Controllable Shadow Generation Using Pixel Height Maps*

Yichen Sheng*[1], Yifan Liu*[2], Jianming Zhang[3], Wei Yin[2], A. Cengiz Oztireli[4], He Zhang[3], Zhe Lin[3], Eli Shechtman[3], and Bedrich Benes[1]

[1] Purdue University [2] University of Adelaide
[3] Adobe Research [4] University of Cambridge

Abstract. Shadows are essential for realistic image compositing from 2D image cutouts. Physics-based shadow rendering methods require 3D geometries, which are not always available. Deep learning-based shadow synthesis methods learn a mapping from the light information to an object's shadow without explicitly modeling the shadow geometry. Still, they lack control and are prone to visual artifacts. We introduce "Pixel Height", a novel geometry representation that encodes the correlations between objects, ground, and camera pose. The Pixel Height can be calculated from 3D geometries, manually annotated on 2D images, and can also be predicted from a single-view RGB image by a supervised approach. It can be used to calculate hard shadows in a 2D image based on the projective geometry, providing precise control of the shadows' direction and shape. Furthermore, we propose a data-driven soft shadow generator to apply softness to a hard shadow based on a softness input parameter. Qualitative and quantitative evaluations demonstrate that the proposed Pixel Height significantly improves the quality of the shadow generation while allowing for controllability.

1 Introduction

Shadow generation is an important step for image composting that enhances photo realism and adds positional and directional cues for the composed objects. Advanced image editing techniques enable composing objects into a new background with accurate segmentation and matting [23] and harmonization of color styles [17]. However, the composited objects are not realistic if no matching shadows are synthesized (see the 1st and 3rd images in the second row in Fig. 1). Manually creating a perceptually plausible shadow for a 2D object is tedious, even for an experienced artist, especially for extended (linear or area) light sources.

Mature techniques that calculate soft shadows for 3D scenes exist [6,26]. However, 3D shape information is often unavailable when we composite objects from real images. Recent deep learning advancements brought significant progress to shadow generation in 2D images. A series of methods [14,21,45]

* Y. Sheng and Y. Liu contributed equally.

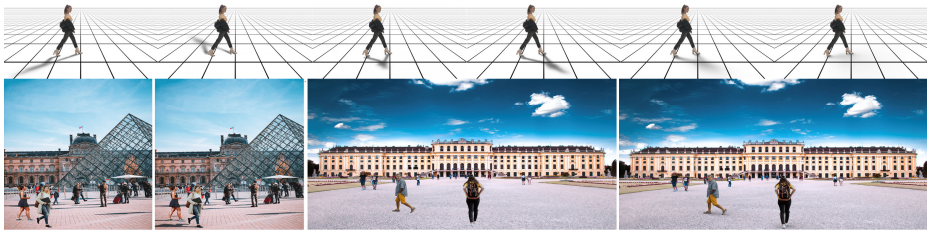

Fig. 1. Controllable shadow generation with the proposed method. **First row**: With the help of our new introduced *Pixel Height* of an object, users can control the position and the softness of the generated shadows. **Second row**: The composited images with our generated shadows (2nd and 4th) are much more natural than the direct composites (1st and 3rd).

based on generative adversarial networks (GANs) have been proposed to automatically generate shadows by training with pairs of shadow and shadow-free images. These methods mainly focus on generating hard shadows, and the final results are not editable. Moreover, these methods require the background scene to implicitly provide light information, while in many application scenarios, objects are either composited on abstract or pure color background. Also, shadow editing needs to be applied on separate image layers with background images missing or incomplete at the time of editing. Therefore, shadow generation for object cutouts with user control is more suited for professional image editing workflows. Recently, Sheng et al. [36] proposed to learn a mapping from a 2D cutout of the object to the corresponding soft shadows based on a controllable lightmap and achieved promising results. However, due to the lack of geometry guidance, this method cannot generalize well for varying scenes and may lead to visible artifacts in the generated shadows.

We introduce a controllable and editable shadow generation method for 2D object cutouts. We introduce *Pixel Height*, a new 2.5D shape representation for an image to provide geometry guidance. The Pixel Height is defined as the pixel distance between a point on an object and its *footpoint*, namely its vertical projection on the ground in the image (see Fig. 2-(a)). Based on Pixel Height, we can explicitly compute the shadow point based on projective geometry. The Pixel Height could be measured and annotated on a 2D image or calculated from synthetic data with 3D object models. Similar to monocular depth estimation, Pixel Height can also be estimated from a single RGB image by a data-driven method. We collect synthetic and real annotated data (see Fig. 2-(b) and (c)) to train a Pixel Height map prediction model for object cutouts.

Given the annotated or predicted Pixel Height map of an object, we render a hard shadow based on the position of the horizon and the point light in the 2D image space with a proposed *hard shadow renderer*. To add softness to the shadow, we learn an efficient and controllable mapping from the hard shadow to the soft shadows based on a softness parameter using *a soft shadow generator*. As shown in Fig. 1, our system can generate varying shadow maps controlled by

the light source position and the softness control. Our method explicitly models the shadow geometry that is more controllable and robust than methods that directly predict shadows based on an image background or a light map.

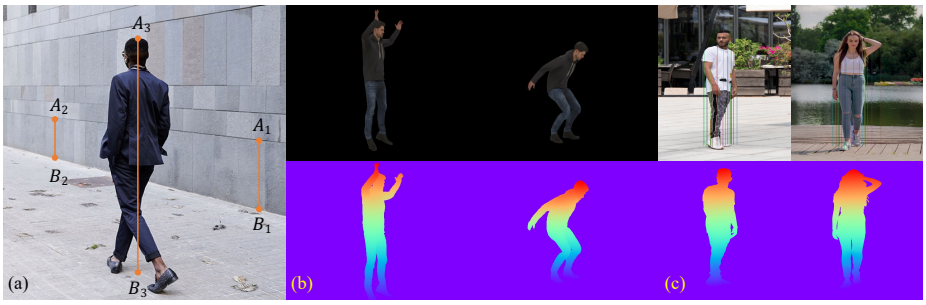

Fig. 2. Pixel Height. (a). The number of pixels between point A_i and B_i is the Pixel Height for point A_i. We collected two datasets with Pixel Height annotation: Synthetic60K and Real1500. (b) shows the sample data with various poses from 3D models. (c) shows samples of the sparsely labeled data and the interpolated Pixel Height map. Note that every Pixel Height map is divided by its max-height value for visualization.

We conduct extensive experiments to show that the Pixel Height map improves the controllability of shadow generation. Realistic shadows are synthesized by easy and intuitive user control given the RGB image and an object segmentation mask. Qualitative and quantitative results demonstrate that our method generates higher quality shadows than previous interactive and automatic shadow generation algorithms in 2D images. Our main contributions are:

- A formulation of hard shadow rendering in images based on a novel geometry representation, Pixel Height, which can be manually labeled or predicted by a model from a single image.
- A controllable shadow generation framework, where users control the position and softness of an object shadow. The framework consists of a Pixel Height estimation, a hard shadow renderer and a soft shadow generator.
- Extensive evaluation and analysis, showing improved quality and controllability of our proposed Pixel Height based shadow synthesis method.

2 Related Work

Shadow rendering in graphics. Shadow rendering based on 3D geometries is a well-studied technique in computer graphics. In real-time rendering, shadow volume [2,34] and shadow map based rendering techniques are mainstream approaches [7,28,35,41]. The soft shadow is approximated either by blurring the hard shadow boundaries [1,4,8,11,12,38] or weighted sum of a set of hard shadows

sampling on an area or volume light source [6]. Many works [9,24,26] have been proposed to speed up this sampling process by adjusting the density and the weight. Besides, some simplified geometries [10,30] or light representations [13] are proposed to render shadows in real-time. Global illumination algorithms [5,18,25,37,40] render soft shadows implicitly. Such methods can render realistic shadows for complicated objects given accurate 3D object models. However, 3D models are not always available for objects in real images, especially in image compositing tasks in computer vision.

Shadow synthesis with deep learning. In recent years, generative adversarial networks (GANs) have achieved significant improvements on image translation tasks [16,22]. A series of works [14,15,21,45,39] have been introduced for generating shadows directly from a composited shadow-free image based on the object mask guidance. ARShadowGan [21] renders a dataset by inserting 3D objects into real background images with augmented reality. Hong *et al.* [14] generate the shadow-free images by removing the shadow region from the real-world images. These methods try to predict the style and the color of the final shadow by a data-driven method, but they cannot provide controllability for the user.

Sheng *et al.* [36] propose an interactive soft shadow generation network based on a user-provided lightmap. Physics-based methods on 3D object models render their training data. The network is trained to learn the mapping from the 2D object cutout and environment lightmap to the soft shadow maps. Contrary to the previous works, we generate soft shadows by first generating a hard shadow and converting it to a soft shadow using a softness input parameter. This hard-to-soft transformation is much easier to learn. The hard shadow can be obtained with our proposed pixel-height map by calculating the occlusion directly in the 2D projection space with a simple shadow projection model.

Geometry representation. Similar to monocular depth estimation [33], recovering Pixel Height map from a single image is an ill-posed problem. Numerous methods [20,43,44] exist to estimate depth from a single view image by supervised methods. As the depth is a 2.5D representation, the intrinsic camera parameters are required for recovering the 3D shape of the object. The 3D point cloud [29] is another geometry representation for 3D objects' shape. They can be captured by special scanners, recovered from multi-view images, but cannot be labeled directly just from a monocular image. Furthermore, methods [31,32] have been proposed to directly recover the 3D shape, especially for humans from a monocular image. The proposed Pixel Height map is a new geometry representation, which reflects the correlation among the object, shadow receiver, and camera pose. It is easier to interpret and annotate, and it is useful for applications that require explicit occluder-receiver constraints such as shadow generation.

3 Method

We propose a new approach to generate perceptually plausible soft shadows on 2D images without 3D object models. The key idea of our approach is to render the object's hard shadow from a point light in the image plane following

a simplified projective geometry constraint (see Sec. 3.1), and then synthesize the corresponding soft shadow based on the hard shadow using a data-driven approach (see Sec. 3.3).

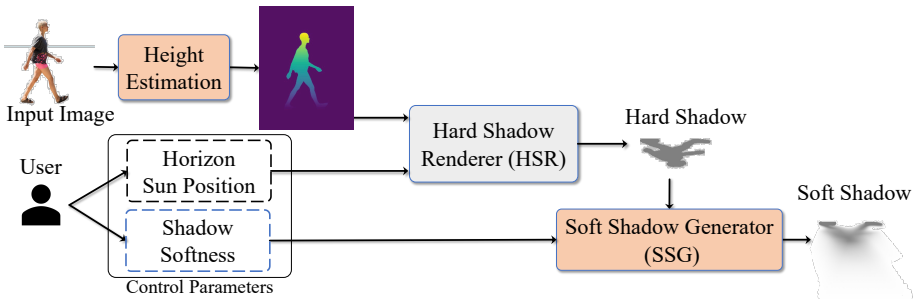

Fig. 3. Given a 2D foreground image and its Pixel Height map, a user can control the position and the softness of the generated shadow by the user-defined light information or the existing image-based light information. The Pixel Height map can be manually annotated on images or predicted from a single image by training a model.

We need to know the shape of the object and its relationship with the shadow receiver and the camera to render a hard shadow on the image plane. A new geometry representation Pixel Height is proposed to represent the object shape in 2D images, which is essential to rendering the hard shadow. We verify that this representation can be estimated by a data-driven approach (see Sec. 3.2).

As shown in Fig. 3, given the foreground object image and its mask, we can annotate or estimate its Pixel Height map. The hard shadow's position and shape can be determined by the controllable light information (the sun position) and the ground (the horizon line). Finally, based on the hard shadow, the soft shadow generator can produce a perceptually pleasing soft shadow according to the softness control parameter. The user could provide all the controllable variables with a simple GUI (see the supplementary videos), but they can also be potentially estimated from the background image.

3.1 Hard Shadow Renderer in 2D Image

This section introduces our novel hard shadow rendering method based on the following assumptions: (1) images are upright, and the vertical lines are parallel. This corresponds to the one-point perspective or the two-point perspective, which is very common, and (2) the light source is a point light and is always above ground. Note that if the first assumption does not exactly hold, the generated hard shadow will be slightly distorted, but still a good approximation.

A simple example of the projective geometry following our assumptions is shown in Fig. 4. Given an object $A'B'$ that stands vertically on the ground and a point light source P', the object's shadow is then cast to $B'C'$. Given an image

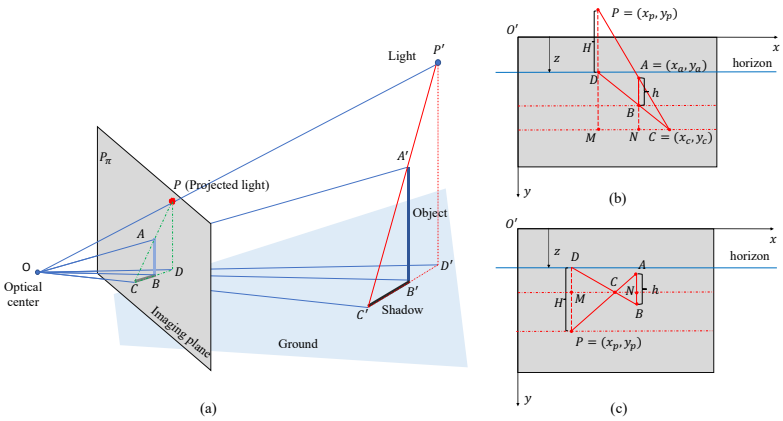

Fig. 4. Hard shadow renderer using Pixel Height representation. (a) shows the camera model, where $A'B'$ is the object standing upright on the ground, P' is the point light source and C' is the shadow point of A'. (b) and (c) shows two typical cases of the projection of the light, the object, and its shadow on the image plane. D is the perpendicular feet of P. The intersection point C of DB and PA is a projection of the shadow point C'. The 3D collinear property in the shadow geometry is preserved after being projected to the image plane.

plane, the light source, the object, and the shadow are projected to P, AB, and BC, respectively. The point D' is the perpendicular footpoint of the light, which is projected to D. Note that P', A' and C' are always collinear; and C', B' and D' are always co-linear. Thus, the projections of these points are also collinear in the image plane. For a non-planar shadow receiver, *e.g.* a wall, a similar collinear condition still holds except that the shadow point C' will be above the ground, and it will have its footpoint. For simplicity, we study the special case where the shadow receiver is the ground plane in the following. A more general formulation can be found in the supplementary material.

Fig. 4 (b) and (c) show the image plane and the relevant variables. We define the upper left corner of the image as the origin of the coordinate system. The light P and its projected perpendicular footpoint D are located at (x_p, y_p) and $(x_p, y_p + H)$ respectively, where H is the pixel distance between P and its footpoint, and we call it the *Pixel Height* of the light. Similarly, the object point A, its footpoint B and its shadow point C are located at (x_a, y_a), $(x_a, y_a + h)$ and (x_c, y_c), where h is the Pixel Height of A.

According to the triangle similarity in Fig. 4-(b) and Fig. 4-(c), we have

$$\frac{h}{H} = \frac{CN}{CM} = \frac{x_c - x_a}{x_c - x_p} = \frac{AN}{PM} = \frac{y_c - y_a}{y_c - y_p}. \tag{1}$$

The shadow point C can be derived from (x_a, y_a, h) and (x_p, y_p, H) by

$$C = [x_c, y_c] = \frac{1}{H - h} \left[Hx_a - hx_p, Hy_a - hy_p \right]. \tag{2}$$

Note that H may take a positive or a negative value. A negative value of H indicates that the light is behind the camera, and the shadow will be cast away from the camera (see Fig. 4-(c)). Note that the derived C may not exist. For example, when $h > H > 0$, the ray $\overrightarrow{PA}$ will not intersect with the ground. In this case, the derived C is actually the ground intersection point in the opposite direction of the ray. There is a special case when the light is infinity, and its footpoint is on the horizon. Let Z denote the y coordinate of the horizon. In this case, we can replace H with $Z - y_p$ in Eq. 2, and control the perspective of the shadow using the horizon line (see Fig. 9).

The above formulation describes how the shadow geometry is derived for our Pixel Height representation. For generic scenarios, the Pixel Height map of the shadow receiver needs to be provided, and the shadow map can be calculated by checking the collinear conditions similar to the ones mentioned above. We implemented the rendering algorithm for generic shadow receivers using CUDA. Please refer to supplementary materials for details. The visibility of the pixels on the shadow receiver can be computed in 20 ms for a 512×512 image.

3.2 Pixel Height Map Estimation

A Pixel Height map is a 2.5D representation similar to a depth map. Different from the depth map, the Pixel Height map uses the ground plane as a world frame reference to locate the object. It captures the object-ground relation so that the contact points and the uprightness of the object are explicitly enforced. In addition, Pixel Height map is measurable in the image space and can be annotated manually. In contrast, traditional 2.5D representations like depth are challenging to annotate from a single image. Objects reconstructed from a depth map can also be tilted if the camera intrinsic parameters are unknown.

The proposed Pixel Height representation is essential to the hard shadow rendering in a 2D image, and can be useful in other applications as well. In this section, we propose several methods to obtain the Pixel Height map.

Calculated from 3D geometries. Given a 3D geometry and camera parameters, the Pixel Height map can be computed by calculating the projection of the distance between each projected point and its footpoint. Fig. 2-(b) illustrates the rendererd RGB image and its Pixel Height map. With an accurate Pixel Height map, the proposed approach can render realistic soft shadows in real-time and generate visually comparable results with the renderings from a physics-based renderer (see Fig. 7).

Labeled from 2D images. The Pixel Height could also be annotated from a real RGB image by experienced annotators. Annotators are required to label sparse points on the object masks. For each point, its perpendicular footpoint on the ground is annotated. Thus, the Pixel Height could be calculated by the distance along the y-axis. Bi-linear interpolation is employed to get the dense Pixel Height map for the object of interest. Although the interpolation method is not physically correct, the generated hard shadows with the interpolated dense Pixel Height maps are perceptually pleasing. Fig. 2-(c) illustrates the sparsely labeled RGB images and interpolated Pixel Height maps.

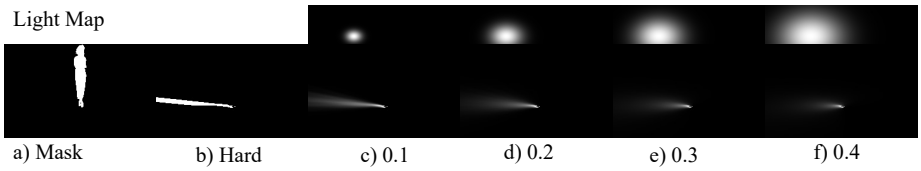

Fig. 5. The training set for the soft shadow generator. a) the mask of the object. b) the hard shadow of the object. Figures c)-f) show the soft shadows and visualized light maps at different levels of softness. Softness models the size of the light source.

Estimated from 2D images. Similar to the monocular depth estimation [33], estimating the Pixel Height from a single view image is an ill-posed problem. We verify that the Pixel Height could be estimated from a single view image. We propose a neural network for estimating humans' Pixel Height. The input to the network is the concatenation of the foreground image, the object mask, and a Y-Coordinate Map (YCM). We normalize the YCM by setting the lowest point in the object mask to be zero. Pixel Height map estimation is a high-level prediction problem, and the network should encode global information to get a better understanding of the geometry of the object. We employ an off-the-shelf transformer backbone, Mix Transformer encoder (MiT) [42]. A simple decoder merges features from different scales. The network's output is a one-channel Pixel Height map. For the training, We minimize the mean square error for each pixel inside the object mask between the prediction and the ground truth Pixel Height map. A total variation loss is used to regularize the prediction.

We use a synthetic dataset consisting of $60K$ renderings of 3D human models with various poses. We name this dataset *Synthetic60K*. To improve the model's generalization on real images, $1,500$ real images are collected and sparsely annotated to build a benchmark named *Real1500*. We used $1,000$ (500) images as the training (validation set). The ground truth Pixel Height of Synthetic60K and Real1500 are generated based on the methods described earlier in this section. We merge the *Synthetic60K* and the *Real1500* training set to train a Pixel Height Estimation Network (HENet). Each mini-batch is evenly sampled from the the two datasets. More implementation details about the training are in the supplementary materials.

3.3 Soft Shadow Generator

With the Pixel Height map and the proposed hard shadow renderer, a hard shadow map can be generated given a point light position in the image. To add softness to the shadow, we train a soft shadow generator to create the effect of an area light and control the softness based on user input.

Data generation. The soft shadows are generated following the pipeline in the Soft Shadow Network (SSN) [36]. They divided the location of the light into grids, and then randomly sampled an environment light map based on a 2D Gaussian distribution at one random grid. Soft shadow bases are generated

by merging the hard shadows of a local patch for each grid. The hard shadow is generated by a GPU-based render with 3D models. Soft shadow based on the environment lightmap will be the weighted sum for the shadow bases. SSN enforces the network to learn a mapping from the lightmap and the soft shadow, which is very complicated. Different from their method, we want to render a soft shadow based on the hard shadow and a pre-define softness. To get our training samples, for each soft shadow and its paired environment lightmap, we find the corresponding hard shadow and the softness from the lightmap. The hard shadow is rendered with a given point light, which locates at the center of the area light. The softness is defined as the size of the Gaussian which is used to generate the lightmap. Fig. 5 illustrates that an environment lightmap can be represented as a hard shadow and a softness value. Finally, we get our training triplet (hard shadow, soft shadow, and softness) on the fly during training.

Network structure. The soft shadow generator (SSG) is a variant of the U-Net. Similar to the shadow render in SSN [36], the encoder of the network is composed of a series of 3×3 convolution layers. Skip connections are employed to capture the low-level features. SSN [36] estimates the soft shadow based on the object mask and the environment lightmap. It requires the network to learn a complex mapping between the object shadow and the light source. In contrast, we use a physical model to render the hard shadow in 2D space (described in Sec. 3.1). The input of the encoder network is the concatenation of the mask and the hard shadow. To inject a softness control into the network, we uniformly discretize the continuous softness into multiple bins in the log space and then sampled a soft Gaussian distribution on these bins following [3,44]. Thus, a softness value can be represented by an embedding with a fixed dimension. Following [19], the adaptive instance normalization is then employed in the decoder to take the softness embedding for the softness control. The training details follow [36].

4 Experiments and Evaluation

Our system consists of several key components: the Height Estimation Network (HENet), the Hard Shadow Renderer (HSR), and the Soft Shadow Generator (SSG). We first validate the effectiveness of HENet on human images and then the soft shadow quality from SSG. Finally, a user study and qualitative comparisons are conducted to evaluate our full system on real images.

HENet is trained to predict the Pixel Height for human images in our current implementation. In the following experiments, unless otherwise specified, the Pixel Height maps for humans are predicted by our HENet. The Pixel Height maps for other general objects are manually labeled. The average labeling time for one object is about two minutes.

4.1 Evaluation of HENet

We conduct ablation studies on the proposed components to estimate the Pixel Height map. The network is trained on the merged dataset of Synthetic60K and

Table 1. Effectiveness of each components in predicting the Pixel Height. YCM: using normalized Y-coordinate Map as input. Real: training on the real and synthetic data. ℓ_{tv}: training with the total variation loss. The metrics are evaluated on the sparse points labelled by annotators on natural images. Base: Employing Y-Coordinate Map as the Pixel Height.

	YCM	Real	ℓ_{tv}	Abs ↓	rel ↓
Base				10.84	3.64
a	✓	✓		6.12	2.01
b		✓	✓	6.04	1.98
c	✓		✓	7.05	2.34
d	✓	✓	✓	5.92	1.94

Fig. 6. For the input image without shadow (a) we use the Y-Coordinate Map to replace the Pixel Height map in our system (b). It can not handle the foot contact with the ground properly. Based on our predicted Pixel Height map, the shadow in the foot contact area is more realistic (c).

the Real1500 training set. The results are evaluated on the sparse points labeled by annotators of the Real1500 validation set. In Tab. 1, the evaluation results show that adding the Y-Coordinate Map (YCM) and using the total variation loss (ℓ_{tv}) can both reduce the error. Moreover, training on the merged dataset can significantly improve the model's generalization ability, reducing the relative error from 2.34% to 1.82%. We also list the evaluation results of the baseline to verify that HENet does not just learn a trivial identity mapping of the YCM. As shown in Fig. 6, using the YCM instead of the Pixel Height map can not generate the correct shadow in the foot contact area.

4.2 Evaluation of SSG

Instead of implicitly learning a mapping from the light source to a shadow [36], our SSG only focuses on adding softness to the hard shadow based on a controllable input scalar. We build an evaluation benchmark to evaluate the model. We used 20 new assets of 3D models that have no overlap with the training set, and they are collected from the Internet. For each new asset, we uniformly sample 4×66 positions of the light source and divide them into three groups on average based on the length of the generated shadows, named as 'Short', 'Medium', and

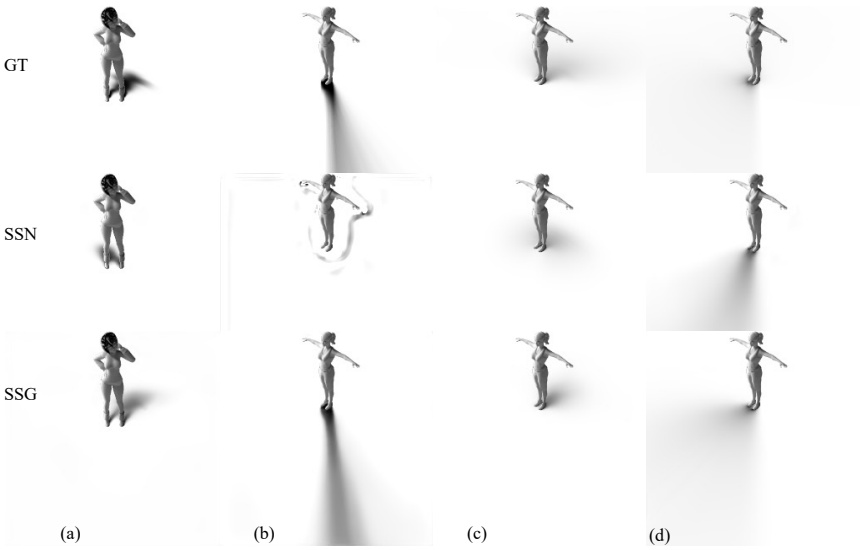

Fig. 7. Comparison between our proposed SSG, SSN [36] and synthetic ground truth based on 3D models. (a) The direction of the shadow is more accurate as a hard shadow is given. (b) SSN may fail to generate a long hard shadow. (c-d) Both methods perform well on soft shadows. The shadows from SSG are comparable to the physics-based renderer.

'Long'. For each position, 9 types of softness are sampled. The evaluation benchmark is also divided into 'Soft', 'Medium', and 'Hard' based on the softness. The ground truth shadows are rendered with Mitsuba with 3D models. The evaluation metrics include the average of the pixel-level absolute error (Abs) and the zero normalized cross-correlation (ZNCC). The first one evaluates the pixel-level error, and the second one considers the similarity of the shape.

Results. The evaluation results are shown in Tab. 2. On average, the proposed SSG outperforms SSN on both evaluation metrics, improving Abs and ZNCC by 27% and 112%, respectively. For hard shadows, the proposed SSG reduces the Abs error of SSN from 0.039 to 0.025, and increases the ZNCC from 0.198 to 0.761. The SSN performs slightly better on 'Short' shadows according to the Abs, indicating that directly learning the mapping has some advantage in those cases. Still, it gets unstable for generating long and hard shadows due to the lack of model capacity in long-distance geometric modeling. Samples of visualization results are shown in Fig. 7. SSN may produce inaccurate direction of the shadow based on the given lightmap as shown in Fig. 7-(a). The errors on harder shadows are more apparent, and people are less sensitive to the difference in soft shadows.

Table 2. Quantitative evaluation. Abs: pixel-level absolute error. ZNCC: zero normalized cross-correlation. The ground truth shadows are rendered with Mitsuba given 3D shapes. On average, our proposed SSG outperforms SSN [36]. The quality of the 'Long' shadow and the 'Hard' shadow is improved with a larger margin.

	Mean Abs			Mean ZNCC		
SSN [36]	0.033			0.370		
Ours	**0.024**			**0.788**		
	Abs ↓			ZNCC ↑		
Length	Long	Medium	Short	Long	Medium	Short
SSN [36]	0.041	0.029	**0.031**	0.330	0.311	0.437
Ours	**0.028**	**0.012**	0.033	**0.743**	**0.883**	**0.725**
Softness	Hard	Medium	Soft	Hard	Medium	Soft
SSN [36]	0.039	0.034	0.024	0.198	0.336	0.606
Ours	**0.025**	**0.028**	**0.017**	**0.761**	**0.779**	**0.834**

Table 3. User study on natural images. In a 2AFC study the users chose the more realistic image from a pair. The results indicate that 74% of users perceived the shadows generated by our algorithm as more realistic.

Rate	Length			Softness			Mean
	Long	Medium	Short	Hard	Medium	Soft	
SSN	0.27	0.22	0.35	0.20	0.29	0.31	0.26
SSG	**0.73**	**0.78**	**0.65**	**0.80**	**0.71**	**0.69**	**0.74**

4.3 Full System Evaluation

We qualitatively evaluated our entire system on natural images. Specifically, we performed a user study, where we asked human subjects to compare the perceived visual quality of the generated shadows from our method and SNN.

We conducted a user study on the shadows generated for 2D natural images. For SSN, the shadow is rendered with the cutout of the object and an interactive light source. For our method, the shadow is rendered with an interactive interface with shadow position and softness controls (see the video demo in our supplementary materials). We prepared 24 shadow pairs mimicking the effect of different lengths and softness. We have shown pairs of images in random order and random position (left-right) to 50 users (80% males and 20% females) and asked the participants which of the two images looks more realistic. Tab. 3 shows that 74% of the users perceived the shadows rendered with our method as more realistic, especially for long and hard shadows (see Fig. 8).

5 Discussions

Controllability. Our system based on Pixel Height improves the controllability of the shadow synthesis. A demo video of our simple GUI is in the *supplementary materials*, enabling users to change the direction of the shadow by a simple click on the preferred position, similar to the method presented in [27]. Our method also allows the control of the shadow shape using the horizon line, mimicking

Fig. 8. Samples images from our user study. Our results have clearer shapes on hard shadows and less artifact on long shadows compared with SSN [36].

Fig. 9. Controllability. We can mimic the shadow effect for different camera poses by changing the horizon line.

the perspective effect from a camera (see Fig. 9). The softness is controlled by a slider. Fig. 10 shows some example results generated from our GUI using height maps obtained from different approaches. Fig. 11 shows a case where the object's shadow is cast on a complex shadow receiver with a floating effect. Our method can also be applied on animated objects. Please check out our supplementary materials for more examples.

Potential applications of the Pixel Height map. Pixel Height map can also be used to generate reflection effects. A slightly modified checking condition is used to compute the correspondence between a point and its reflection on the ground. We demonstrate this in the supplementary material.

6 Conclusion

We proposed an approach for generating controllable perceptually plausible shadows based on the Pixel Height map. The new geometry representation, Pixel Height map, encodes the correlations among objects shape, camera pose, and the ground. It can be directly labeled or estimated from 2D images. The position and softness is controlled in an easy interactive way. Qualitative and quantitative comparisons demonstrate the results and generalization ability of the proposed

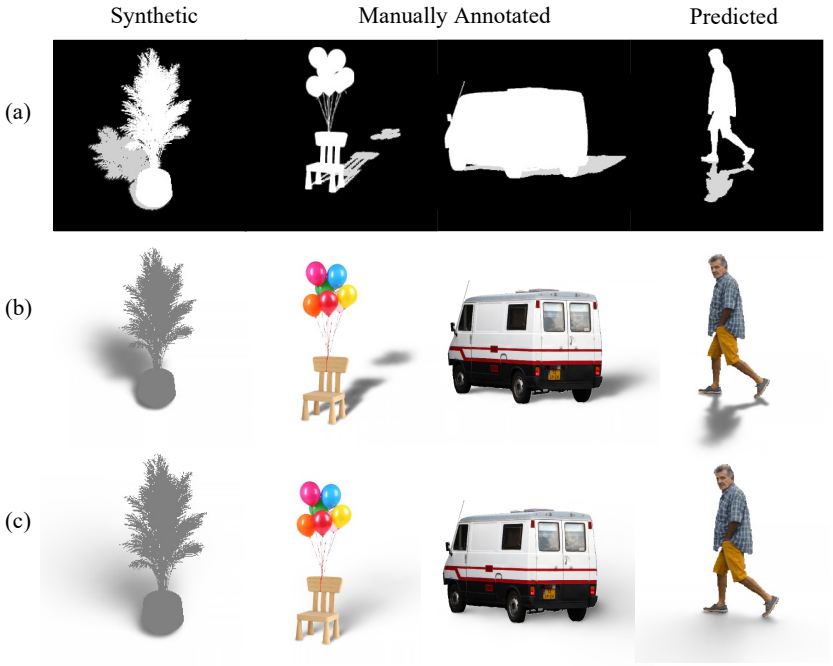

Fig. 10. Given a Pixel Height map, our method can produce realistic shadows with desired position and softness. The Pixel Height map can be calculated from 3D models, manually annotated or predicted by HENet. (a) Hard shadow mask. (b) Softness is 0.05. (c) Softness is 0.4.

Fig. 11. Shadow generation for a floating object and a complex shadow receiver. (a) and (c) show that hard shadows can be cast on a non-planar shadow receiver using the Pixel Height map of background as input. Our hard shadow renderer can also render shadows for floating objects by simply adding a shift value to the Pixel Height map of the object. (b) and (d) show the corresponding soft shadow generated by SSG.

method outperforms previous deep learning-based shadow generation methods. However, our Pixel Height map representation only considers the frontal surface of the object. A thickness is worth future exploration to address the problem.

Acknowledgment Most of the work was done during Yifan and Yichen's internship at Adobe. This work was also supported by a UKRI Future Leaders Fellowship [grant number G104084]. We thank Dr. Zhi Tian for the discussions.

References

1. Annen, T., Dong, Z., Mertens, T., Bekaert, P., Seidel, H.P., Kautz, J.: Real-time, all-frequency shadows in dynamic scenes. ACM TOG **27**(3), 1–8 (2008) 3
2. Assarsson, U., Akenine-Möller, T.: A geometry-based soft shadow volume algorithm using graphics hardware. ACM TOG **22**(3), 511–520 (2003) 3
3. Cao, Y., Wu, Z., Shen, C.: Estimating depth from monocular images as classification using deep fully convolutional residual networks. IEEE TCSVT **28**(11), 3174–3182 (2017) 9
4. Chan, E., Durand, F.: Rendering fake soft shadows with smoothies. In: Rendering Techniques. pp. 208–218. Citeseer (2003) 3
5. Cook, R.L., Porter, T., Carpenter, L.: Distributed ray tracing. In: ACM SIGGRAPH. pp. 137–145 (1984) 4
6. Crow, F.C.: Shadow algorithms for computer graphics. ACM SIGGRAPH **11**(2), 242–248 (1977) 1, 4
7. Donnelly, W., Lauritzen, A.: Variance shadow maps. In: Proceedings of the 2006 symposium on Interactive 3D graphics and games. pp. 161–165 (2006) 3
8. Fernando, R.: Percentage-closer soft shadows. In: ACM SIGGRAPH, pp. 35–es (2005) 3
9. Franke, T.A.: Delta voxel cone tracing. In: 2014 IEEE International Symposium on Mixed and Augmented Reality (ISMAR). pp. 39–44. IEEE (2014) 4
10. Fuchs, H., Goldfeather, J., Hultquist, J.P., Spach, S., Austin, J.D., Brooks Jr, F.P., Eyles, J.G., Poulton, J.: Fast spheres, shadows, textures, transparencies, and imgage enhancements in pixel-planes. ACM SIGGRAPH **19**(3), 111–120 (1985) 4
11. Guennebaud, G., Barthe, L., Paulin, M.: Real-time soft shadow mapping by backprojection. In: Rendering techniques. pp. 227–234 (2006) 3
12. Guennebaud, G., Barthe, L., Paulin, M.: High-quality adaptive soft shadow mapping. In: Computer Graphics Forum. vol. 26, pp. 525–533. Wiley Online Library (2007) 3
13. Heitz, E., Dupuy, J., Hill, S., Neubelt, D.: Real-time polygonal-light shading with linearly transformed cosines. ACM TOG **35**(4), 1–8 (2016) 4
14. Hong, Y., Niu, L., Zhang, J., Zhang, L.: Shadow generation for composite image in real-world scenes. arXiv preprint arXiv:2104.10338 (2021) 1, 4
15. Hu, X., Jiang, Y., Fu, C.W., Heng, P.A.: Mask-shadowgan: Learning to remove shadows from unpaired data. In: ICCV. pp. 2472–2481 (2019) 4
16. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: CVPR. pp. 1125–1134 (2017) 4
17. Jiang, Y., Zhang, H., Zhang, J., Wang, Y., Lin, Z., Sunkavalli, K., Chen, S., Amirghodsi, S., Kong, S., Wang, Z.: Ssh: A self-supervised framework for image harmonization. In: ICCV. pp. 4832–4841 (2021) 1
18. Kajiya, J.T.: The rendering equation. In: ACM SIGGRAPH. pp. 143–150 (1986) 4
19. Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. In: CVPR. pp. 4401–4410 (2019) 9
20. Li, Z., Snavely, N.: Megadepth: Learning single-view depth prediction from internet photos. In: CVPR. pp. 2041–2050 (2018) 4
21. Liu, D., Long, C., Zhang, H., Yu, H., Dong, X., Xiao, C.: Arshadowgan: Shadow generative adversarial network for augmented reality in single light scenes. In: CVPR. pp. 8139–8148 (2020) 1, 4

22. Liu, Y., Qin, Z., Wan, T., Luo, Z.: Auto-painter: Cartoon image generation from sketch by using conditional wasserstein generative adversarial networks. Neurocomputing **311**, 78–87 (2018) 4
23. Lu, H., Dai, Y., Shen, C., Xu, S.: Indices matter: Learning to index for deep image matting. In: ICCV. pp. 3266–3275 (2019) 1
24. Mehta, S.U., Wang, B., Ramamoorthi, R.: Axis-aligned filtering for interactive sampled soft shadows. ACM TOG **31**(6), 1–10 (2012) 4
25. Ng, R., Ramamoorthi, R., Hanrahan, P.: All-frequency shadows using non-linear wavelet lighting approximation. In: ACM SIGGRAPH, pp. 376–381 (2003) 4
26. Öztireli, A.C.: Integration with stochastic point processes. ACM TOG **35**(5), 1–16 (2016) 1, 4
27. Pellacini, F., Tole, P., Greenberg, D.P.: A user interface for interactive cinematic shadow design. ACM TOG **21**(3), 563–566 (2002) 12
28. Reeves, W.T., Salesin, D.H., Cook, R.L.: Rendering antialiased shadows with depth maps. In: ACM SIGGRAPH. pp. 283–291 (1987) 3
29. Remondino, F.: From point cloud to surface: the modeling and visualization problem. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences **34** (2003) 4
30. Ren, Z., Wang, R., Snyder, J., Zhou, K., Liu, X., Sun, B., Sloan, P.P., Bao, H., Peng, Q., Guo, B.: Real-time soft shadows in dynamic scenes using spherical harmonic exponentiation. In: ACM SIGGRAPH, pp. 977–986 (2006) 4
31. Saito, S., , Huang, Z., Natsume, R., Morishima, S., Kanazawa, A., Li, H.: Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. ICCV (2019) 4
32. Saito, S., Simon, T., Saragih, J., Joo, H.: Pifuhd: Multi-level pixel-aligned implicit function for high-resolution 3d human digitization. In: CVPR (June 2020) 4
33. Saxena, A., Chung, S.H., Ng, A.Y., et al.: Learning depth from single monocular images. In: NeurIPS. vol. 18, pp. 1–8 (2005) 4, 8
34. Schwarz, M., Stamminger, M.: Bitmask soft shadows. In: Computer Graphics Forum. vol. 26, pp. 515–524. Wiley Online Library (2007) 3
35. Sen, P., Cammarano, M., Hanrahan, P.: Shadow silhouette maps. ACM TOG **22**(3), 521–526 (2003) 3
36. Sheng, Y., Zhang, J., Benes, B.: SSN: Soft shadow network for image compositing. In: CVPR. pp. 4380–4390 (2021) 2, 4, 8, 9, 10, 11, 12, 13
37. Sillion, F.X., Arvo, J.R., Westin, S.H., Greenberg, D.P.: A global illumination solution for general reflectance distributions. In: ACM SIGGRAPH. pp. 187–196 (1991) 4
38. Soler, C., Sillion, F.X.: Fast calculation of soft shadow textures using convolution. In: ACM SIGGRAPH. pp. 321–332 (1998) 3
39. Wang, Y., Curless, B.L., Seitz, S.M.: People as scene probes. In: European Conference on Computer Vision. pp. 438–454. Springer (2020) 4
40. Westin, S.H., Arvo, J.R., Torrance, K.E.: Predicting reflectance functions from complex surfaces. In: ACM SIGGRAPH. pp. 255–264 (1992) 4
41. Williams, L.: Casting curved shadows on curved surfaces. In: ACM SIGGRAPH. pp. 270–274 (1978) 3
42. Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J.M., Luo, P.: Segformer: Simple and efficient design for semantic segmentation with transformers. NeurIPS (2021) 8
43. Yin, W., Liu, Y., Shen, C.: Virtual normal: Enforcing geometric constraints for accurate and robust depth prediction. IEEE TPAMI (2021) 4

44. Yin, W., Liu, Y., Shen, C., Yan, Y.: Enforcing geometric constraints of virtual normal for depth prediction. In: ICCV. pp. 5684–5693 (2019) 4, 9
45. Zhang, S., Liang, R., Wang, M.: Shadowgan: Shadow synthesis for virtual objects with conditional adversarial networks. Computational Visual Media **5**(1), 8 (2019) 1, 4