# Supplementary: 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

## 1 More Results

**Video result.** We provide a video in the supplementary materials to show more results. The video includes demos of our controllable shadow generation for single or multiple objects, comparisons with previous works, improvements brought by our Pixel Height representation, potential future applications, etc.

**Interactive Pixel Height map annotation.** The Pixel Height maps can be interactively modified by users during the shadow generation in real time. As shown in Fig. 1, users can label the Pixel Height for a point by clicking its foot point, and the height map gets updated accordingly. Users can correct mistakes in the Pixel Height map or add more labeled points to refine it. For example, the Pixel Height of the arm in the second image is not accurate, leading to shadow distortion. After adding more annotations, the shape of the shadow looks more real.

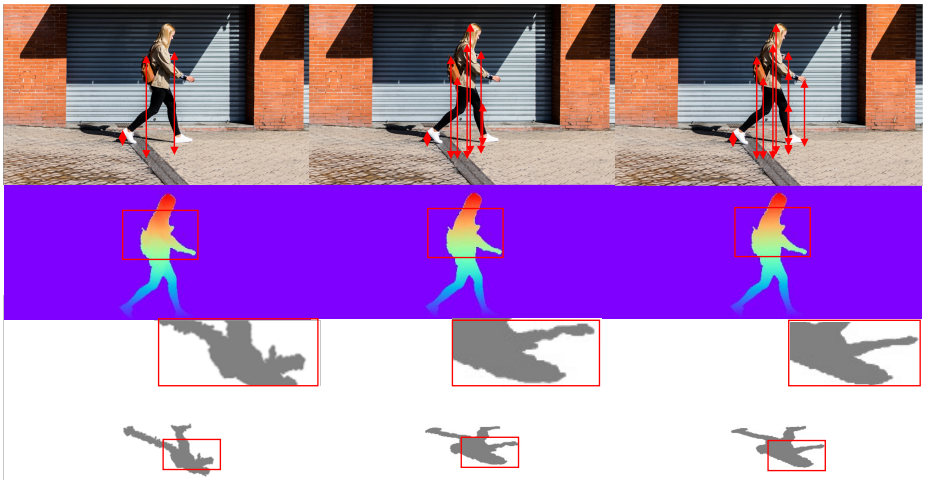

**Fig. 1.** The Pixel Height map becomes more accurate during interactive labelling. The shape of the hard shadow is more realistic with more accurate Pixel Height maps.

**Controllability.** We show more samples on controlling the light position, horizon line and the softness in Fig. 2. Our method can also control the shadows to be consistent for multi objects. The example can be found in the provided video.

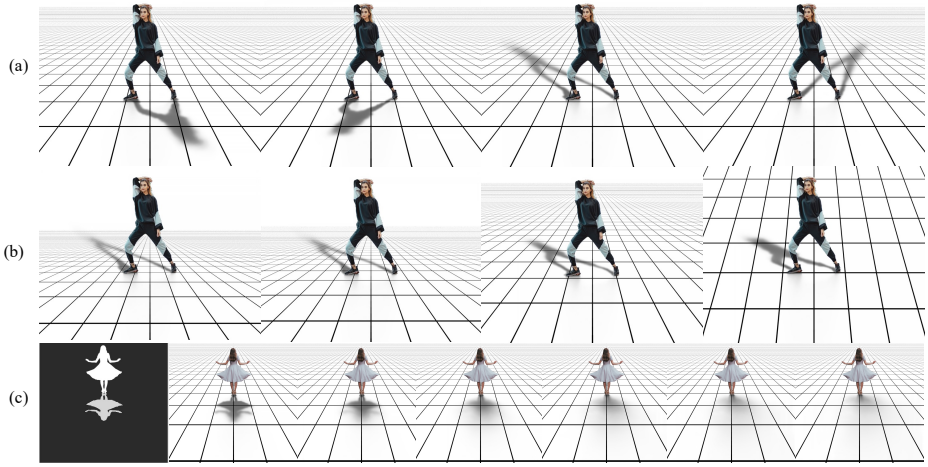


**Fig. 2. Controlbility** (a) Control of light position. (b) Control of horizon line. (c) Control of softness.

**Comparison results between SSN and Ours.** In Fig. 3, we show more comparison results based on different softness between SSN [1] and Ours. SSN can not produce a hard shadow with clear shapes. We also attach a video comparison in the provided video. Controlling the lightmap is not as convenient as our proposed method for generating specific shadows in a desired position. We show some examples in the provided video.

**More results on various classes** In Fig. 4, we show more results with the proposed HSR and SSG. The way of getting Pixel Height Maps is flexible.

## 2 Implementation Details

### 2.1 Hard shadow renderer on general shadow receiver

Casting a shadow on a ground planner is a special case, which assumes the Pixel Height for the shadow receiver is zero. The shadows can also be generated on a non-planner shadow receiver, e.g., wall, stairs, or rocks if the Pixel Height of the shadow receiver could be obtained. In this section, we show the detail of how to render a hard shadow on a general shadow receiver based on the proposed Pixel Height.

The collinear property is preserved after projection under the pinhole camera model assumption(See Fig. 5). We use this property to cast shadows on general

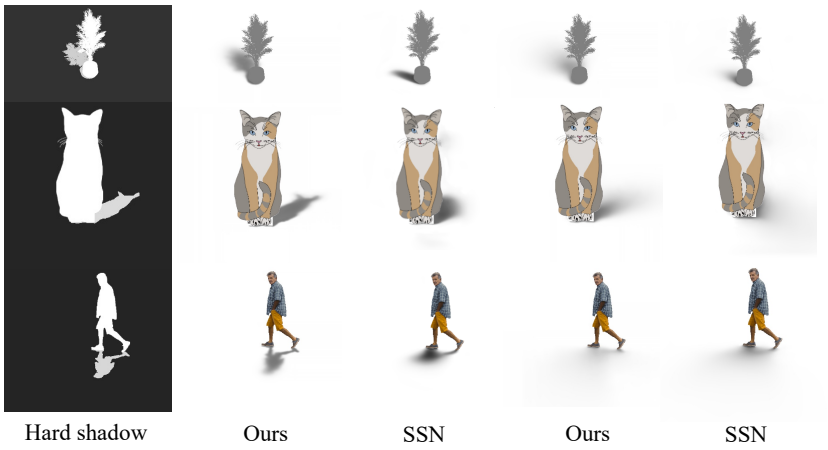


**Fig. 3.** Comparison between SSN and our methods. Our method can generate shadows with clear shape.

shadow receivers in our Pixel Height representation. In detail, as shown in Fig. 5, given the light $P$ in Pixel Height representation, we can check if the background point $C$ is occluded by any foreground point, i.e. is there any point of the foreground object on the line $PC$? If there is a point, e.g. $A$ in Fig. 5 on the line $PC$, the point $C$ is in shadow, otherwise it is lit.

In practice, the Pixel Height map based hard shadow rendering algorithm can be implemented in either forward scattering or back tracing. Forward scattering is simply scattering the occluder's pixels to the shadow receiver. But hard shadow rendered in this way will have holes due to the discrete representation. Instead, we use back tracing Pixel Height based hard shadow rendering algorithm(BT-PHSR). BT-PHSR iterates over all the background pixels and checks if the background pixel is blocked by any point of the foreground object using the collinear property. Implementation details are shown in the Algorithm 1:

### 2.2 Pixel Height Estimation Network

A pixel Height Estimation Network (HENet) is then proposed to estimate the Pixel Height from a single RGB image. In this section, we provide more details regardig the data generation and the training details.

**Data generation.** *Synthetic60K*: We collect 2442 3D models of persons. Each model has around 1 141 poses. For each pose, we set 3 camera poses to capture images from different views. Finally, we get around $60K$ samples. Each sample contains a captured RGB image, a mask for the object, and the pixel height for each pixel inside the mask. *Real1500*: The training samples has the same format as the samples in Synthetic60K. 1000 / 500 images are used as the training/ validation set. Figure 6 shows samples of synthetic data and the labeled data.

**Algorithm 1:** BT-PHSR

**Input:** Shadow occluder Pixel Height map $H$, light position $l_{xy}$ and height $l_h$
**Output:** Hard shadow $S$
Initialize $S$ with 1
$h, w$ = height and width of $H$
**for** $i = 0$ **to** $w$ **do**
  **for** $j = 0$ **to** $h$ **do**
    $p_{ij} = (i, j)$
    **forall** $p$ *on the line* $l_{xy} - p_{ij}$ **do**
      $i', j' = p$
      **if** $H(i'j')$ *does not have value* **then**
        Continue
      **else**
        $h' = H(i'j')$
        $h = l_h \cdot (p - p_{ij})/(l_{xy} - p_{ij})$
        **if** $|h - h'| < \epsilon$ **then**
          S(i, j) = 0
          Break
        **end**
      **end**
    **end**
  **end**
**end**

**Network.** The network structure is shown in Figure 8. As shown in Figure 8, we first insert a convolutional-based adaptor to modify the input into three channels. The embedding module divided the input into patches of size $4 \times 4$, which is more suitable for dense prediction tasks. The Adaptor modules transfer the features from multi-level to the same channels, and then all the adapted features are concatenated together for the final pixel height estimation. The output of the network is the single channel heatmap.
**Training.** The network is trained with an initial learning rate of 0.001 with AdamW for 160000 steps with 24 images in each mini batch. The image crop is $512 \times 512$. For each mini-batch we randomly sample 12 images from the Real1500 and 12 images from the Synthetic60K. Training on real images help the network to generalize better as shown in Fig. 7.

### 2.3 Soft Shadow Generator

We train our SSG on the same dataset as the SSN [1] using a U-net structure with similar computational costs. The main difference is that the input environment lightmap is represented as a hard shadow and a softness. The network is trained for 60 epochs with an initial learning rate of 0.001. We train on $4\times$ Volta100 GPUs for 18 hours. It takes 15 ms to inference on an image with resolution of $512 \times 512$ on a single GTX 1080Ti GPU.

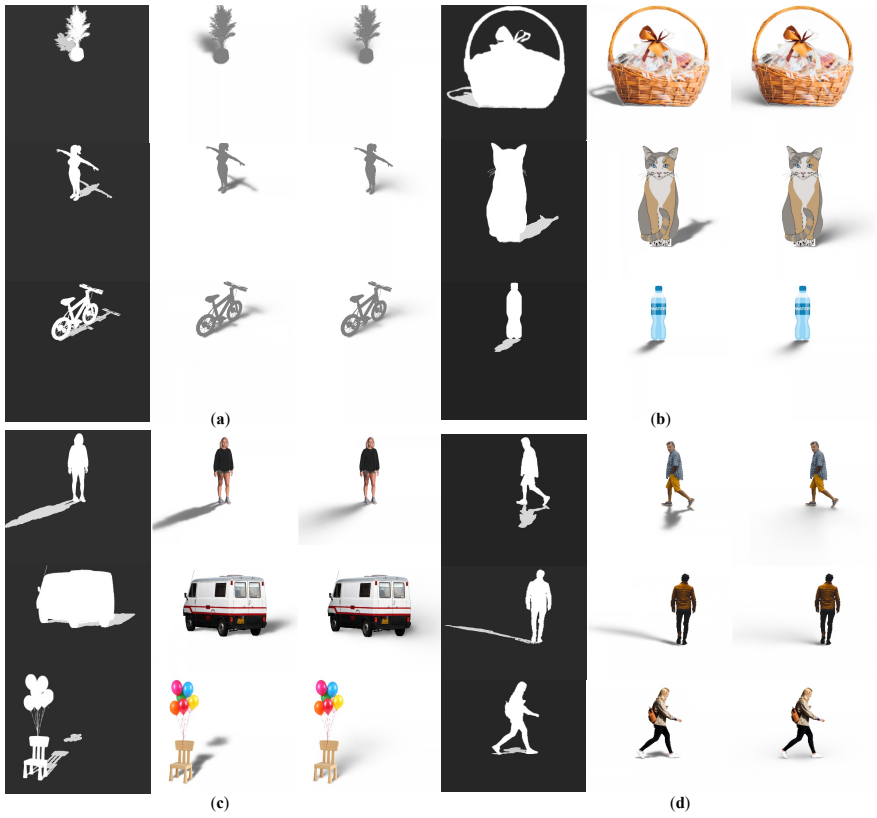


**Fig. 4.** Given a height map, our method can produce realistic shadows with desired position and softness. The method is robust on pixel height maps from different sources. (**a**) **Pixel height maps rendered from 3D models**: Our method can produce realistic shadows for objects with complicated geometry if accurate pixel height maps are given. (**b**)) **Pixel height from human annotation**: The pixel height can be sparsely annotated with grids. A dense pixel height map is further interpolated from the sparse annotations. (**c**) **Y-Coordinate Map**: For objects standing horizontally in the canvas, the Y-Coordinate Map can also mimic the effect of the Pixel Height Map. (**d Predicted pixel height**: The pixel height could be estimated from a single RGB image.

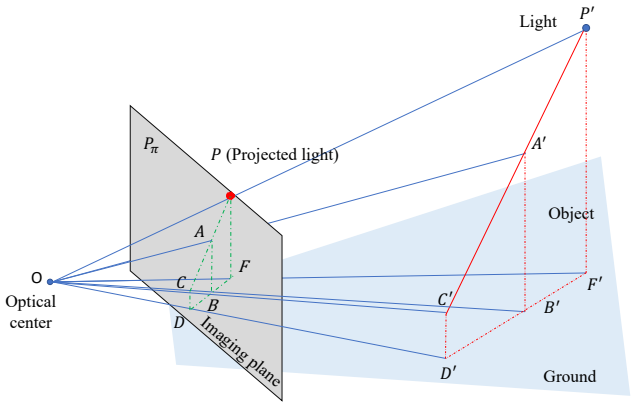


**Fig. 5. Hard shadow renderer using Pixel Height representation on general shadow receiver.** In general, $P'$ is a light position in 3D, $A'$ and $C'$ are two points in 3D, $P', A', C'$ are collinear. $F', B', D'$ are the corresponding foot points on the ground. $P, A, C, F, B, D$ are the corresponding projection points. Under pinhole camera model assumption, $PAC$ in 2D projection space are collinear if $P'A'C'$ in 3D are collinear.

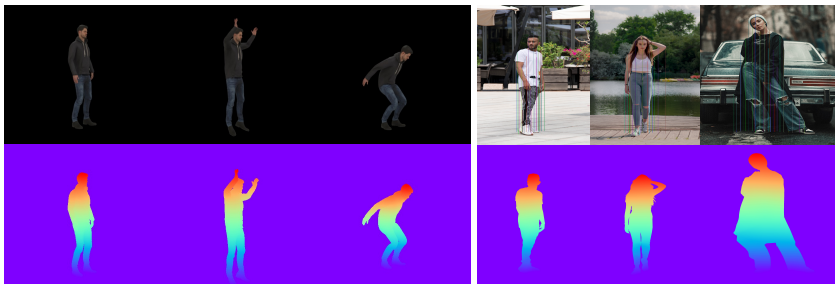


**Fig. 6. Samples from the pixel height estimation dataset.** (**a**) Synthetic60K: Samples with various poses from one 3D model. (**b**) Real1500: Samples of the sparse label effects and the dense interpolated Pixel Height map. Note that every Pixel Height map is divided by its max height value for the visualization purpose.

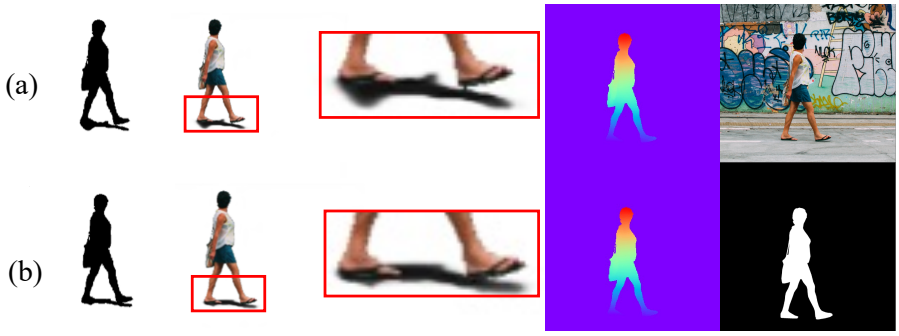


**Fig. 7.** The predicted Pixel Height maps and the corresponding shadows. (a) HENet Trained on Synthetic60K. (b) Training on the merged dataset. The proposed HENet can produce reasonable Pixel Height maps.

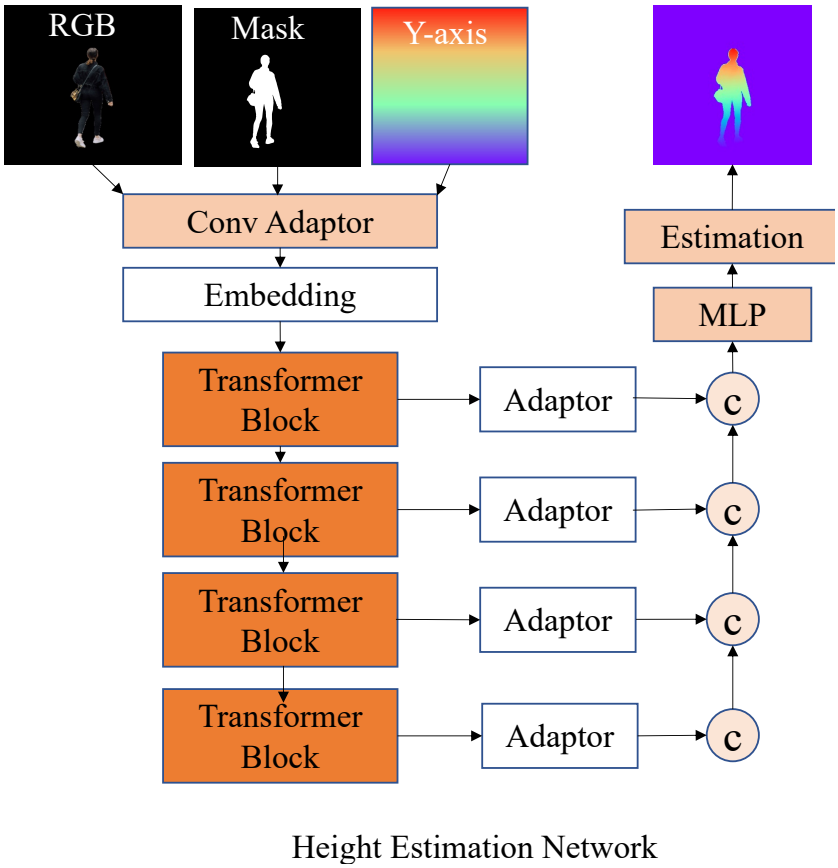


**Fig. 8.** The structure of the pixel Height Estimation Network (**HENet**). The HENet consists of an efficient transformer backbone and a multi-level decoder which merges the features from different scales.

## References

1. Sheng, Y., Zhang, J., Benes, B.: SSN: Soft shadow network for image compositing. In: CVPR. pp. 4380–4390 (2021) 2, 4