Computational Visual Media
https://doi.org/10.1007/s41095-019-0136-1 Vol. 5, No. 1, March 2019, 105–115
Research Article

ShadowGAN: Shadow synthesis for virtual objects with conditional adversarial networks

Shuyang Zhang[1] (✉), **Runze Liang**[2], **and Miao Wang**[3]

Abstract We introduce *ShadowGAN*, a generative adversarial network (GAN) for synthesizing shadows for virtual objects inserted in images. Given a target image containing several existing objects with shadows, and an input source object with a specified insertion position, the network generates a realistic shadow for the source object. The shadow is synthesized by a generator; using the proposed local adversarial and global adversarial discriminators, the synthetic shadow's appearance is locally realistic in shape, and globally consistent with other objects' shadows in terms of shadow direction and area. To overcome the lack of training data, we produced training samples based on public 3D models and rendering technology. Experimental results from a user study show that the synthetic shadowed results look natural and authentic.

Keywords shadow synthesis; deep learning; generative adversarial networks; image synthesis

1 Introduction

Inserting virtual objects into scenes has a wide range of applications in visual media, from movies, advertisements, and entertainment to virtual reality. Consistency of shadows between the original scene and the inserted object contributes greatly to the naturalness of the results. If no prior scene knowledge is provided, it requires much labor and expertise to make the scene look as realistic as possible, in a tedious photo or video editing process. Even an experienced editor spends much effort to produce convincing results using commercial editing software such as Adobe Photoshop. The difficulties in this process stem from the lack of accurate estimates of illumination and scene geometry.

In this paper, we address the shadow synthesis problem for virtual objects inserted in an image. Shadow synthesis can be implemented by use of rendering techniques, which require much information, such as illumination, scene models, rendering frameworks, etc. Other methods [1–4] synthesize shadows with approximately estimated illumination and reconstructed scene geometry. Such computations either require user interaction or precise tools, and yet are time-consuming.

We propose to solve this problem using a novel deep learning-based framework without explicit knowledge of scene geometry and illumination. We use a convolutional neural network to directly predict the shadow map for a virtually inserted object, given only the target scene image and the specified insertion position in the image domain. Specifically, we use a generative adversarial network (GAN) framework, where the generator G tries to produce outputs that cannot be distinguished from "real" results, while the local discriminator D^{L} and global discriminator D^{G} try to detect the generator's "fakes" from local and global perspectives, respectively. During training, the generator and discriminators compete until convergence. As a result, a real-type, single-channel shadow map is predicted, from which the edited result with a synthetic shadow can be generated by a simple pixel-wise original image multiplication. The input constraints to our ShadowGAN are few

1 University of Michigan, Ann Arbor, MI 48109 USA. E-mail: zhangshuyangmary@outlook.com (✉).
2 Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China. E-mail: liangrz15@mails.tsinghua.edu.cn.
3 State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing 100191, China. E-mail: miaow@buaa.edu.cn.
Manuscript received: 2018-12-28; accepted: 2019-02-05

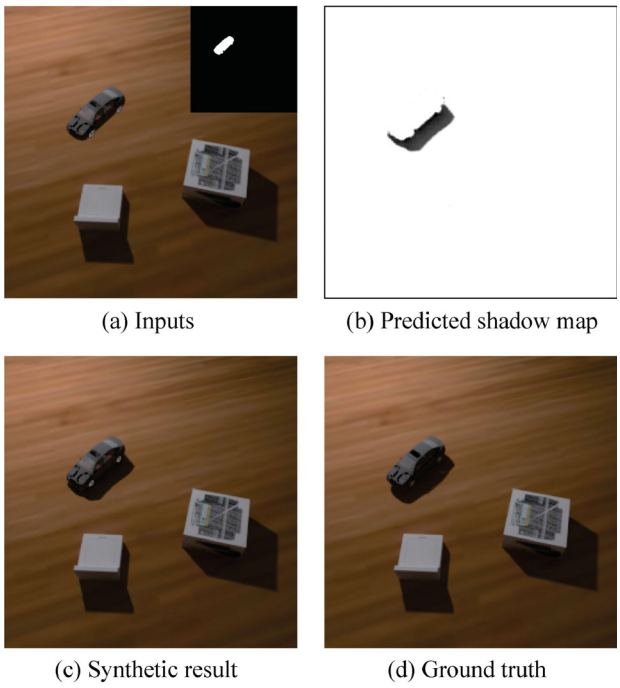

Fig. 1 Input and output of ShadowGAN. (a) Given an input target scene with original objects and a virtually inserted object (here, a toy car), as well as the object mask ((a) top-right), ShadowGAN predicts a shadow map (b) which can be used to synthesize the shadowed result (c) with a simple pixel-wise product operation. The ground truth result is shown in (d).

while the computational efficiency is high as only a simple feed-forward operation through the network is needed.

Our method works for an image of a static scene. We assume scene surfaces to be made of Lambertian materials and we do not model specular reflection or inter-reflections between surfaces in the scene. Despite these assumptions, we can produce plausible results. To summarize, the contributions of our work are:

- A convolutional neural network, ShadowGAN, which can synthesize shadows for virtually inserted objects in target images.
- A local–global conditional adversarial scheme for both shape and direction supervision in shadow synthesis.
- A practical dataset for shadow synthesis network training, produced using rendering techniques and public 3D models.

2 Related work

In this section, we discuss related prior work, mainly on shadow synthesis, shadow detection and removal, and image-to-image translation using generative adversarial networks.

2.1 Shadow synthesis

In image editing, knowledge of illumination and scene geometry is essential to achieving realistic shadow synthesis results. Previous methods have been proposed to recover such information from input images or videos. Intrinsic image decomposition algorithms aim to separate a single image I into a pixel-wise product of an albedo or reflectance layer R and a shading layer S [5–8]. The reflectance layer reveals how the material reflects incident light, and the shading layer accounts for illumination effects due to geometry, shadows, and inter-reflections. However, approaches based on pixel-wise illumination and reflectance maps are not effective enough to support complex editing operations such as object insertion. For visually plausible results, shadows must be carefully computed, which requires an analysis of scene geometry and lighting configuration in 3D space. The problem of estimating illumination from images, or *inverse lighting*, has been investigated. In Refs. [9, 10], illumination distributions in a scene from object shadows of known shapes are recovered. Khan et al. [11] proposed editing object materials in a static image. Liu et al. [4] estimated illumination and scene geometry from video for various video applications. Ge et al. [12] proposed an object-aware image editing approach to obtain consistency in structure, color, and texture in a unified way.

Rendering virtual objects into real scenes has been long investigated. A survey is provided by Kronander et al. [13]. Various ways have been explored to solve the problems of illumination and geometry recovery. Debevec [14] proposed estimating scene radiance and global illumination using a mirrored ball to capture a high-dynamic range lighting environment, to support object insertion. Karsch et al. [1] developed an image composition system to render synthetic objects into legacy photographs. The scene structure and area light are provided by user interaction or a data-driven approach [2]. Briefly, previous methods for shadow synthesis either require user interaction and scene knowledge, or recover explicit representations of scene geometry and illumination. Our method, in contrast, is novel in synthesizing shadows using a convolutional neural network without any requirements about the scene or the inserted object model.

2.2 Shadow detection and removal

The opposite problem to shadow synthesis, i.e., *shadow detection and removal*, has been studied in the computer vision community [15–20]. Its goals are to separate the target image into lit and shadowed areas, and thence to remove the shadows. In early work, color [15, 20], edge [18], or segmentation [19] cues was used to build high level features for shadow description. Ma et al. [21] introduced appearance harmonization that makes the appearance of a de-shadowed region compatible with the rest of the image. Recently, convolutional neural networks for shadow removal have been proposed [16, 17]. In Ref. [17], the input image is decomposed into a shadow-free image and a shadow matte; the shadow matte is predicted using a convolutional neural network. Two stacked conditional GANs successively detect the shadow region and remove the shadow matte.

In the shadow removal problem, the objects casting shadows are commonly absent, while in the shadow synthesis problem, a virtually inserted object is present.

2.3 Image-to-image translation using generative adversarial networks

Goodfellow et al. [22] first introduced the concept of the generative adversarial network (GAN), consisting of two sub-networks: a generator (G) and a discriminator (D). G's task is to generate outputs to resemble the ground truth, while D tries to distinguish between fake and real inputs, i.e., between generated output and the ground truth. G and D work against each other, and the ideal outcome is for G to produce outputs that D cannot discriminate. Since its introduction, the GAN method has been widely applied to image-to-image translation problems, such as face image synthesis [23–25], image super resolution [26], and image completion [27, 28]. Variations of the GAN architecture have also been developed, including conditional GAN [29, 30], CycleGAN [31], StarGAN [24], etc.

Isola et al. [30] proposed a GAN network that translates an image into another domain, such as from a sketch to a photo, from architectural maps to photos, from black-and-white to color photos, etc. Their approach used a U-net structure inside the generator, enabling earlier convolutions to be concatenated with later deconvolutional layers to pass down information about the input. In an image completion task [27], the contents of an arbitrary image region conditioned on its surroundings are generated by a convolutional neural network. Later, Iizuka et al. [28] proposed an image completion network with global and local discriminators. The addition of a local discriminator helps scrutinize the details of the completed image. Portenier et al. [32] developed the Faceshop system which supports interactive face editing with user provided sketch and color information as input conditions for the GAN architecture. Wei et al. [33] proposed to learn adaptive receptive fields instead of manually selecting dilated convolutional kernels.

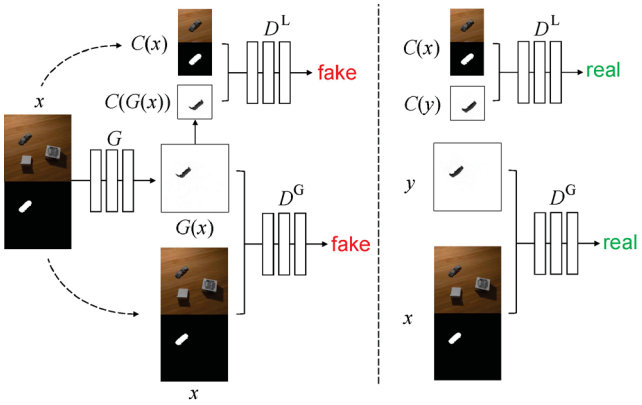

Fig. 2 Training a conditional generative adversarial network to synthesize shadow maps. The local discriminator D^{L} learns to classify between fake and real cropped tuples. The global discriminator D^{G} learns to classify between fake and real tuples from a global view. The generator G learns to fool the discriminator.

Our proposed ShadowGAN is an adaption of GAN, which uses a local discriminator to guarantee shape correctness and a global discriminator to guarantee direction and area compatible with other objects' shadows.

3 Method

3.1 Training data

3.1.1 Approach

Our proposed ShadowGAN is trained on synthetic data, where static scene images are rendered using 3D models indexed by ShapeNet [34]. Given an input target scene image I_{t} including original objects with shadows and a virtually inserted object without shadow, whose position in the scene is specified by a mask m_{s}, our goal is to predict a shadow map S, with which the output image I_{o} with a synthetic shadow

can be obtained by a simple pixel-wise product operation $I_o = I_t * S$. With the scene image I_t and source object mask m_s as inputs, the shadow map S is predicted using a generative network (see Fig. 3), where a reconstruction loss and two adversarial losses are used to guarantee the synthesis produces realistic output.

As a supervised deep learning-based image synthesis method, ShadowGAN requires paired input and ground truth images as training data, where the input scene image I_t contains N objects ($N \leqslant 3$ is assumed in our work) with shadows and one virtually inserted source object without a shadow; its mask m_s indicating the insertion region is also provided. The ground truth shadow map S has the same size as I_t. Each position p of S is associated with a real number, indicating that the output synthetic image color $I_o(p)$ can be obtained by multiplying the scene image color $I_t(p)$ by the coefficient $S(p)$, under the assumption that ambient light is present in the scene.

Such data are impossible to effectively collect in real life. Firstly, on one hand, scenes in which a few objects have shadows and one object is fully lit do not realistically occur in reality, while on the other hand, if the virtually inserted object is copied and pasted from other photos, the ground truth shadow map S cannot be generated efficiently and realistically. Secondly, a wide variety of illumination, scenes, and camera configurations are required for training data, which is both tedious and challenging for real-life photo capture.

Instead of using real-life photos, we use rendering technology to generate the training data. We render each target scene image I_t with N objects placed on the ground with shadows and one object with its shadow turned off. The shadow map S is generated by rendering a scene image I'_t with all the shadows turned on, then dividing it by I_t: $S = I'_t/I_t$.

3.1.2 Scenes

We use a sub-set of commonly seen 3D model categories such as *can*, *printer*, *bed*, etc. from a publicly available dataset, ShapeNet [34]. The object categories used for rendering are listed in Table 1. In total, 9265 objects were selected for rendering scenes. To render realistic ground planes, we downloaded textures from Internet using key-words search for, e.g., *woollen*, *stone*, *tablecloth*. A total of 110 textures were randomly chosen for rendering the plane. In each target scene image, up to four objects were randomly selected from the model collection, one of them being the virtually inserted object, and the rest being the original objects in the scene.

Table 1 ShapeNet 3D model categories used to render the target scene

bus	coach	mug	printer	stove
bowl	dishwasher	can	machine	motorcar
bag	grip	suitcase	bathtub	bookshelf
cabinet	auto	car	mailbox	microwave
washer	tower			

We assume each of the x, y, z coordinates to be in the range $[-1, 1]$: the ground plane is set to $P = \{(x, y, z)|x \in [-1, 1], y \in [-1, 1], z = 0\}$. The four randomly selected objects are placed at locations $(0.6, 0.6, 0), (-0.6, 0.6, 0), (-0.6, -0.6, 0), (0.6, -0.6, 0)$, randomly rotated about the z-axis.

3.1.3 Camera

The camera position $P_c = (x_c, y_c, z_c)$ was randomly chosen in the 3D space within the range:

$$3.5 \leqslant \sqrt{{x_c}^2 + {y_c}^2 + {z_c}^2} \leqslant 4.5$$

$$\pi/6 \leqslant \arcsin\left(z_c/\sqrt{{x_c}^2 + {y_c}^2 + {z_c}^2}\right) \leqslant \pi/3$$

3.1.4 Illumination

All scenes were illuminated by a single white point light with fixed intensity. The distance between the light and the center of the floor was randomly chosen in a limited range: the light position $P_l = (x_l, y_l, z_l)$ was randomly chosen in the following range:

$$3.5 \leqslant \sqrt{{x_l}^2 + {y_l}^2 + {z_l}^2} \leqslant 4.5$$

$$\pi/4 \leqslant \arcsin\left(z_l/\sqrt{{x_l}^2 + {y_l}^2 + {z_l}^2}\right) \leqslant \pi/3$$

3.1.5 Rendering

We used path tracing [35] to render the scenes, with 128 samples per pixel. To find the mask of the inserted object, we rendered it again with its material set to pure black, and then extracted its mask from the rendered image.

3.1.6 Training data

As a result, 12,400 training samples were generated, comprising a scene image I_t, source object mask m_s, and ground truth shadow map S, rendered at resolution 256×256.

3.2 Formulation

3.2.1 Approach

Our goal is to train a generator G that learns a

mapping function from domain X to domain Y, where $X = \{x_i\}_{i=1}^N$ are input scenes with virtually inserted object mask $x_i = \langle I_{\mathrm{t}}^i, m_{\mathrm{s}}^i \rangle$, and $Y = \{y_j\}_{j=1}^N$ are the corresponding shadow maps $y_i = S^i$. The key requirement for learning is that the generated shadow map $G(x)$ should reconstruct the shadow map, while not being distinguished from the ground truth shadow map data $y \approx p_{\mathrm{data}}(y)$. We introduce a local discriminator D^{L} and a global discriminator D^{G} which are trained to detect the generated shadow maps as "fakes" from aspects of local shape and global direction and area, respectively. Our objective thus contains a reconstruction loss $\mathcal{L}_{L_1}$, a local adversarial loss $\mathcal{L}_{\mathrm{GAN}}^{\mathrm{L}}$, and a global adversarial loss $\mathcal{L}_{\mathrm{GAN}}^{\mathrm{G}}$.

3.2.2 Reconstruction loss

Reconstruction loss is commonly used in supervised image-to-image translation problems [28, 30, 36], to constrain the generated result to be similar to the ground truth in an L_1 or L_2 sense. Here we use L_1 norm reconstruction loss to measure the error between the predicted shadow map $G(x)$ and the ground truth shadow map y:

$$\mathcal{L}_{L_1}(G) = \| y - G(x) \|_1 \tag{1}$$

3.2.3 Local adversarial loss

The local discriminator D^{L} tries to distinguish the generated fake results $G(x)$ from real samples y from local considerations, so only looks at the region around the source object. Intuitively, the generated shadow $G(x)$ for the source object should be as similar as possible to the ground truth sample y within a local region. We crop a square region centered at the source object, of side half the original image size, i.e., 128×128 pixels, and only pass the cropped region of the predicted shadow map $C(G(x))$ or ground truth shadow map $C(y)$, with conditional input scene image and source object mask $C(x)$, to the local discriminator. Here, $C(\cdot)$ is the cropping operator. The local adversarial loss is defined to be

$$\begin{aligned}\mathcal{L}_{\mathrm{GAN}}^{\mathrm{L}}(G, D^{\mathrm{L}}) = {} & \mathbb{E}_{x,y}[\log D^{\mathrm{L}}(C(x), C(y))] \\ & + \mathbb{E}_x[1 - \log D^{\mathrm{L}}(C(x), C(G(x)))]\end{aligned} \tag{2}$$

G tries to minimize this objective against the local adversarial D^{L} that tries to maximize it. D^{L} takes the cropped version of either conditional real samples $\langle x, y \rangle$ or generated fake samples $\langle x, G(x) \rangle$ as inputs. The discriminator determines whether the samples are *real* or *fake*.

3.2.4 Global adversarial loss

The global discriminator D^{G} tries to distinguish the generated fake results $G(x)$ from real samples y using a global view of the whole shadow map. In particular, the generated shadow $G(x)$ for the source object should be compatible with other objects' shadows in the original scene in terms of direction and area.

$$\begin{aligned}\mathcal{L}_{\mathrm{GAN}}^{\mathrm{G}}(G, D^{\mathrm{G}}) = {} & \mathbb{E}_{x,y}[\log D^{\mathrm{G}}(x, y)] \\ & + \mathbb{E}_x[1 - \log D^{\mathrm{G}}(x, G(x))]\end{aligned} \tag{3}$$

where G tries to minimize this objective against the global adversarial D^{L} that tries to maximize it. D^{G} takes either conditioned real samples $\langle x, y \rangle$ or conditioned generated fake samples $\langle x, G(x) \rangle$ as inputs.

3.2.5 Full objective

The overall objective is the weighted sum of the loss terms:

$$\begin{aligned}\mathcal{L}(G, D^{\mathrm{L}}, D^{\mathrm{G}}) = {} & \mathcal{L}_{\mathrm{GAN}}^{\mathrm{L}}(G, D^{\mathrm{L}}) \\ & + \mathcal{L}_{\mathrm{GAN}}^{\mathrm{G}}(G, D^{\mathrm{G}}) \\ & + \lambda \mathcal{L}_{L_1}(G)\end{aligned} \tag{4}$$

where $\lambda = 200$ controls the relative importance of the objective terms. The goal is to determine:

$$G^* = \arg \min_G \max_{D^{\mathrm{L}}, D^{\mathrm{G}}} \mathcal{L}(G, D^{\mathrm{L}}, D^{\mathrm{G}}) \tag{5}$$

3.3 Implementation

3.3.1 Conditional shadow map generator

Figure 3 visualizes the conditional shadow map generator. The generator takes an input of size 256×256 with 4 channels; 3 are RGB channels from the target scene and 1 is the source object mask m_{s}. The output is a single channel shadow map of size 256×256. We adopt the encoder–decoder architecture

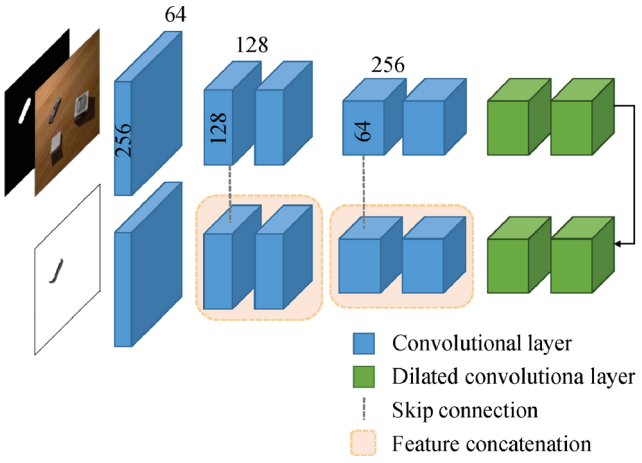

Fig. 3 Conditional shadow map generator.

proposed by Isola et al. [30], where skip connections (U-net) are set up to concatenate the corresponding layers in encoder and decoder. The generator downsamples the input using strided convolutions, followed by intermediate layers of dilated convolutions [37] before upsampling using transposed convolutions. We use the ReLU activation function after each layer except for the output layer, which uses a tanh activation function. In total, the proposed editing network has 15 convolutional layers with up to 256 feature channels.

3.3.2 Discriminator networks

Following Iizuka et al. [28] and Portenier et al. [32], we use local and global discriminators as adversaries for generator training (see Fig. 4). The input to the global discriminator is a $256 \times 256 \times 5$ tensor: a fake shadow map sample S_{f} or a real shadow map sample S_{r}, conditional input target scene image I_{t}, and the inserted object mask m_{s}. The local discriminator uses the same input tensor but works on a cropped region of size 128×128 centered around the inserted object position.

Both discriminators are fully-convolutional networks, with the spatial tensor dimension gradually downsampled to 1×1. Feature channels increase up to 512 channels then decrease to 1. The outputs of discriminators are predictions whether the inputs are more like *real* samples or *fake* ones. We use leaky ReLU activation functions with slope set to 0.2 everywhere in the discriminators, except for the last layer which uses a sigmoid activation function. Full network architectural details are provided in Tables 2 and 3.

3.4 Optimization and parameters

To optimize the proposed ShadowGAN, we follow Ref. [30] in which gradient descent steps for D and G are alternately performed. We apply the Adam solver [38] with learning rate set to 0.0002, and momentum parameters $\beta_1 = 0.5$, $\beta_2 = 0.999$. The training process using 100 epochs takes about 5 hours on a Titan 1080 Ti graphic card.

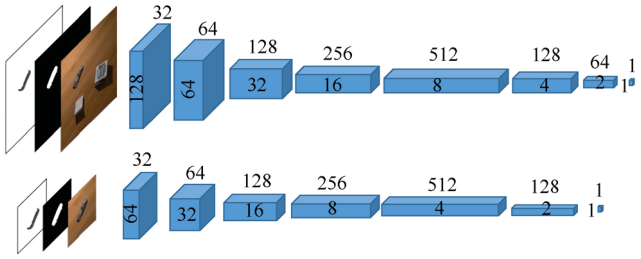

Fig. 4 Discriminator architecture, comprising a global (top) and a local (bottom) network.

Table 2 Generator architecture. After each convolutional layer, except the last, there is a rectified linear unit (ReLU) layer. The output layer consists of a convolutional layer with a tanh function instead of a ReLU layer. "Outputs" gives the number of output channels for the output of the layer

Type	Kernel	Dilation	Stride	Outputs
Conv.	5×5	1	1×1	64
Conv.	3×3	1	2×2	128
Conv.	3×3	1	1×1	128
Conv.	3×3	1	2×2	256
Conv.	3×3	1	1×1	256
Dilated Conv.	3×3	2	1×1	256
Dilated Conv.	3×3	4	1×1	256
Dilated Conv.	3×3	8	1×1	256
Dilated Conv.	3×3	16	1×1	256
Conv.	3×3	1	1×1	256
Deconv.	4×4	1	1/2×1/2	128
Conv.	3×3	1	1×1	128
Deconv.	4×4	1	1/2×1/2	64
Conv.	3×3	1	1×1	64
Output	3×3	1	1×1	1

Table 3 Discriminator architectures. All Conv. layers are followed by leaky ReLU activation (slope 0.2). The output layer consists of a convolutional layer with sigmoid activation; it predicts the probability that an input shadow map is from real samples rather than the generator network

(a) Local discriminator

Type	Kernel	Stride	Outputs
Conv.	4×4	2×2	32
Conv.	4×4	2×2	64
Conv.	4×4	2×2	128
Conv.	4×4	2×2	256
Conv.	4×4	2×2	512
Conv.	4×4	2×2	128
Output	4×4	2×2	1

(b) Global discriminator

Type	Kernel	Stride	Outputs
Conv.	4×4	2×2	32
Conv.	4×4	2×2	64
Conv.	4×4	2×2	128
Conv.	4×4	2×2	256
Conv.	4×4	2×2	512
Conv.	4×4	2×2	128
Conv.	4×4	2×2	64
Output	4×4	2×2	1

4 Results

4.1 Initial tests

We have tested ShadowGAN on rendered synthetic scenes from the test set. The test set was rendered

using the same rendering strategy as for the training set, with randomly selected models, placed object positions and orientations, illumination and camera configurations. Time for shadow synthesis was about 0.3 s for a 256×256 input image on a Titan 1080 Ti graphic card. A gallery of corresponding synthetic shadowed results is shown in Fig. 8. Figure 5 shows synthetic results with the same scene and illumination, but viewed from randomly selected viewpoints. It can be seen that even when observed from different view points, the synthetic shadows are visually realistic. As a further test, Fig. 6 shows results with the same scene and illumination, but slightly different camera poses caused by camera rotation. It can be seen that the synthetic shadows are temporally consistent during the camera movement.

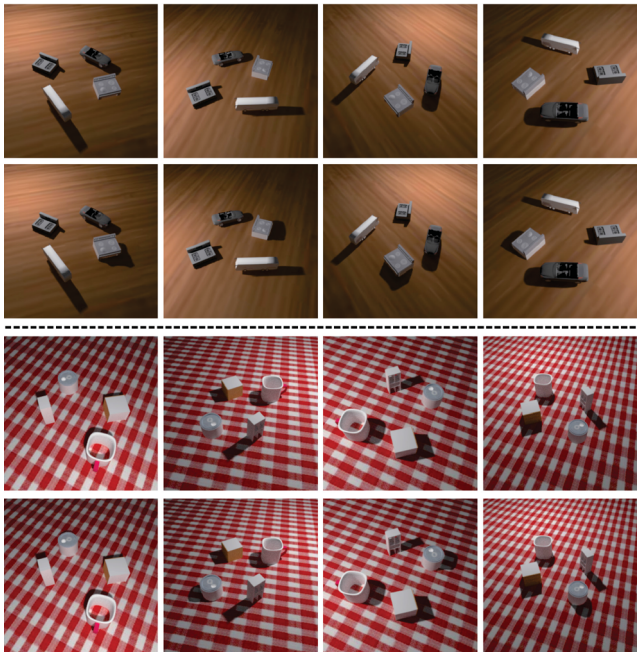

Fig. 5 Two examples of shadow synthesis for the same scene and illumination, with different viewing angles. In each example, top row: input scenes, bottom row: corresponding synthetic results.

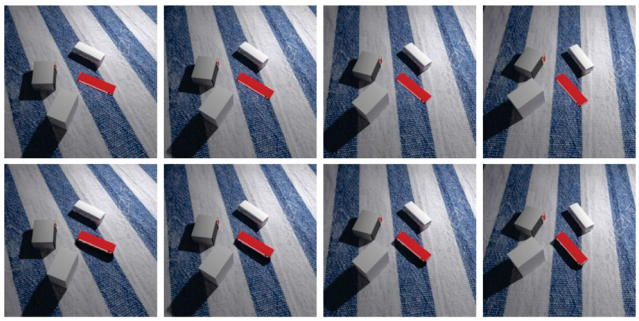

Fig. 6 Shadow synthesis for the same scene and illumination, with a slightly rotated camera. Top row: input scenes, bottom row: corresponding synthetic results.

ShadowGAN supports inserting virtual objects in sequence. Figure 7 shows an example of step-by-step object insertions with shadows synthesized using our method.

As ShadowGAN is the first deep learning-based shadow synthesis network, we next present an ablation study to demonstrate the benefits of the components of our system, followed by a user study to verify whether *fake* results from ShadowGAN are indistinguishable from *real* ones.

4.2 Ablation study

In order to evaluate the effectiveness of components of the proposed method, we re-evaluated ShadowGAN with alternative loss functions: with only the reconstruction loss (denoted as L_1), with the reconstruction loss and the local adversarial loss (denoted as $L_1 + \text{Local}$) and with the reconstruction loss and the global adversarial loss (denoted as $L_1 + \text{Global}$). Representative visual results are shown in Fig. 9. The results indicate that with some losses turned off, using functions L_1, $L_1 + \text{Local}$, and $L_1 + \text{Global}$ do not generalize well to the test samples and fail to predict visually plausible shadows with correct shape, area, and direction.

We also evaluated an input variation, in which the input source object position was not explicitly provided by a mask m_s either for the generator or for the discriminators. Figure 10 provides a visual comparison under input variations. The results indicate that the source object mask m_s is essential for ShadowGAN to obtain good results.

4.3 User study

To further assess whether the synthetic shadows for virtually inserted objects are visually natural and authentic, we conducted a user study with the task of observing and determining whether the shadows from our synthetic results look real. We also showed *real* scenes to the subjects and asked them to determine whether the images were *real*.

Fig. 7 Inserting virtual objects in sequence.

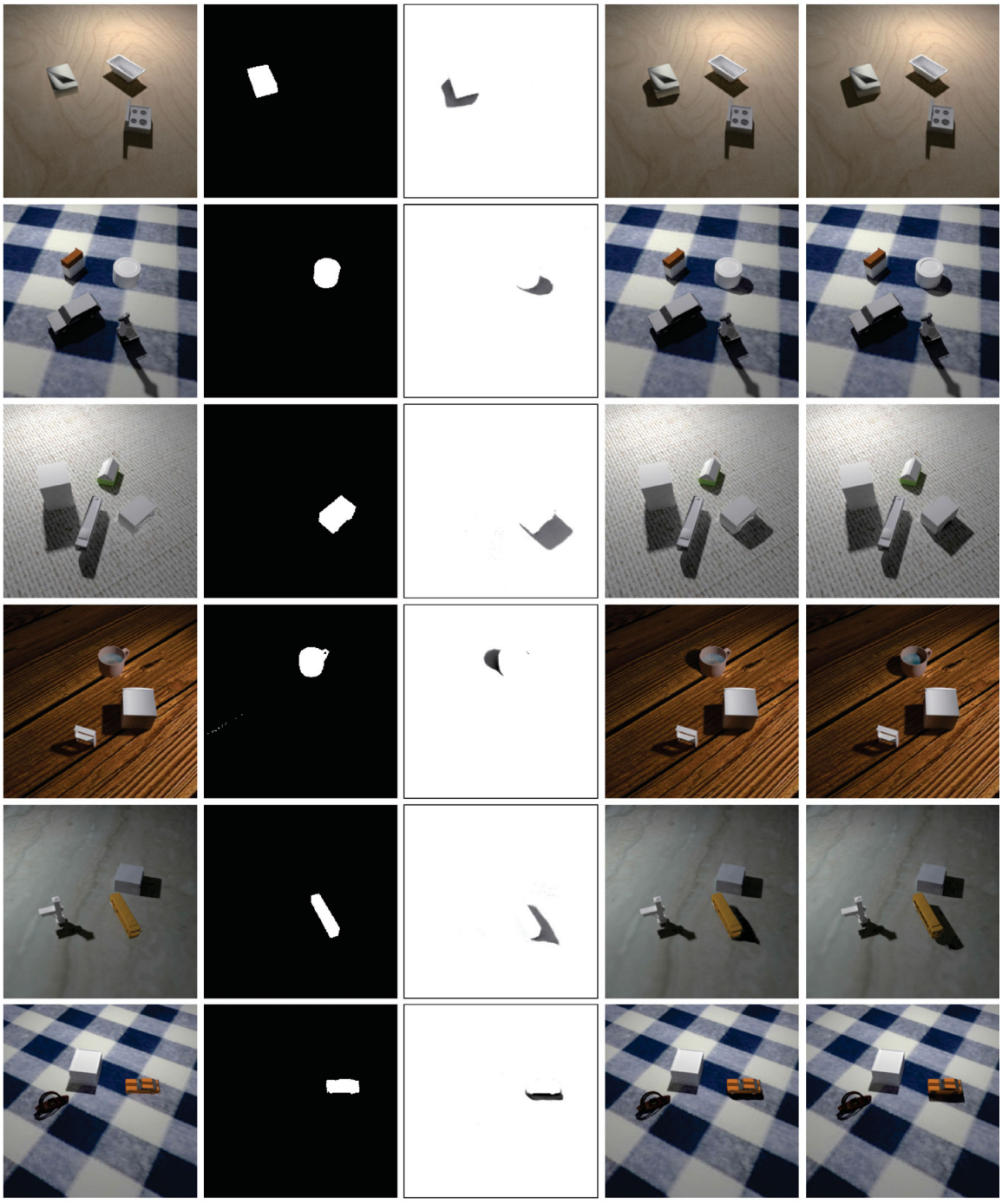

Fig. 8 Gallery of synthetic results. Each example, left to right: (a) input target scene with a virtual source object, (b) input source object mask, (c) predicted shadow map using ShadowGAN, (d) synthetic shadowed result, and (e) ground truth shadowed result.

We collected 20 pairs of *real* and *fake* shadowed results from the test scenes; each pair shows the same scene. We invited 20 subjects without viewing or perception issues to observe and rate the images. Each subject observed a randomly selected image from each scene pair—either the synthetic result or the real shadowed image, and assessed whether the shadows in the image were *real*. We collected all votes from the subjects, and summarise the results of the user study in Table 4. As a result, 50.48% of our synthetic shadows were assessed to be *real* images. Even shadows in the real images were sometimes considered to be *fake*; only 57.14% were considered to be real. The summary indicates that the visual

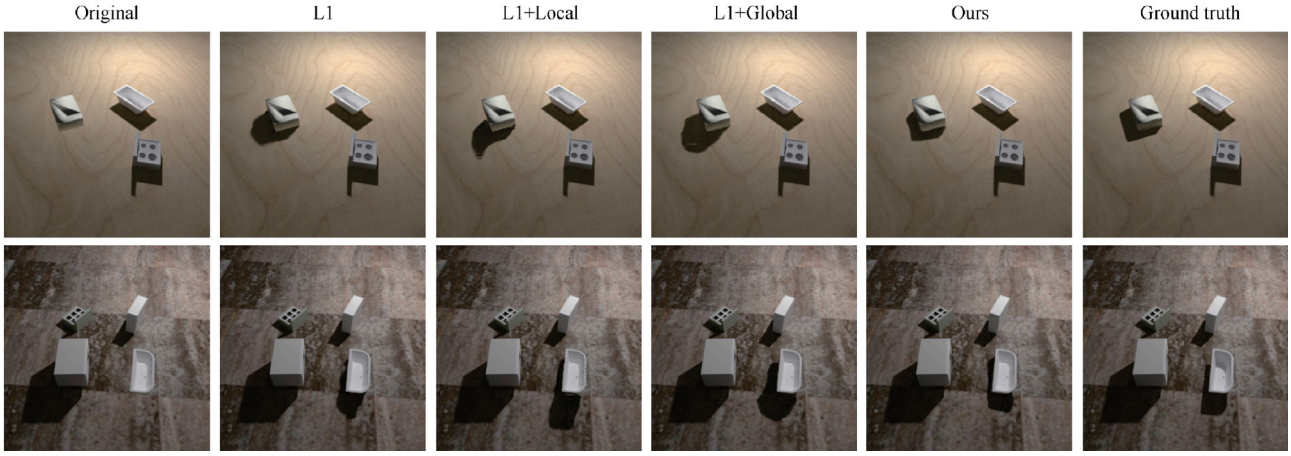

Fig. 9 Ablation study for loss functions. Different losses lead to different qualities of results. Each column shows results trained under a different loss.

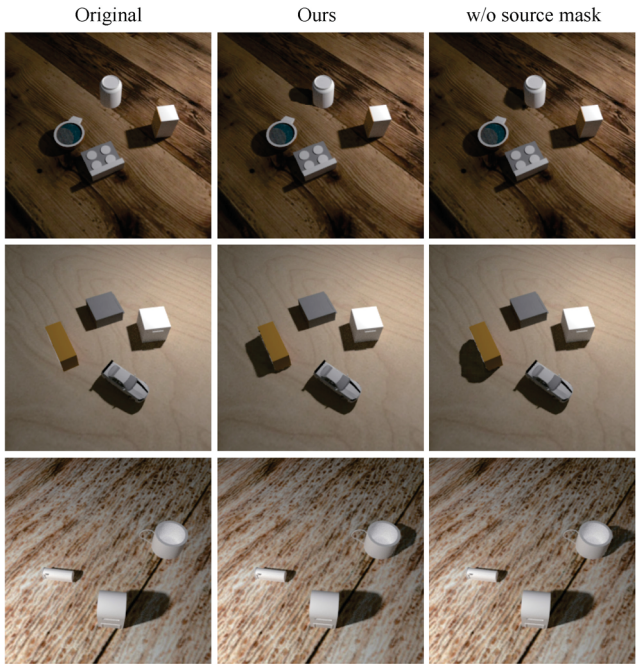

Fig. 10 Ablation study for source mask. Each row shows a scene with our shadow synthesis result and the result without source mask, m_s.

Table 4 User study summary

	Rated real	Rated fake
Real scene	57.14%	42.86%
Synthetic scene	50.48%	49.52%

effectiveness of synthetic results from ShadowGAN is close to that in rendered scenes.

5 Limitations and conclusions

ShadowGAN has limitations. Firstly, as discussed in Section 3.1, our training set and test set were produced using rendering technology on public 3D models rather than using real-life photos. As collecting real-life photos with some objects' shadows turned off is a challenging task, we regard collecting and testing real-life photos as requiring further work. Secondly, when testing our model, a scene with only *one* virtually inserted object is fed into the network. Synthesizing shadows for multiple objects is not supported by ShadowGAN. However as we have shown in the experimental results, users may iteratively perform insertion operations, one object at a time. As pioneering work that uses GAN to synthesize shadows for virtual object, we only tested our model on 256×256 images (as did Ref. [30]).

In summary, we have presented a generative adversarial network—ShadowGAN—which can synthesize shadows for virtual objects in images. Shadows are predicted from a generator which during training competes against a local discriminator and a global discriminator. To our knowledge, this is the first novel shadow synthesis solution using a deep learning-based framework. It benefits from being free from input constraints and is computational effective. For network training, we have produced a large set of rendered scenes using public 3D models in commonly seen object categories. We believe both the training data and ShadowGAN will benefit the community of computer graphics and virtual reality.

Acknowledgements

The authors would like to thank all the reviewers. This work was supported by the National Natural Science Foundation of China (Project Nos. 61561146393 and 61521002), the China Postdoctoral

Science Foundation (Project No. 2016M601032), and a Research Grant of Beijing Higher Institution Engineering Research Center.

References

[1] Karsch, K.; Hedau, V.; Forsyth, D.; Hoiem, D. Rendering synthetic objects into legacy photographs. *ACM Transactions on Graphics* Vol. 30, No. 6, Article No. 157, 2011.

[2] Karsch, K.; Sunkavalli, K.; Hadap, S.; Carr, N.; Jin, H.; Fonte, R.; Sittig, M.; Forsyth, D. Automatic scene inference for 3D object compositing. *ACM Transactions on Graphics* Vol. 33, No. 3, Article No. 32, 2014.

[3] Kee, E.; O'Brien, J. F.; Farid, H. Exposing photo manipulation from shading and shadows. *ACM Transactions on Graphics* Vol. 33, No. 5, Article No. 165, 2014.

[4] Liu, B.; Xu, K.; Martin, R. R. Static scene illumination estimation from videos with applications. *Journal of Computer Science and Technology* Vol. 32, No. 3, 430–442, 2017.

[5] Bell, S.; Bala, K.; Snavely, N. Intrinsic images in the wild. *ACM Transactions on Graphics* Vol. 33, No. 4, Article No. 159, 2014.

[6] Bi, S.; Han, X.; Yu, Y. An L_1 image transform for edge-preserving smoothing and scene-level intrinsic decomposition. *ACM Transactions on Graphics* Vol. 34, No. 4, Article No. 78, 2015.

[7] Bousseau, A.; Paris, S.; Durand, F. User-assisted intrinsic images. *ACM Transactions on Graphics* Vol. 28, No. 5, Article No. 130, 2009.

[8] Fan, Q.; Yang, J.; Hua, G.; Chen, B.; Wipf, D. Revisiting deep intrinsic image decompositions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 8944–8952, 2018.

[9] Panagopoulos, A.; Samaras, D.; Paragios, N. Robust shadow and illumination estimation using a mixture model. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 651–658, 2009.

[10] Sato, I.; Sato, Y.; Ikeuchi, K. Illumination from shadows. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 25, No. 3, 290–300, 2003.

[11] Khan, E. A.; Reinhard, E.; Fleming, R. W.; Bulthoff, H. H. Image-based material editing. *ACM Transactions on Graphics* Vol. 25, No. 3, 654–663, 2006.

[12] Ge, S.; Jin, X.; Ye, Q.; Luo, Z.; Li, Q. Image editing by object-aware optimal boundary searching and mixed-domain composition. *Computational Visual Media* Vol. 4, No. 1, 71–82, 2018.

[13] Kronander, J.; Banterle, F.; Gardner, A.; Miandji, E.; Unger, J. Photorealistic rendering of mixed reality scenes. *Computer Graphics Forum* Vol. 34, 643–665, 2015.

[14] Debevec, P. Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography. In: Proceedings of the ACM SIGGRAPH 2008 Classes, Article No. 32, 2008.

[15] Shor, Y.; Lischinski, D. The shadow meets the mask: Pyramid-based shadow removal. *Computer Graphics Forum* Vol. 27, No. 2, 577–586, 2008.

[16] Wang, J.; Li, X.; Yang, J. Stacked conditional generative adversarial networks for jointly learning shadow detection and shadow removal. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1788–1797, 2018.

[17] Qu, L.; Tian, J.; He, S.; Tang, Y.; Lau. R. W. H. DeshadowNet: A multi-context embedding deep network for shadow removal. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 4067–4075, 2017.

[18] Xu, L.; Qi, F.; Jiang, R. Shadow removal from a single image. In: Proceedings of the 6th International Conference on Intelligent Systems Design and Applications, 1049–1054, 2006.

[19] Guo, R.; Dai, Q.; Hoiem, D. Single-image shadow detection and removal using paired regions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2033–2040, 2011.

[20] Zhang, L.; Zhang, Q.; Xiao, C. Shadow remover: Image shadow removal based on illumination recovering optimization. *IEEE Transactions on Image Processing* Vol. 24, No. 11, 4623–4636, 2015.

[21] Ma, L.-Q.; Wang, J.; Shechtman, E.; Sunkavalli, K.; Hu, S.-M. Appearance harmonization for single image shadow removal. *Computer Graphics Forum* Vol. 35, No. 7, 189–197, 2016.

[22] Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. In: Proceedings of the Advances in Neural Information Processing Systems 27, 2672–2680, 2014.

[23] Li, M.; Zuo, W.; Zhang, D. Deep identity-aware transfer of facial attributes. *arXiv preprint* arXiv:1610.05586, 2016.

[24] Choi, Y.; Choi, M.; Kim, M.; Ha, J.-W.; Kim, S.; Choo, J. StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 8789–8797, 2018.

[25] Shen, W.; Liu, R. Learning residual images for face attribute manipulation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 4030–4038, 2017.

[26] Ledig, C.; Theis, L.; Huszar, F.; Caballero, J.; Cunningham, A.; Acosta, A.; Aitken, A.; Tejani, A.; Totz, J.; Wang, Z.; Shi, W. Photo-realistic single image super-resolution using a generative adversarial network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 4681–4690, 2017.

[27] Pathak, D.; Krahenbuhl, P.; Donahue, J.; Darrell, T.; Efros, A. A. Context encoders: Feature learning by inpainting. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2536–2544, 2016.

[28] Iizuka, S.; Simo-Serra, E.; Ishikawa, H. Globally and locally consistent image completion. *ACM Transactions on Graphics* Vol. 36, No. 4, Article No. 107, 2017.

[29] Mirza, M.; Osindero, S. Conditional generative adversarial nets. *arXiv preprint* arXiv:1411.1784, 2014.

[30] Isola, P.; Zhu, J.-Y.; Zhou, T.; Efros, A. A. Image-to-image translation with conditional adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1125–1134, 2017.

[31] Zhu, J.-Y.; Park, T.; Isola, P.; Efros, A. A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proceedings of the IEEE International Conference on Computer Vision, 2223–2232, 2017.

[32] Portenier, T.; Hu, Q.; Szabo, A.; Bigdeli, S. A.; Favaro, P.; Zwicker, M. Faceshop: Deep sketch-based face image editing. *ACM Transactions on Graphics* Vol. 37, No. 4, Article No. 99, 2018.

[33] Wei, Z.; Sun, Y.; Lin, J.; Liu, S. Learning adaptive receptive fields for deep image parsing networks. *Computational Visual Media* Vol. 4, No. 3, 231–244, 2018.

[34] Chang, A. X.; Funkhouser, T.; Guibas, L.; Hanrahan, P.; Huang, Q.; Li, Z.; Savarese, S.; Savva, M.; Song, S.; Su, H.; Xiao, J.; Yi, L.; Yu, F. ShapeNet: An information-rich 3D model repository. *arXiv preprint* arXiv:1512.03012, 2015.

[35] Lafortune, E. P.; Willems, Y. D. Bi-directional path tracing. In: Proceedings of the 3rd International Conference on Computational Graphics and Visualization Techniques, 145–153, 1993.

[36] Shrivastava, A.; Pfister, T.; Tuzel, O.; Susskind, J.; Wang, W.; Webb, R. Learning from simulated and unsupervised images through adversarial training. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2107–2116, 2017.

[37] Yu, F.; Koltun, V. Multi-scale context aggregation by dilated convolutions. *arXiv preprint* arXiv:1511.07122, 2015.

[38] Kingma, D. P.; Ba, J. Adam: A method for stochastic optimization. *arXiv preprint* arXiv:1412.6980, 2014.

Shuyang Zhang is an undergraduate at the University of Michigan. Her research interests include computer graphics, computer vision, and machine learning.

Runze Liang is an undergraduate at Tsinghua University, Beijing. His research interests include computer graphics, image processing, and computer vision.

Miao Wang is an assistant professor in the State Key Laboratory of Virtual Reality Technology and Systems, Beihang University. He received his Ph.D. degree from Tsinghua University in 2016. During 2013–2014, he visited the Visual Computing Group in Cardiff University as a joint Ph.D. student. In 2016–2018, he worked as a postdoctoral researcher at Tsinghua University. His research interests lie in computer graphics with particular focus on interactive image and video editing and video in virtual reality.