plane assumption. (c) SSG [47] renders specular reflection, but the shadow on the wall is uniformly softened. (d) Our PixHt-Lab renders realistic soft shadows on the wall guided by 3D-aware buffer channels. The shadow is softened according to the background geometry with more realistic details. PixHt-Lab also renders realistic reflections with physically-based surface materials.

Abstract

Lighting effects such as shadows or reflections are key in making synthetic images realistic and visually appealing. To generate such effects, traditional computer graphics uses a physically-based renderer along with 3D geometry. To compensate for the lack of geometry in 2D Image compositing, recent deep learning-based approaches introduced a pixel height representation to generate soft shadows and reflections. However, the lack of geometry limits the quality of the generated soft shadows and constrains reflections to pure specular ones. We introduce PixHt-Lab, a system leveraging an explicit mapping from pixel height representation to 3D space. Using this mapping, PixHt-Lab reconstructs both the cutout and background geometry and renders realistic, diverse lighting effects for image compositing. Given a surface with physically-based materials, we can render reflections with varying glossiness. To generate more realistic soft shadows, we further propose using 3D-aware buffer channels to guide a neural renderer. Both quantitative and qualitative evaluations demonstrate that PixHt-Lab significantly improves soft shadow generation. Project: https://shengcn.github.io/PixHtLab/

1. Introduction

Image compositing is a powerful tool widely used for image content creation, combining interesting elements from different sources to create a new image. One challenging task is adding lighting effects to make the compositing realistic and visually appealing. Lighting effects often involve complex interactions between the objects in the compositing, so their manual creation is tedious. It requires a significant amount of effort, especially for soft shadows cast by area lights and realistic reflections on the glossy surface with the Fresnel effect [62].

Many methods that generate lighting effects for 3D scenes have been well-studied [21], but 3D shapes are often unavailable during image compositing. Recent advancements in deep learning made significant progress in lighting

effect generation in 2D images, especially for shadow generation. A series of generative adversarial networks (GANs) based methods [15, 27, 60, 68] have been proposed to automatically generate hard shadows to match the background by training with pairs of shadow-free and shadow images. Those methods only focus on hard shadow generation, and their generated hard shadow cannot be edited freely. More importantly, the light control of those methods is implicitly represented in the background image. In real-world image creation scenarios, however, the background is often well-lit or even in pure color under a studio lighting setting, making those methods unusable. Also, editing the shadow is often needed on a separate image layer when the image editing is still incomplete.

To address these issues, a recent work SSN [48] proposes to learn the mapping between the image cutouts and the corresponding soft shadows based on a controllable light map. Although it achieves promising results, it assumes that the shadow receiver is just a ground plane and the object is always standing on the ground, which limits its practical usage. This limitation is addressed by SSG [47], which proposes a new 2.5D representation called pixel height, which is shown to be better suited for shadow generation than previous 2.5D presentations like depth map. Hard shadow on general shadow receivers can be computed by a ray tracing algorithm in the pixel-height space. A neural network renderer is further proposed to render the soft shadow based on the hard shadow mask. It achieves more controllability and it works in more general scenarios, but the lack of 3D geometry guidance makes the soft shadows unrealistic and prone to visual artifacts when they are cast on general shadow receivers like walls. In addition, SSG proposes an algorithm to render the specular reflection by flipping the pixels according to their pixel height. However, the use case is very limited as it cannot be directly applied to simulate realistic reflection effects on more general materials (see Fig. 1 (c)).

We introduce a controllable pixel height-based system called PixHt-Lab that provides lighting effects such as soft shadows and reflections for physically based surface materials. We introduce a formulation to map the 2.5D pixel height representation to the 3D space. Based on this mapping, geometry of both the foreground cutout and the background surface can be directly reconstructed by their corresponding pixel height maps. As the 3D geometry can be reconstructed, the surface normal can also be computed. Using a camera with preset extrinsic and intrinsics, light effects, including reflections, soft shadows, refractions, etc., can be rendered using classical rendering methods based on the reconstructed 3D geometry or directly in the pixel height space utilizing the efficient data structure (See Sec. 3.3) derived from the pixel height representation.

As the soft shadow integration in classical rendering algorithms is slow, especially for large area lights, we propose to train a neural network renderer SSG++ guided by 3D-aware buffer channels to render the soft shadow on general shadow receivers in real-time. Quantitative and qualitative experiments have been conducted to show that the proposed SSG++ guided by 3D-aware buffer channels significantly improves the soft shadow quality on general shadow receivers than previous soft shadow generation works. Our main contributions are:

- A mapping formulation between pixel height and the 3D space. Rendering-related 3D geometry properties, e.g., normal or depth, can be computed directly from the pixel height representation for diverse 3D effects rendering, including reflection and refraction.
- A novel soft shadow neural renderer, SSG++, guided by 3D-aware buffer channels to generate high-quality soft shadows for general shadow receivers in image composition.

2. Previous Work

Single Image 3D Reconstruction Rendering engines can be used to perform image composition. However, they require a 3D reconstruction of the image, which is a challenging problem. Deep learning-based methods [3, 14, 16, 24, 38, 39, 51] have been proposed to perform dense 3D reconstruction via low dimensional parameterization of the 3D models, though rendering quality is impacted by the missing high-frequency features. Many single-image digital human reconstruction methods [4, 23, 25, 33, 43, 44, 65–67, 69] show promising results, albeit they assume specific camera parameters. Directly rendering shadows on their reconstructed 3D models yields hard-to-fix artifacts [47] in the contact regions between the inserted object and the ground. More importantly, those methods cannot be applied to general objects, which limits their use for generic image composition.

Single Image Neural Rendering Image harmonization blends a cutout within a background in a plausible way. Classical methods achieve this goal by adjusting the appearance statistics [18, 34, 37, 41, 57]. Recently, learning-based methods [7, 19, 20, 26, 52, 58, 59] trained with augmented real-world images were shown to provide more robust results. However, these methods focus on global color adjustment without considering shadows during composition. Single image portrait relighting methods [49, 56, 70] adjust the lighting conditions given a user-provided lighting environment, although they only work for human portraits. [11] considers the problem of outdoor scene relighting from a single view using intermediary predicted shadow layers, which could be trained on cutout objects, but their method only produces hard shadows. Neural Radiance Field-based methods (e.g., [29, 30, 42, 54]) propose to encode the scene geometry implicitly, but require multiple images as input.

Soft Shadow Rendering is a well-studied technique in computer graphics, whether for real-time applications [1, 2, 5,9,10,12,13,31,40,45,46,53,55,63] or global illumination methods [8, 22, 50, 61]. It requires exact 3D geometry as input, preventing its use for image composition.

Recent neural rendering methods can address the limited input problem and render shadows for different scenarios. Scene level methods [35, 36] show promising results but require multiple input images. Generative adversarial networks (GANs) have achieved significant improvements on image translation tasks [17, 28], and subject-level shadow rendering methods [15,27,60,68] propose to render shadow using GANs. Unfortunately, these methods have two main limitations: they can only generate hard shadows, and prevents user editability, which is desired for artistic purposes. SSN [48] proposed a controllable soft shadow generation method for image composition, but is limited to the ground plane and cannot project shadows on complex geometries. Recently, SSG [47] further proposed a new representation called **pixel height** to cast soft shadows on more general shadow receivers and render specular reflection on the ground. Unfortunately, the shadow quality of SSG degrades on complex geometries as they are not explicitly taken into account by the network. Furthermore, its reflection rendering is limited to specular surfaces. In contrast, our proposed SSG++ is guided by 3D geometry-aware buffer channels that can render more realistic soft shadows on generic shadow receivers. We further connect the pixel height representation to 3D by using an estimated per-pixel depth and normal, increasing the reflections' realism.

3. Method

We propose a novel algorithm (Fig. 2) to render reflection and soft shadow to increase image composition realism based on pixel height [47], which has been shown to be more suitable for shadow rendering. Pixel height explicitly captures the object-ground relation, and thus it better keeps the object uprightness and the contact point for shadow rendering. Moreover, it allows intuitive user control to annotate or correct the 3D shape of an object.

Our first key insight is that 3D information that highly correlates with rendering many 3D effects, e.g., spatial 3D position, depth, normal, etc., can be recovered by a mapping (see Sec. 3.1) given the pixel height representation. The second key idea is that soft shadows on general shadow receivers are correlated with the relative 3D geometry distance between the occluder and the shadow receiver. Based on the mapping from pixel height to 3D, several geometry-aware buffer channels (see Sec. 3.2) are proposed to guide the neural renderer to render realistic soft shadows on general shadow receivers. Moreover, acquiring the 3D information enables rendering additional 3D effects, e.g., reflections and refractions.

Fig. 2 shows the overview of our method. Given the 2D cutout and background, the pixel height maps for the cutout and background can be either predicted by a neural network [47] or labeled manually by the user. 3D geometry that is used in rendering can be computed using our presented method (see Sec. 3.1). Finally, our renderer (see Sec. 3.3) can add 3D effects to make the image composition more realistic.

3.1. Connecting 2.5D Pixel Height to 3D

Here we describe the equation that connects 2.5D pixel height to its corresponding 3D point, and Fig. 3 shows the camera and relevant geometry and variables. We define O, the foot point of O', as the origin of the coordinate system. For convenience, we define the camera intrinsics by three vectors: 1) the vector c from the camera center O' to the top left corner of the image plane, 2) the right vector of the image plane a, and 3) the down vector of the image plane b. Any point P and its foot point Q can be projected by the camera centered at O'. The points P and Q projected on the image plane are denoted as P' and Q'.

$$\begin{bmatrix} x_{O'} \\ y_{O'} \\ z_{O'} \end{bmatrix} + \begin{bmatrix} x_a & x_b & x_c \\ y_a & y_b & y_c \\ z_a & z_b & z_c \end{bmatrix} \begin{bmatrix} u_{P'} \\ v_{P'} \\ 1 \end{bmatrix} w = \begin{bmatrix} x_P \\ y_P \\ z_P \end{bmatrix} \tag{1}$$

$$y_{o'} + \begin{bmatrix} y_a & y_b & y_c \end{bmatrix} \begin{bmatrix} u_{Q'} \\ v_{Q'} \\ 1 \end{bmatrix} w = y_Q \tag{2}$$

The relationship between a 3D point P and its projection P' is described by the projection Eq. 1 under the pinhole camera assumption. From the definition of pixel height representation, the foot point Q' of P' in the image space is on the ground plane, i.e., $y_Q = 0$. Solving Eq. 2 provides w:

$$w = \frac{-y_{o'}}{y_a u_{q'} + b_y v_{q'} + y_c} \tag{3}$$

The pixel height representation has no pitch angle assumption, thus P can be directly computed using the w in Eq. 3. By re-projecting the P' back, the 3D point P can be calculated as

$$P = \begin{bmatrix} x_{O'} \\ y_{O'} \\ z_{O'} \end{bmatrix} + \begin{bmatrix} x_a & x_b & x_c \\ y_a & y_b & y_c \\ z_a & z_b & z_c \end{bmatrix} \begin{bmatrix} u_{P'} \\ v_{P'} \\ 1 \end{bmatrix} \frac{-h}{y_a u_{Q'} + y_b v_{Q'} + y_c}. \tag{4}$$

Horizon controllability is important as different horizons will change the soft shadow distortion. However, as shown in Fig. 4, changing the horizon will change the camera pitch, which violates the no pitch assumption from pixel height representation [47] and leads to tilted geometry. To resolve the issue, we propose to use a tilt-shift camera model for our application. When the user changes the horizon, the vector c in Fig. 3 will move vertically to align the horizon to keep the image plane perpendicular to the ground. In this way, the no-pitch assumption is preserved for the correct reconstruction, as shown in Fig. 4 (d).

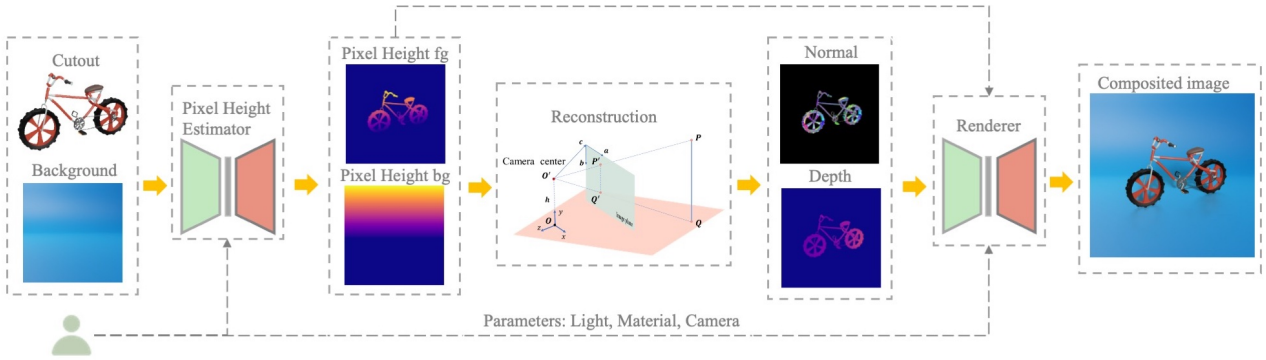


Figure 2. **System overview of PixHt-Lab**. Given a 2D cutout and background, the pixel height maps for the cutout and background can be either predicted by a neural network [47] or labeled manually by the user. 3D scene information and the relevant buffer channels can then be computed from pixel height based on our formulation presented in Sec. 3.1. Finally, our neural renderer SSG++ renders the requested lighting effects using the buffer channels (see Sec. 3.3).

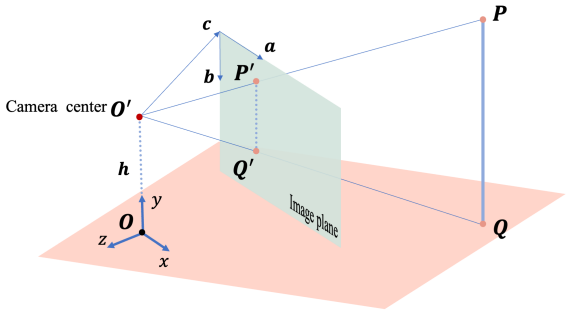
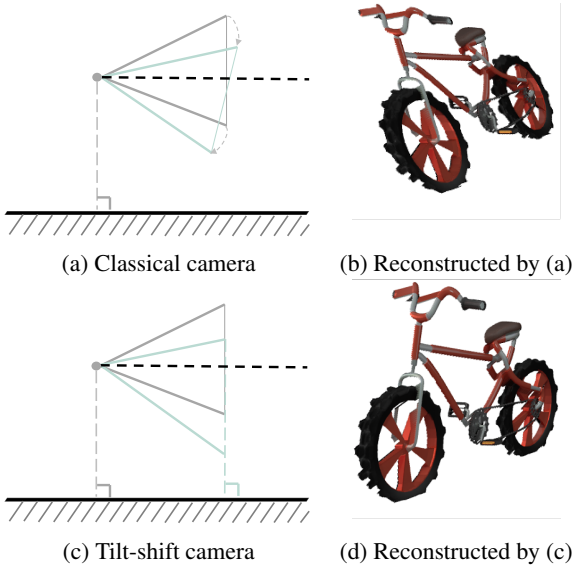


Figure 3. **Connecting pixel height to 3D.** Given the camera at the center O' and its foot point O, a point P in 3D space with its foot point Q. P' and Q' are their projection positions on the image plane. c is the vector from O' to the up left corner of the image plane. a and b are the right directions and down direction vector relative to the image plane.



(a) Classical camera (b) Reconstructed by (a)

(c) Tilt-shift camera (d) Reconstructed by (c)

Figure 4. **Tilt shift camera model**. Horizon position is a controllable parameter to change the shadow perspective shape. (a) shows changing the horizon is equivalent to changing the pitch in the classical model. (b) shows the reconstructed 3D model. (c) shows we use a tilt-shift camera. (d) shows the 3D vertical line can be preserved to be perpendicular to the ground after reconstruction.

3.2. 3D Buffer Channels for Soft Shadow Rendering

Our methods can be applied to arbitrary shape lights, but for discussion purposes, we assume the light for our discussion is disk shape area light. It is challenging to render high-quality soft shadows for general shadow receivers given only image cutouts and the hard shadow mask, as the soft shadow is jointly affected by multiple factors: the light source geometry, the occluder, the shadow receiver, the spatial relationship between the occluder and the shadow receiver, etc. SSG [47] is guided by the cutout mask, the hard shadow mask, and the disk light radius as inputs. The shadow boundary is softened uniformly (see Figs. 1 and 7) as SSG is unaware of the geometry-related information relevant to soft shadow rendering. We propose to train a neural network SSG++ to learn how these complex factors will jointly affect the soft shadow results.

3D-Aware Buffer channels. Our SSG++ takes several 3D-aware buffer channels (see Fig. 5) relevant to soft shadow rendering. The buffer channels are composed of several maps: the cutout pixel height map; the gradient of the background pixel height map; the hard shadow from the center of the light L; the sparse hard shadows map; the relative distance map between the occluder and the shadow receiver in pixel height space. For illustration purposes, we composite foreground pixel height and background pixel

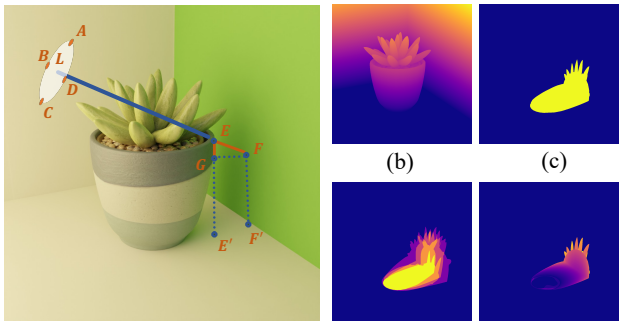


Figure 5. **Buffer channels**. (a) illustrates how the buffer channels are computed. See the text for more details. (b) shows the pixel height maps of the foreground cutout and the background. (c) is the hard shadow cast by the center of the disk area light L. (d) is the sparse hard shadows map cast by A, B, C, D, which are four extreme points of the area light L. (e) is the distance between EF in pixel height space.

height in Fig. 5 (b).

The cutout pixel height and background pixel height map describe the geometry of the cutout and the background. In our implementation, we use the gradient map of the background pixel height as input to make it translation invariant. The pixel height gradient map will capture the surface orientation similar to a normal map.

The sparse hard shadows map can also guide the network to be aware of the shadow receiver's geometry. Another important property of this channel is that the sparse hard shadows describe the outer boundary of the soft shadow. The overlapping areas of the sparse hard shadows are also a hint of darker areas in the soft shadow. Experiments in Sec. 4 show this channel plays the most important role among all the buffer channels. The four sparse hard shadows are sampled from the four extreme points of the disk light L as shown in Fig. 5.

The relative distance map in pixel height space defines the relative spatial distance between the occluder and the shadow receiver. The longer the distance, the softer the shadow will be in general. This channel guides the network to pay attention to shadow regions that have high contrast. The formal definition of the relative distance in pixel height space is: $||(u_p, v_p, h_p) - (u_q, v_q, h_q)||_2^2$, where p, q are two points, u, v are the coordinates in the pixel space, h is the pixel height.

Dataset and Training. We follow SSN [48] and SSG [47] to generate a synthetic dataset to train SSG++. In practice, we randomly picked 100 general foreground objects from ModelNet [64] and ShapeNet [6] with different categories, including human, plants, cars, desks, chairs, airplanes, etc. We also picked different backgrounds: ground plane, T shape wall, Cornell box, and curved plane. To

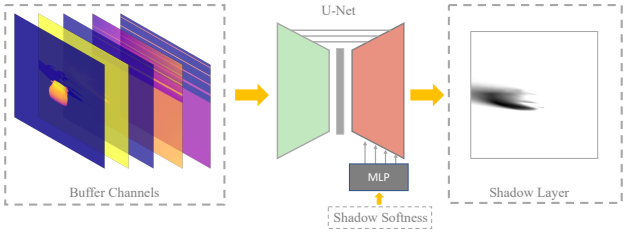


Figure 6. **SSG++ pipeline**. SSG++ uses a modified U-net where the embedded shadow softness is concatenated in the decoder.

cover diverse relative positions between the foreground and background, we randomly generate 200 scenes from the selected foreground and background objects. For each scene, we further randomly sampled 100 lights per scene from a different direction with random area light sizes. In total, the synthetic dataset has 20k training data. SSG++ follows the SSG architecture as shown in Fig. 6. We implement the SSG++ using PyTorch [32]. The training takes 8 hrs on average to converge with batch size 50 and learning rate $2e^{-5}$ in a RTX-3090.

3.3. Ray Tracing in Pixel Height Representation

Eq. 1 in Sec. 3.1 connects the pixel height to 3D. Although we do not know the camera extrinsic and intrinsic for the image cutout or the background, we can use a default camera to reconstruct the scene, given the pixel height. When the 3D position for the 2D pixel can be computed, the surface normal can be approximated if we assume neighborhood pixels are connected. When the surface normal can be reconstructed, 3D effects, including reflection, refraction and relighting, can be rendered if surface materials are given.

One can perform the 3D effects rendering using a classical graphics renderer to render lighting effects for image compositing. We noticed that the pixel height space naturally provides an acceleration mechanism for tracing. Specifically, SSG [47] proposes a ray-scene intersection algorithm in pixel height space. Although the ray-scene intersection is designed for tracing visibility, it can be easily modified to trace the closest hit pixel given a ray origin and ray direction in pixel height space. In the pixel height space, the ray-scene intersection check happens only along a line between the start and the end pixels. The complexity of the ray-scene intersection check in pixel height space is $\mathcal{O}(H)$ or $\mathcal{O}(W)$, without the need to check the intersection with each pixel or each reconstructed triangle. Therefore, in practice, we perform ray tracing in the pixel height space in PixHt-Lab. We implemented the method using CUDA. It took around 7s to render a noise free reflection for 512×512 resolution image with 200 samples per pixel. Reflection results on different surface materials can be found in Fig. 8.

Table 1. Comparison with SSN [48] and SSG [47] on the ground-shadow benchmark.

Method	RMSE ↓	RMSE-s ↓	SSIM ↑	ZNCC ↑
SSN	0.1207	0.1064	0.8379	0.6118
SSG	0.0254	0.0221	0.8547	0.5679
SSG++(ours)	**0.0165**	**0.0140**	**0.9216**	**0.8180**

Additional examples can be found in *supplementary materials*.

4. Experiments

Here we show quantitative (the benchmark, metrics for comparison, ablation study) and qualitative evaluation of the buffer channels by comparing to related work.

4.1. Quantitative Evaluation of Buffer Channels

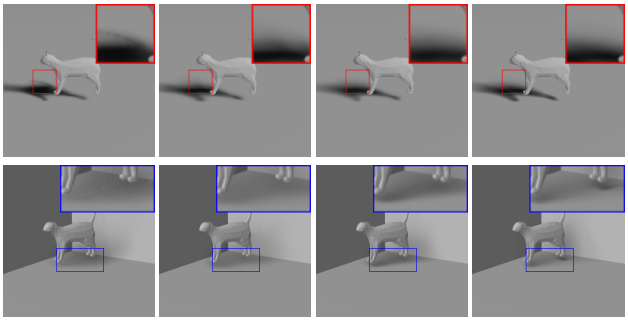
Benchmark: To compare our 3D-aware buffer channels fairly with SSN [48] that has ground plane assumption, we build two evaluation benchmarks: 1) a ground-shadow benchmark and 2) a wall-shadow benchmark.

The two benchmarks share the same assets, but the ground shadow benchmark only has shadows cast on the ground plane, and the wall shadow benchmark always has part of the shadows cast on walls. The foreground objects in the assets are composed of 12 new models randomly selected online with different types: robots, animals, humans, bottles, etc. The background objects in the assets are four new backgrounds with different shapes: one wall corner, two wall corners, steps, and curved backgrounds to test the generalization ability to unseen backgrounds. We randomly generate 70 scenes using those unseen foregrounds and background models with different poses of the foreground and background.

We uniformly sample 125 lights with different positions and different area sizes per scene for the wall shadow benchmark. As shadows on the ground have less variation than the shadows on the wall, we sample eight lights with different positions and different area sizes per scene for the ground shadow benchmark. In total, the ground shadow benchmark is composed of 560 data, and the wall shadow benchmark is composed of 8,750 data. The resolution for each data is 256×256.

Metrics: We use the per-pixel metric RMSE and a scale-invariant RMSE-s [56]. Similar to [56], shadow intensity may vary, but the overall shapes are correct. We also used perception-based metrics SSIM and ZNCC to measure shadow quality. Our SSIM implementation uses 11×11 Gaussian filter with $\sigma = 1.5, k_1 = 0.01, k_2 = 0.03$.

SSG++ on the ground-shadow benchmark. As SSN has a ground plane assumption, we use the ground-shadow benchmark to compare fairly. We compare our SSG++ with SSN and the other soft shadow rendering network SSG pro-



(a) SSG (b) SSG-SS (c) SSG++ (d) GT

Figure 7. **Effects of buffer channels.** Best zoom-in. (a) shows the shadows rendered by SSG will be softened uniformly. (b). shows sparse hard shadow channels guide the neural network to be 3D-aware. (c) shows SSG++ can render better quality in the relatively darker regions(note the foot shadow in the second row). (d) is the ground truth shadow rendered by the physically based renderer.

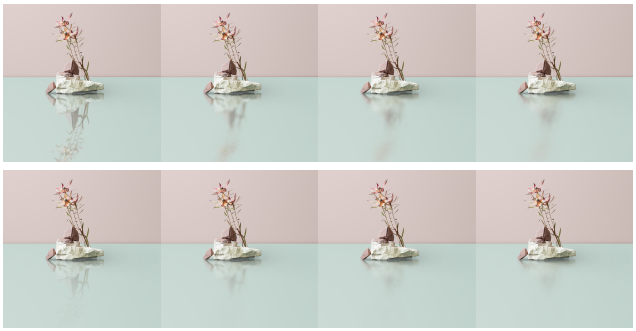


Figure 8. **Reflection.** PixHt-Lab can render reflection with different physical materials. From left to right, the ground surface glossiness increases. The top to bottom, the ground uses different η in Fresnel effects.

Table 2. Result on the wall-shadow benchmark. We show the effectiveness of each buffer channel. SSG-BH: SSG with background pixel height. SSG-D: SSG with XYH distance channel. SSG-D-BH: SSG with XYH distance and background pixel height. SSG-SS: SSG with the sparse shadow channel. SSG-SS-BH: SSG with sparse shadow channel and background pixel height. SSG++: SSG with all the buffer channels.

Method	RMSE ↓	RMSE-s ↓	SSIM ↑	ZNCC ↑
SSG	0.0242	0.0209	0.8561	0.6460
SSG-BH	0.0248	0.0207	0.8587	0.6506
SSG-D	0.0230	0.0210	0.8739	0.6499
SSG-D-BH	0.0231	0.0201	0.8752	0.6719
SSG-SS	0.0164	0.0149	0.9139	0.8228
SSG-SS-BH	0.0184	0.0158	0.9136	0.8029
SSG-SS-D	0.0159	0.0139	0.9153	0.8494
SSN++(ours)	**0.0153**	**0.0136**	**0.9277**	**0.8575**

Figure 9. **More results.** PixHt-Lab is agnostic to the cutout object categories. Lighting effects can be generated for general backgrounds. PixHt-Lab can also generate multiple soft shadows shown in the first column. The first column uses a step background. The second column uses a curved background. The third column uses an L shape wall background. The fourth column uses a corner background.

posed recently on the ground-shadow benchmark. Results are shown in Tab. 1. Our SSG++ outperforms SSN and SSG in all metrics. Compared with the STOA SSG, SSG++ improves RMSE by 35%, RMSE-s by 36%, SSIM by 7.8%, ZNCC by 44%. The statistics show in the simplest ground plane shadow receiver case, SSG++ still has significant improvement. Even in the simplest case, buffer channels significantly improve the soft shadow rendering quality.

SSG++ on the wall-shadow benchmark. SSG++ and SSG share the same backbone, but they are guided by different buffers. Therefore, we treat SSG as the basic baseline and do the ablation study together in this section. Results are shown in Tab. 2.

Our proposed SSG++ outperforms all the other methods guided by other subsets of the buffer channels. Each buffer channel outperforms SSG in all the metrics, showing that those 3D-aware buffer channels are useful to guide the SSG to render better soft shadows. SSG-D-BN fixes more errors than SSG-D or SSG-BN, showing that the combination of relative distance in pixel height space helps the neural network improve the soft shadow quality. SSG-SS significantly outperforms all the previous baselines by improving RMSE by 29% and SSIM by 4% than SSG-D-BH, which shows that the sparse shadows channel plays the most important role in guiding the SSG to render soft shadows. Combining the sparse shadow channel with the relative distance channel only improves RMSE by 3% and SSIM by 0.15% than only using the sparse shadow channel as additional channels while combing the sparse shadow channel with the background pixel height channel performs worse than only using the sparse shadow channels as an additional channel for SSG, with RMSE degraded by 12% and SSIM by 0.03%.

Our SSG++ combines the sparse shadow channel, the relative distance channel, and the background pixel height

Figure 10. Real foreground and background examples created with our GUI. Zoom in for the best view.

Figure 11. **Refraction.** Given the cutout and the background in the left image, the refraction lighting effect for the crystal ball can also be rendered by PixHt-Lab.

channel together and achieves the best performance, improving in all the metrics significantly. Compared with SSG, our SSG++ improves RMSE by 38%, RMSE-s by 33%, SSIM by 8% and ZNCC by 32%.

4.2. Qualitative Evaluation of Buffer Channels

Effects of buffer channels. Fig. 7 shows the effects of the buffer channels. Fig. 7 (b). shows the sparse shadow guides the neural network to render better contour shapes as the sparse hard shadows are samples from the outer contour of the shadow regions. However, when the geometry has complex shapes, and the sparse hard shadows are mixed together, e.g., the foot regions of the cat in the second row of Fig. 7, the relative spatial information is ambiguous. The relative distance map can further guide SSG++ to keep the regions close to the objects dark instead of over soft(See Fig. 7 (c) in the second row.).

5. Discussion

Light effects generated by PixHt-Lab. PixHt-Lab can reconstruct the surface normal solely based on the pixel height inputs. PixHt-Lab does not have assumptions on the cutout object types and background types. No matter for realistic cutouts or cartoon cutouts, studio backgrounds or real-world backgrounds, PixHt-Lab can render the light effects. (see Fig. 9, Fig. 10 and the demo video showing the PixHt-Lab system in the *supplementary materials*). Similar to SSG, PixHt-Lab allows the user to intuitively control the shadow direction and softness, control the horizon position to tweak the shadow distortion, and change parameters to control the physical parameters of the reflection. Our methods can also be applied to multiple object compositing and multiple shadows. Please refer to *supplementary materials* for more examples. There exist more potential additions for PixHt-Lab and other light effects such as refraction(see Fig. 11). Parameters related to the refraction surface, like the refraction index, can be controlled as well.

Limitation. As PixHt-Lab is based on the pixel height map and the common limitations for the pixel height representation apply to our methods as well. One of them is that it takes the image cutout as the proxy of the object and the back face or hidden surface contributing to the light effect generation is missing. A back face prediction neural network can be explored to address this problem. Another limitation specific to PixHt-Lab is that the proposed method uses the cutout color as the reflected color, which is not precise for cases when the surface has view-dependent colors.

6. Conclusion and Future Work

We propose a novel system PixHt-Lab for generating perceptually plausible light effects based on the pixel height representation. The mapping between the 2.5D pixel height and 3D has been presented to reconstruct the surface geometry directly from the pixel height representation. Based on the reconstruction, more light effects, including physically based reflection and refraction, can be synthesized for image compositing. Also, a novel method SSG++ guided by 3D-aware buffer channels is proposed to improve the soft shadow quality that is cast on general shadow receivers. Quantitative and qualitative experiments demonstrate that the results and generalization ability of the proposed SSG++ significantly outperforms previous deep learning-based shadow synthesis methods. However, our PixHt-Lab synthesizes the light effect solely based on the cutout colors. A back face prediction neural network may address the issue and is worth future exploration.

Acknowledgment: We appreciate constructive comments from the reviewers. Part of this work was done during Yichen's internship at Adobe.

References

[1] Thomas Annen, Zhao Dong, Tom Mertens, Philippe Bekaert, Hans-Peter Seidel, and Jan Kautz. Real-time, all-frequency shadows in dynamic scenes. *ACM Transactions on Graphics*, 27(3):1–8, Aug. 2008.

[2] Ulf Assarsson and Tomas Akenine-Moller. A Geometry-based Soft Shadow Volume Algorithm using Graphics Hardware. page 10.

[3] Manush Bhatt, Rajesh Kalyanam, Gen Nishida, Liu He, Christopher May, Dev Niyogi, and Daniel Aliaga. Design and deployment of photo2building: A cloud-based procedural modeling tool as a service. In *Practice and Experience in Advanced Research Computing*, pages 132–138. 2020.

[4] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J Black. Keep it smpl: Automatic estimation of 3d human pose and shape from a single image. In *European conference on computer vision*, pages 561–578. Springer, 2016.

[5] Eric Chan and Fredo Durand. Rendering Fake Soft Shadows with Smoothies. page 12.

[6] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015.

[7] Wenyan Cong, Jianfu Zhang, Li Niu, Liu Liu, Zhixin Ling, Weiyuan Li, and Liqing Zhang. Dovenet: Deep image harmonization via domain verification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8394–8403, 2020.

[8] Robert L Cook, Thomas Porter, and Loren Carpenter. Computer Graphics Volume18, Number3 July 1984. page 9, 1984.

[9] William Donnelly and Andrew Lauritzen. Variance shadow maps. In *Proceedings of the 2006 symposium on Interactive 3D graphics and games - SI3D '06*, page 161, Redwood City, California, 2006. ACM Press.

[10] Randima Fernando. Percentage-closer soft shadows. In *ACM SIGGRAPH 2005 Sketches on - SIGGRAPH '05*, page 35, Los Angeles, California, 2005. ACM Press.

[11] David Griffiths, Tobias Ritschel, and Julien Philip. Outcast: Single image relighting with cast shadows. *Computer Graphics Forum*, 43, 2022.

[12] Gaël Guennebaud, Loïc Barthe, and Mathias Paulin. Real-time soft shadow mapping by backprojection. page 8.

[13] Gaël Guennebaud, Loïc Barthe, and Mathias Paulin. High-Quality Adaptive Soft Shadow Mapping. *Computer Graphics Forum*, 26(3):525–533, Sept. 2007.

[14] Liu He, Jie Shan, and Daniel Aliaga. Generative building feature estimation from satellite images. *IEEE Transactions on Geoscience and Remote Sensing*, 61:1–13, 2023.

[15] Xiaowei Hu, Yitong Jiang, Chi-Wing Fu, and Pheng-Ann Heng. Mask-shadowgan: Learning to remove shadows from unpaired data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2472–2481, 2019.

[16] Yuchun Huang, Ping Ma, Zheng Ji, and Liu He. Part-based modeling of pole-like objects using divergence-incorporated 3-d clustering of mobile laser scanning point clouds. *IEEE Transactions on Geoscience and Remote Sensing*, 59(3):2611–2626, 2020.

[17] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.

[18] Jiaya Jia, Jian Sun, Chi-Keung Tang, and Heung-Yeung Shum. Drag-and-drop pasting. *ACM Transactions on Graphics*, 25(3):631–637, July 2006.

[19] Yifan Jiang, He Zhang, Jianming Zhang, Yilin Wang, Zhe Lin, Kalyan Sunkavalli, Simon Chen, Sohrab Amirghodsi, Sarah Kong, and Zhangyang Wang. SSH: A Self-Supervised Framework for Image Harmonization. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4812–4821, Montreal, QC, Canada, Oct. 2021. IEEE.

[20] Yifan Jiang, He Zhang, Jianming Zhang, Yilin Wang, Zhe Lin, Kalyan Sunkavalli, Simon Chen, Sohrab Amirghodsi, Sarah Kong, and Zhangyang Wang. Ssh: A self-supervised framework for image harmonization. *arXiv preprint arXiv:2108.06805*, 2021.

[21] James T Kajiya. The rendering equation. In *Proceedings of the 13th annual conference on Computer graphics and interactive techniques*, pages 143–150, 1986.

[22] James T Kajiya. THE RENDERING EQUATION. 20(4):8, 1986.

[23] Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7122–7131, 2018.

[24] Chen Kong, Chen-Hsuan Lin, and Simon Lucey. Using locally corresponding cad models for dense 3d reconstructions from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4857–4865, 2017.

[25] Christoph Lassner, Javier Romero, Martin Kiefel, Federica Bogo, Michael J Black, and Peter V Gehler. Unite the people: Closing the loop between 3d and 2d human representations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6050–6059, 2017.

[26] Jun Ling, Han Xue, Li Song, Rong Xie, and Xiao Gu. Region-aware adaptive instance normalization for image harmonization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9361–9370, 2021.

[27] Daquan Liu, Chengjiang Long, Hongpan Zhang, Hanning Yu, Xinzhi Dong, and Chunxia Xiao. Arshadowgan: Shadow generative adversarial network for augmented reality in single light scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8139–8148, 2020.

[28] Yifan Liu, Zengchang Qin, Tao Wan, and Zhenbo Luo. Auto-painter: Cartoon image generation from sketch by using conditional wasserstein generative adversarial networks. *Neurocomputing*, 311:78–87, 2018.

[29] Ricardo Martin-Brualla, Noha Radwan, Mehdi S. M. Sajjadi, Jonathan T. Barron, Alexey Dosovitskiy, and Daniel Duckworth. NeRF in the Wild: Neural Radiance Fields for Unconstrained Photo Collections. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7206–7215, Nashville, TN, USA, June 2021. IEEE.

[30] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, Jan. 2022.

[31] Ren Ng, Ravi Ramamoorthi, and Pat Hanrahan. All-Frequency Shadows Using Non-linear Wavelet Lighting Approximation. page 6.

[32] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[33] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. Osman, Dimitrios Tzionas, and Michael J. Black. Expressive Body Capture: 3D Hands, Face, and Body From a Single Image. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10967–10977, Long Beach, CA, USA, June 2019. IEEE.

[34] Patrick Pérez, Michel Gangnet, and Andrew Blake. Poisson image editing. In *ACM SIGGRAPH 2003 Papers*, pages 313–318. 2003.

[35] Julien Philip, Michaël Gharbi, Tinghui Zhou, Alexei A Efros, and George Drettakis. Multi-view relighting using a geometry-aware network. *ACM Trans. Graph.*, 38(4):78–1, 2019.

[36] Julien Philip, Sébastien Morgenthaler, Michaël Gharbi, and George Drettakis. Free-viewpoint indoor neural relighting from multi-view stereo. *ACM Transactions on Graphics (TOG)*, 40(5):1–18, 2021.

[37] F. Pitie, A.C. Kokaram, and R. Dahyot. N-dimensional probability density function transfer and its application to color transfer. In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, volume 2, pages 1434–1439 Vol. 2, Oct. 2005. ISSN: 2380-7504.

[38] Jhony K Pontes, Chen Kong, Anders Eriksson, Clinton Fookes, Sridha Sridharan, and Simon Lucey. Compact model representation for 3d reconstruction. *arXiv preprint arXiv:1707.07360*, 2017.

[39] Jhony K Pontes, Chen Kong, Sridha Sridharan, Simon Lucey, Anders Eriksson, and Clinton Fookes. Image2mesh: A learning framework for single image 3d reconstruction. In *Asian Conference on Computer Vision*, pages 365–381. Springer, 2018.

[40] William T Reeves, David H Salesin, and Robert L Cook. Rendering antialiased shadows with depth maps. In *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, pages 283–291, 1987.

[41] E. Reinhard, M. Adhikhmin, B. Gooch, and P. Shirley. Color transfer between images. *IEEE Computer Graphics and Applications*, 21(5):34–41, July 2001. Conference Name: IEEE Computer Graphics and Applications.

[42] Viktor Rudnev, Mohamed Elgharib, William Smith, Lingjie Liu, Vladislav Golyanik, and Christian Theobalt. NeRF for Outdoor Scene Relighting. In Shai Avidan, Gabriel Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *Computer Vision – ECCV 2022*, volume 13676, pages 615–631, Cham, 2022. Springer Nature Switzerland. Series Title: Lecture Notes in Computer Science.

[43] Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, and Hao Li. Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2304–2314, 2019.

[44] Shunsuke Saito, Tomas Simon, Jason Saragih, and Hanbyul Joo. Pifuhd: Multi-level pixel-aligned implicit function for high-resolution 3d human digitization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 84–93, 2020.

[45] Michael Schwarz and Marc Stamminger. Bitmask Soft Shadows. *Computer Graphics Forum*, 26(3):515–524, Sept. 2007.

[46] Pradeep Sen, Mike Cammarano, and Pat Hanrahan. Shadow silhouette maps. *ACM Transactions on Graphics (TOG)*, 22(3):521–526, 2003.

[47] Yichen Sheng, Yifan Liu, Jianming Zhang, Wei Yin, A Cengiz Oztireli, He Zhang, Zhe Lin, Eli Shechtman, and Bedrich Benes. Controllable shadow generation using pixel height maps. In *European Conference on Computer Vision*, pages 240–256. Springer, 2022.

[48] Yichen Sheng, Jianming Zhang, and Bedrich Benes. Ssn: Soft shadow network for image compositing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4380–4390, 2021.

[49] Zhixin Shu, Sunil Hadap, Eli Shechtman, Kalyan Sunkavalli, Sylvain Paris, and Dimitris Samaras. Portrait Lighting Transfer Using a Mass Transport Approach. *ACM Transactions on Graphics*, 37(1):2:1–2:15, Oct. 2017.

[50] François X Sillion, James Arvo, Stephen Westin, and Donald P Greenberg. A Global Illumination Solution for General Reflectance Distributions. *Computer Graphics*, 25(4):11, 1991.

[51] Ayan Sinha, Asim Unmesh, Qixing Huang, and Karthik Ramani. Surfnet: Generating 3d shape surfaces using deep residual networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6040–6049, 2017.

[52] Konstantin Sofiiuk, Polina Popenova, and Anton Konushin. Foreground-aware semantic representations for image harmonization. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1620–1629, 2021.

[53] Cyril Soler and François X. Sillion. Fast calculation of soft shadow textures using convolution. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques - SIGGRAPH '98*, pages 321–332, Not Known, 1998. ACM Press.

[54] Pratul P. Srinivasan, Boyang Deng, Xiuming Zhang, Matthew Tancik, Ben Mildenhall, and Jonathan T. Barron. NeRV: Neural Reflectance and Visibility Fields for Relighting and View Synthesis. In *2021 IEEE/CVF Conference*

on Computer Vision and Pattern Recognition (CVPR), pages 7491–7500, Nashville, TN, USA, June 2021. IEEE.

[55] Marc Stamminger and George Drettakis. Perspective shadow maps. In John Hughes, editor, *Proceedings of ACM SIGGRAPH*, Annual Conference Series, pages 557 – 562. ACM Press/ ACM SIGGRAPH, July 2002.

[56] Tiancheng Sun, Jonathan T. Barron, Yun-Ta Tsai, Zexiang Xu, Xueming Yu, Graham Fyffe, Christoph Rhemann, Jay Busch, Paul Debevec, and Ravi Ramamoorthi. Single image portrait relighting. *ACM Transactions on Graphics*, 38(4):1–12, Aug. 2019.

[57] Michael W. Tao, Micah K. Johnson, and Sylvain Paris. Error-Tolerant Image Compositing. In Kostas Daniilidis, Petros Maragos, and Nikos Paragios, editors, *Computer Vision – ECCV 2010*, Lecture Notes in Computer Science, pages 31–44, Berlin, Heidelberg, 2010. Springer.

[58] Yi-Hsuan Tsai, Xiaohui Shen, Zhe Lin, Kalyan Sunkavalli, Xin Lu, and Ming-Hsuan Yang. Deep Image Harmonization. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2799–2807, Honolulu, HI, July 2017. IEEE.

[59] Jeya Maria Jose Valanarasu, He Zhang, Jianming Zhang, Yilin Wang, Zhe Lin, Jose Echevarria, Yinglan Ma, Zijun Wei, Kalyan Sunkavalli, and Vishal M Patel. Interactive portrait harmonization. *arXiv preprint arXiv:2203.08216*, 2022.

[60] Yifan Wang, Brian L Curless, and Steven M Seitz. People as scene probes. In *European Conference on Computer Vision*, pages 438–454. Springer, 2020.

[61] Stephen H Westin, James R Arvo, and Kenneth E Torrance. Predicting Reflectance Functions from Complex Surfaces. page 10.

[62] Wikipedia. Fresnel equations. `https://en.wikipedia.org/wiki/Fresnel_equations`.

[63] Lance Williams. CASTING CURVED SHADOWS ON CURVED SURFACES. page 5.

[64] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1912–1920, 2015.

[65] Donglai Xiang, Hanbyul Joo, and Yaser Sheikh. Monocular Total Capture: Posing Face, Body, and Hands in the Wild. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10957–10966, Long Beach, CA, USA, June 2019. IEEE.

[66] Yuanlu Xu, Song-Chun Zhu, and Tony Tung. DenseRaC: Joint 3D Pose and Shape Estimation by Dense Render-and-Compare. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7759–7769, Seoul, Korea (South), Oct. 2019. IEEE.

[67] Richard Zhang, Jun-Yan Zhu, Phillip Isola, Xinyang Geng, Angela S. Lin, Tianhe Yu, and Alexei A. Efros. Real-Time User-Guided Image Colorization with Learned Deep Priors. Technical Report arXiv:1705.02999, arXiv, May 2017. arXiv:1705.02999 [cs] type: article.

[68] Shuyang Zhang, Runze Liang, and Miao Wang. Shadowgan: Shadow synthesis for virtual objects with conditional adversarial networks. *Computational Visual Media*, 5(1):105–115, 2019.

[69] Zerong Zheng, Tao Yu, Yixuan Wei, Qionghai Dai, and Yebin Liu. Deephuman: 3d human reconstruction from a single image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7739–7749, 2019.

[70] Hao Zhou, Sunil Hadap, Kalyan Sunkavalli, and David Jacobs. Deep Single-Image Portrait Relighting. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7193–7201, Seoul, Korea (South), Oct. 2019. IEEE.