Image Relighting with Object Removal from Single Image*

Yujia Zhang[1], Monica Perusquia Hernandez[2], Naoya Isoyama[3,*,†], Norihiko Kawai[4,†], Hideaki Uchiyama[5,†], Nobuchika Sakata[6,†] and Kiyoshi Kiyokawa[7,†]

[1] *Nara Institute of Science and Technology, 8916-5 Takayama-cho, Ikoma, Nara, Japan*
[2] *Osaka Institute of Technology, 1-79-1 Kitayama, Hirakata, Osaka, Japan*
[3] *Ryukoku University, 67 Tsukamoto-cho, Fukakusa, Fushimi-ku, Kyoto, Japan*

Abstract

We propose a method to relight scenes in a single image while removing unwanted objects by the combination of 3D-aware inpainting and relighting for a new functionality in image editing. First, the proposed method estimates the depth image from an RGB image using single-view depth estimation. Next, the RGB and depth images are masked by the user by specifying unwanted objects. Then, the masked RGB and depth images are simultaneously inpainted by our proposed neural network. For relighiting, a 3D mesh model is first reconstructed from the inpainted depth image, and is then relit with a standard relighting pipeline. In this process, removing cast shadows and sky areas and albedo estimation are optionally performed to suppress the artifacts in outdoor scenes. Through these processes, various types of relighting can be achieved from a single photograph while excluding the colors and shapes of unwanted objects.

Keywords

Image inpainting, Relighting, Image processing, Virtual reality

1. Introduction

VR systems using real-world photographs have been spreading, referred to as augmented virtuality [1]. For example, image-based rendering allows users to walk through a virtual space that imitates a real environment [2]. Also, systems such as "Tour into the picture" allow users to experience as if they jump into the world inside a photograph by setting up a single photograph and its corresponding geometry [3]. Since real-world photographs can enhance reality in VR, building VR systems using real-world images is essential.

Two issues must be addressed when using real-world photographs for those applications. (1) When using a photograph taken at a certain time with a certain light source, it is desirable to change the photograph's appearance so that users can experience a VR space that imitates the real world at various conditions. (2) In places such as sightseeing spots, many people can be included in the photographs. It is desirable to use such photographs as they are for the creation of some VR systems. In addition, some objects in the photographs, such as billboards for advertisements, can be obstacles for some VR creators.

Each issue has conventionally been solved as follows. For (1), geometry-based and learning-based methods have been proposed for relighting a single image. For (2), diffusion-based, patch-based, and learning-based inpainting methods have been proposed to remove unwanted objects from a single image. However, these two issues have been studied separately and are not integrated into one framework. To the best of our knowledge, no framework solves them simultaneously for image synthesis. Such a framework is essential for creating VR systems.

We propose a method for relighting scenes in an image while removing unwanted objects in one framework. The proposed method first estimates the depth image from an input RGB image using neural network-based single-view depth estimation. Next, the RGB and depth images are masked with a mask image by specifying unwanted objects by the user. They are simultaneously inpainted by our proposed neural network. Then, a 3D mesh model is reconstructed from the inpainted depth image. The mesh model is used to relight the inpainted image by taking into account the scene geometry from the 3D model. In this process, removing cast shadows and sky areas and albedo estimation are performed

APMAR'22: The 14th Asia-Pacific Workshop on Mixed and Augmented Reality, Dec. 02-03, 2022, Yokohama, Japan

*Corresponding author.
†These authors contributed equally.

zhang.yujia.zv2@is.naist.jp (Y. Zhang); m.perusquia@is.naist.jp (M. P. Hernandez); isoyama@is.naist.jp (N. Isoyama); norihiko.kawai@oit.ac.jp (N. Kawai); hideaki.uchiyama@is.naist.jp (H. Uchiyama); sakata@rins.ryukoku.ac.jp (N. Sakata); kiyo@is.naist.jp (K. Kiyokawa)

to suppress the influence of existing light sources in outdoor scenes. Through these processes, various types of relighting can be achieved from a single photograph while excluding the colors and shapes of unwanted objects. Finally, the processed image can be applied to VR applications, image editing, and other applications using image synthesis.

2. Related Work

2.1. Relighting

Image relighting is a technique to reproduce the shadows, brightness, and color of an object or scene taken in a different lighting environment. It solves the problem of reconstructing the light of a photograph. We classify the algorithms into geometry-based and learning-based approaches.

2.1.1. Geometry-based Methods

Prior work on image relighting has generally relied on scene geometry, light, and reflectance models using inverse rendering. The full use of geometry, materials, and lighting in scene representation allows for conventional rendering and shading techniques with promising results. Techniques such as semi-automatic vision-based geometry reconstruction [4] or parameter estimation by viewing the same scene under different lighting conditions [5] can simplify the capture process. In addition, advanced capture setups such as Light Stage have been used in film production [6].

2.1.2. Learning-based Methods

Learning-based methods have significantly improved the performance of multiview relighting systems for scene-scale relighting. A typical approach is to use a single neural network to map the input image and a set of approximate guide maps, such as depth maps or shadow maps, to new lighting conditions [7]. Some methods remove the original lighting influence and transform it into new lighting conditions, depending on the geometric property [8].

2.2. RGB Image Inpainting

RGB image inpainting is a technique that fills in missing regions (ROI: Region of Interest) with image texture consistency throughout the entire image. The techniques remove scratches and text masks in photographs, remove unwanted targets, and reproduce image block contents due to network packet loss during image transmission. We classify the algorithms into diffusion-based, patch-based, and learning-based approaches.

2.2.1. Diffusion-based and Patch-based Methods

Diffusion-based methods smoothly transfer the effective information from the known region to the target region by diffusing the pixels at the junction. The method of Bertalmio et al., who first proposed the term "image inpainting" as an early study, belongs to this category [9]. In this category, diffusion is mathematically formulated in various manners. Mumford-Shah segmentation model was adapted for image inpainting by introducing Euler's Elastica [10]. Li *et al.* proposed diffusion-based inpainting by analyzing the local variance of image Laplacian along the isophote direction [11].

Patch-based methods search for patches that can match the ROI in the source region of the image and then fill it with the patches. An early study [12] synthesizes patches into the ROI sequentially, followed by overall optimization of patch-based costs [13]. Various improvements have been made since then. For example, the Markov random field (MRF) modeling method segments the image into blocks and uses a prior to limit the effective matching context candidate patches of the source region [14]. Another method uses Gaussian-weighted nonlocal texture similarity measure to obtain multiple candidate patches and nonlinear filtering (α-trimmed mean filter) to the inpainting target region in pixel [15].

2.2.2. Learning-based Methods

Deep neural networks have been introduced for image inpainting. We divide the mainstream approaches into convolutional neural networks (CNN) and generative adversarial networks (GAN) based on the network architecture.

CNN-based approaches can compensate for the lack of global information distortion. Networks with the encoder-decoder structure are common in this field. The context encoder in the encoder-decoder network [16] can effectively use the local information around the target region and the global information of the whole image to generate information. Zeng *et al.* proposed a pyramidal context encoding the network PEN-Net based on the U-Net structure [17]. It can encode the contextual semantics from the full-resolution input and decode the learned semantic features for inpainting defective content.

GAN-based approaches have become the active research direction in image synthesis. The face feature point generation network [18], an image inpainting method for human faces, consists of three branching networks of image segmentation networks, and a cooperative GAN based on CGAN was proposed. Two-branch network Pluralistic [19] based on the CGAN architecture. One path is a reconstruction, and the other path is a generative path.

2.3. Depth Image Inpainting

Studies have been conducted to inpaint missing regions not only in RGB images but also in depth images. As for the inpainting of depth images, some research has been conducted to fill in missing regions in a depth image using the corresponding RGB image as a guide because missing regions are more likely to occur in depth images than in RGB images due to the difference in the measurement device [20]. As a different application, a patch-based method has been proposed for removing unwanted objects from two stereo RGB images while preserving consistency and reconstructing the depths of the two images [21]. The method most relevant to this research has been proposed to remove unwanted objects from RGB and depth images and simultaneously inpaint the ROI in RGB and depth images using a neural network [22].

2.4. Summary and Our Contribution

As mentioned above, relighting and inpainting have been studied separately. In other words, there is no study that has performed them simultaneously for image synthesis. In this paper, we propose an integrated framework that takes advantage of the features of these studies. First, due to the excellent performance of deep learning on inpainting, we take a cue from the literature [22] and use neural networks to simultaneously inpaint RGB and depth images. This two-stage structure can reduce the gap between the RGB map and the estimated depth map to some extent since we use the same intermediate output as a guide. Then, for now, learning-based relighting methods do not allow for free illumination changes or reasonable shadow generation. We utilize the inpainted depths to achieve relighting in a geometry-based manner.

3. Proposed Method

3.1. Overview

The goal of the proposed method is to remove unwanted objects from a photograph and change lighting conditions by considering 3D geometry. The method takes a single RGB image and an object mask that the user wants to remove as input. The inpainted image is used as input for relighting so that the user is free to change the illumination to output the requested image.

The overview of our proposed method is illustrated in Fig. 1. First, the depth image is estimated from the input RGB image by using single-view depth estimation (Fig. 1(a)) with MegaDepth [23]. The RGB and estimated depth images are inpainted together with our proposed inpainting network (Fig. 1(c)) after removing target regions masked by the user (Fig. 1(b)). Then, optional processing is performed before relighting for the outdoor environment (Fig. 1(d)). The processing includes shadow removal, sky removal for outdoor scenes, and albedo map estimation. The 3D mesh model is generated from the depth image, and is used for relighiting with a conventional rendering pipeline (Fig. 1(e)). By feeding the image with illumination mapping, we can obtain an image relit with new illumination that users can control freely.

The depth of the RGB image is first estimated to obtain the geometry of the RGB image by using the proposed depth estimation method. Next, the estimated depth map and the original image are masked with the object mask image and are inpainted by the inpainting network. Relighting runs the conventional rendering pipeline on the 3D model recovered from the depth map.

3.2. Single-View Depth Dstimation

In our method, the depth image estimation is important to realize relighting because the input is only a single RGB image. A depth image can be obtained in various ways, such as a depth camera or a depth map estimated using a depth estimation algorithm. Since surface normals have a strong guiding effect on illumination, the depth estimation with surface estimation is essential for relighting. In our implementation, we estimate a depth image from a single RGB image with MegaDepth [23].

3.3. Masking and RGBD Inpainting

Both the estimated depth image and the input RGB image are masked with the object mask image generated by the user. Since our method requires inpainting both the RGB image and its depth image, we propose to inpaint a 4-channel RGBD image to increase their information agreement.

3.3.1. Network Architecture

We propose an inpainting network based on two-stage structure [24]: 1) edge generators and 2) image

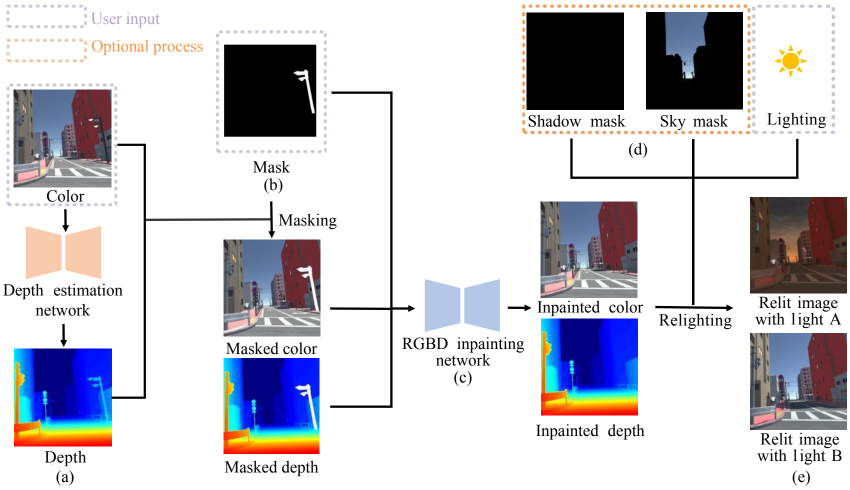


Figure 1: Method overview.

generators, referring to the two-stage GAN network architecture shown in Fig. 2. Both stages consist of a bundle of generators and discriminators. Let G_{edge} and D_{edge} denote the generator and discriminator of the edge generator, G_{image} and D_{image} denote the generator and discriminator of the image inpainting network, I_{source} be an RGBD image, and E_{source} and I_{gray} be an edge image generated by an edge detector and a grayscale image of the RGB image, respectively. The edge generator takes the masked grayscale image $\tilde{I}_{gray} = I_{gray} \odot \bar{M}$, the corresponding edge image $\tilde{E}_{source} = E_{source} \odot \bar{M}$, and the image mask M as pre-condition (1 for missing regions and 0 for background), where $\odot$ denotes the Hadamard product. Then, the generator predicts the edge image E_{pred} by filling the edges in the masked regions.

$$E_{pred} = G_{edge}(\tilde{I}_{gray}, \tilde{E}_{source}, M).$$

The image generators take a missing RGBD image $\tilde{I}_{source} = I_{source} \odot \bar{M}$ and a composed edge image E_{comp}, which is generated by compositing the ground-truth edges in the background region with the edges generated in the missing regions. That is, $E_{comp} = E_{source} \odot \bar{M} + E_{pred} \odot M$. Finally, the image generator outputs a RGBD image I_{pred} with the same resolution as the input image and with missing regions inpainted as follows.

$$I_{pred} = G_{image}(\tilde{I}_{source}, E_{comp}).$$

3.3.2. Loss Function

Reconstruction loss L_{rec} achieves the consistency between the overall structure of the missing region and the context. We introduce L1-smooth loss L_{1_smooth} because it corrects the zero-point non-smooth problem of L1 loss and is more robust against outliers than L2 loss. Also, adversarial Loss L_{adv} increases the flexibility.

In addition, we incorporate perceptual loss L_{perc} and style loss L_{style} with reference to [25]. These two losses are computed only for the three channels for RGB represented with a subsript $_{rgb}$.

$$\begin{aligned} L_{perc} &= \frac{1}{C_i H_i W_i} |\phi_i(I_{source\_rgb}) - \phi_i(I_{pred\_rgb})| \\ L_{style} &= \|G_{\phi_i(I_{source\_rgb})} - G_{\phi_i(I_{pred\_rgb})}\|_2 \end{aligned}$$

where ϕ_i is the activation map of the i-th activation layer of the network ϕ with the $C_j \times H_j \times W_j$ feature map. In this equation, ϕ is the learned VGG-19 network and ϕ_i corresponds to the activation maps from layers $relu1\_1$, $relu2\_1$, $relu3\_1$, $relu4_1$and $relu5\_1$ of the VGG-19 network. The Gram matrix G_{ϕ_i} is a $C_i \times C_i$ matrix $G_{\phi_i} = \psi\phi^T / C_i H_i W_i$. ψ is the matrix that resizes the ϕ_i matrix into $C_i \times H_i \times W_i$.

3.4. Relighting

Relighting is based on a conventional rendering pipeline on the 3D model recovered from the depth image. The world coordinates of the pixel points are computed directly. We use the most basic Laplace

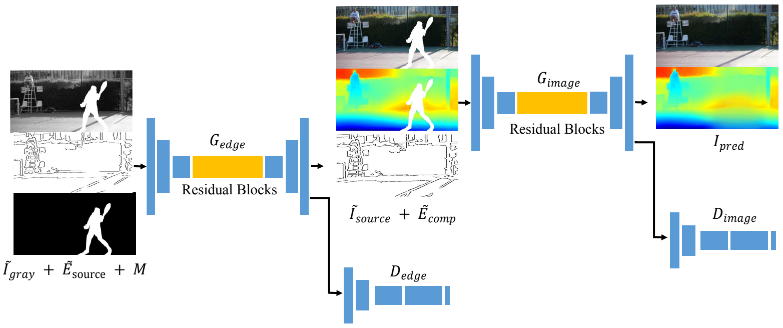

G_{edge}
Residual Blocks
$\tilde{I}_{gray}$ + $\tilde{E}_{source}$ + M
D_{edge}
$\tilde{I}_{source}$ + $\tilde{E}_{comp}$
G_{image}
Residual Blocks
I_{pred}
D_{image}

Figure 2: RGBD image inpainting network.

smoothing algorithm to smooth the model. The lightmap is baked directly on Unity3D and used for rendering.

3.4.1. Mesh Model Construction

We use the basic conversion from a depth image to a 3D mesh model. In other words, the world coordinates of the pixel points are computed directly from a depth image. Since the input is a single $W \times H$ image only, camera parameters are not available. Therefore, we set a virtual viewing angle α and calculates the virtual focal distance f by $f = \frac{W}{2} cot \frac{\alpha}{2}$. In practice, because of this basic construction method, there is no change in the model UVs and the model UV mapping, which is useful for our relighting in the next step.

We use the Laplace smoothing algorithm. The algorithm directly shifts the vertex positions without destroying the UVs of the model. The triangle mesh is centered at a vertex P and its adjacent vertices $P_1 .... P_{n-1}$ and all of its edges. More formally, the smoothing operation for each vertex P is as follows.

$$U(P) = \frac{1}{n} \sum_{i=1}^{n} Adj_i(P),$$

where n is the number of vertices adjacent to a vertex P, $Adj_i(P)$ is the i-th adjacent vertex, and $U(P)$ is the new position of the vertex P.

3.4.2. Light Map Generation

The implementation of relighting in the proposed method is mainly based on the generation of lightmaps. When the lighting of the model is changed, the lightmap of the corresponding lighting is generated and displayed. We bake a lightmap directly on Unity3D.

Enlighten simplifies the rendering equation by the following iterative formula

$$B_i = L_i + \rho_i \sum_{j=1}^{n} F_{ij} L_j$$

where B_i is the final light at a point i, L_i is the light at the point i itself, the bounce coefficient of the light between the two clusters is determined by F_{ij}, and L_j is the light at a point j, and ρ_i denotes the material property. This is why Enlighten can support changing the light source while leaving the scene objects unchanged.

3.4.3. Optional Processing

Light from strong sources, such as outdoor sunlight, produces distinct cast shadows. If the cast shadows in the original image are left as are, it may look strange due to inconsistencies with the cast shadows after relighting. Therefore, it is necessary to remove the shadows from the input image. We combine the Triple-cooperative Video Shadow Detection (ViSha [26]) and Stacked Generative Adversarial Networks (STCGAN [27]) for shadow detection and removal.

Although the sky in an image should be at infinity, the shape of the sky may be reconstructed by depth estimation. Such sky shapes cause negative effects when relighting. Therefore, removing the sky is an additional necessary process in outdoor scenes. To remove the sky, we first apply Pyramid Scene Parsing Network [28] to the RGB image to perform semantic segmentation. Next, pixels labeled as sky are removed and relighting is performed withoug the influence of the sky. After that, the sky area is composited with either the original sky or, if necessary, a virtual sky generated by computer graphics.

Albedo maps primarily reflect the texture and color of the model and are often referred to as diffuse reflectance maps. The albedo maps defines the color of the diffuse light. Albedo maps are estimated to effectively remove the effect of light from the original image. For the estimation of albedo and original illumination, we use InverseRenderNet [29].

4. Evaluation

To demonstrate the effectiveness, we first evaluate the performance of our RGBD inpainting network. Next, we show the results of the proposed method that combines image inpainting and relighting, using indoor and outdoor scenes constructed with computer graphics. Finally, we investigate the impact of optional processing: shadow removal, sky removal, and albedo estimation.

4.1. Training for RGBD Image Inpainting

We used the Microsoft Common Objects in Context (COCO) dataset to train the network on the irregular mask dataset provided by Liu et al. [30]. Especially we used the 2015 release COCO dataset, which contains a total of 165,482 training images, 81,208 validation images, and 81,434 test images. Since the dataset does not contain depth images, we estimated depth images from RGB images by MegaDepth [23] to obtain RGBD images and used them as the ground truth. For the mask dataset, 24,866 random datasets from the test dataset were used for training.

Some of the results are shown in Fig. 3. The results for the first and second row images are qualitatively good, while the edges are slightly blurred when compared to the ground truth. On the other hand, for the image in the third row, there is a relatively large missing region around the boundary where the two persons overlap, resulting in generating an unnatural texture. Based on these results, the trained network can produce good results in relatively simple cases where the background of the target to be removed is on the same object. However, the performance became worse in scenes where multiple objects overlap in the background of the target.

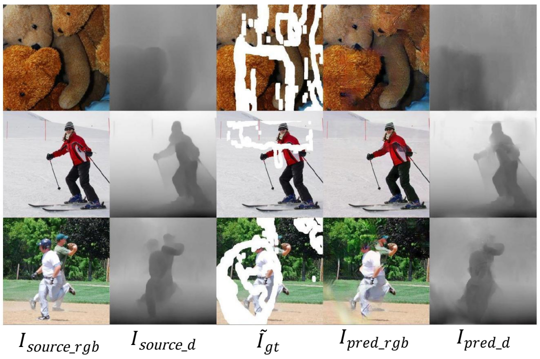

I_{source_rgb} I_{source_d} $\tilde{I}_{gt}$ I_{pred_rgb} I_{pred_d}

Figure 3: Results of RGBD image inpainting.

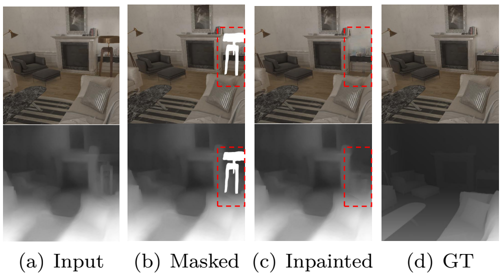
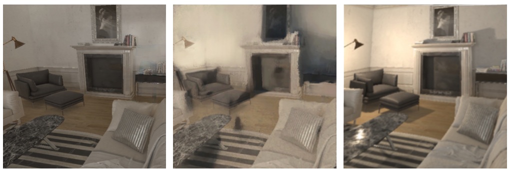

(a) Input (b) Masked (c) Inpainted (d) GT

Figure 4: Result of RGBD image inpainting.

4.2. Inpainting and Relighting in Indoor Scenes

Figure 4 shows the comparison of the RGBD-inpainted result image with the ground truth image for an indoor scene. For the inpainting part, the lamp on the right of the image was deleted in the scene. By comparing the RGB images of Fig. 4(a) and Fig. 4(c), we can clearly see that the lamp was removed and the area was successfully filled with the bookshelf and wall texture in the back. In the depth map, we can also see that the lamp was successfully removed. Note that the depth differs significantly between the resulting image and the ground truth even outside of the target region because the depth in the input image was estimated using MegaDepth and that in the ground truth was created by computer graphics.

For the inpainted result, we added a yellow point light source at the location of another lamp on the left side of the image. We can see in the resulting image (Fig. 5(b)) that the shadow of the fireplace is projected onto the wall on the right, and the lighting of the black sofa changes on the left of the image. Comparing the result with the ground truth (Fig. 5(c)), we can see that the shadow does not show the shape of the object well. This result is because the shadow projection depends on the estimated depth, but the accuracy of depth estimation from a single image is not very high. However, the cast shadow is adequately represented to the extent that the added light source position can be seen from the image.



(a) Input (b) Relit image (c) GT

Figure 5: Result of relighting.

4.3. Inpainting and Relighting In Outdoor Scenes

Figure 6 compares the RGBD-inpainted result image with the ground truth image for an outdoor scene. For the inpainting part, the street lights along the roadside were deleted in the outdoor scene. Here, we also marked some shadows as target regions for inpainting. The result should be compared with the ground truth as shown in Fig. 6. By comparing the RGB images of Fig. 6(a) and Fig. 6(c), we can clearly see that the street light was removed, and the area was successfully filled with the red building texture in the back. The depth map was also naturally inpainted to match the building wall shape.

For relighting, we removed the sky area by the sky mask and added a virtual sky to represent ambient light in Fig. 7. The impact of sky removal is discussed in the next section. Because sunlight can be seen as the only light source during the day in outdoor scenes, relighting is done primarily to redirect sunlight. In Fig. 7(a), by the shadow of the red building on the right, we can see that the sun is shining from almost directly above. The first row of Fig. 7(c) shows the result when the position of the sun was moved to the back of the red building. We can see that the shadow of the red building has become longer. The cast shadow covers the whole road in front of the red building. The second row of Fig. 7(c) shows the image in which the scene is illuminated by dark ambient light and city lights such as street lamps and vending machines in the evening and at night. The sky is the virtual one created by computer graphics. This way, various light source settings can convert a scene from day to evening or night.

Finally, we explain the addition of point lights. When adding a light source, the light source is not added directly to the image but to the generated model. For example, if we want to illuminate a street light, we find the position of the street light in the model and insert a point light source at that position. The characteristics and position of the light source can be defined freely.

As noted in the indoor scene, the estimated depths and their ground truth are different. Thus, the positions of the additional light sources will be completely different from their positions in the ground truth scene. In the outdoor scene, as shown in Fig. 8, the positions are also different. Therefore, we added the light sources in different positions between the estimated scenes and the ground truth but used the same parameters for the corresponding light sources.

4.4. Advantages of Optional Processing

Several optional processes can improve the quality of the lighting results. This section investigates the impact of removing cast shadows and sky from the original image and estimating albedo and illumination.

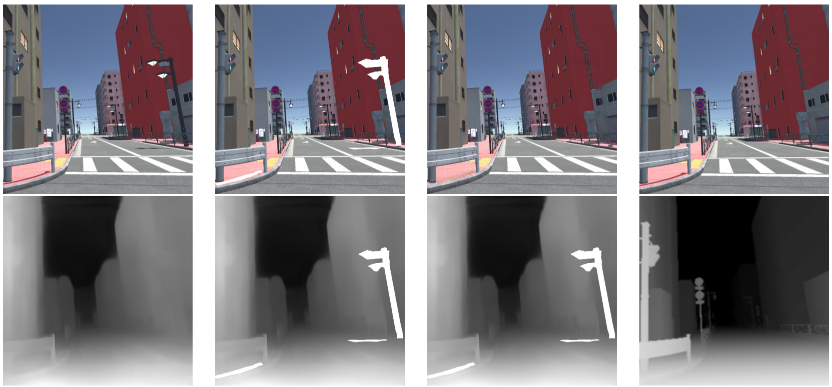
4.4.1. Shadow Removal

First, we discuss the performance of shadow removal. Experiments show that STCGAN [27] performs relatively well in removing shadows but has significant problems in shadow detection. On the other hand, ViSha [26] is superior to the STCGAN method for detecting a wide range of dark areas of an image as shadows. Therefore, we used ViSha for shadow detection and STCGAN for removal.

The removal results are shown in Fig. 9. From the figure, we can see that cast shadows are successfully removed from the photographed image. The results show that the shadow removal algorithm could effectively remove shadows automatically. However, the algorithm has some limitations. For example, the shadow removal algorithm used in this study, which combines ViSha and STCGAN, requires the image to be resized to a 256 x 256 image before feeding it into the network for calculation. As a result, the final output image has a lower resolution than the original image. Another problem is that the overall color of the image is slightly altered as cast shadows are removed.

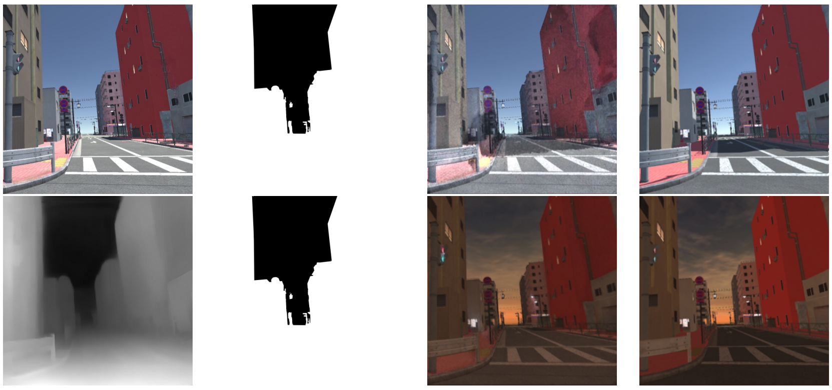
4.4.2. Sky Removal

Figure 10 shows the results of sky detection by PSPNet [28]. We can confirm that the sky areas are successfully detected from the results. Using one of the results, we examined the effect of sky removal on relighting. Figure 11 compares results relit with and without sky removal. Without sky removal, the depth estimation generated shapes in the sky as well, resulting in casting shadows in various places,



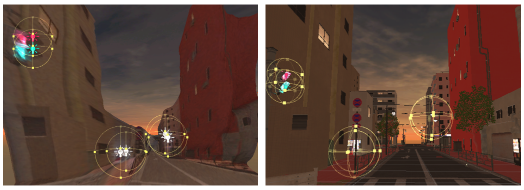
(a) Input (b) Masked (c) Inpainted (d) GT

Figure 6: Comparison of inpainting results and their ground truth in outdoor scene.



(a) Input (b) Sky mask (c) Relit (d) GT

Figure 7: Comparison of relighting results and their ground truth in outdoor scene.



(a) On the estimated model (b) On the GT model

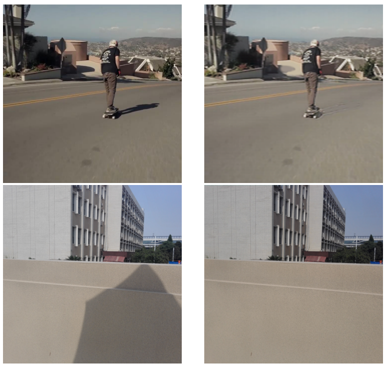
Figure 8: Comparison of relighting with addtional lights on the estimated model and the ground truth model

as shown in Fig. 11(b). On the other hand, the unnatural shadow does not appear as in (b) when the shadow is removed. It should be noted that a virtual sky is needed instead of the real sky in the image.

4.4.3. Albedo Map Estimation

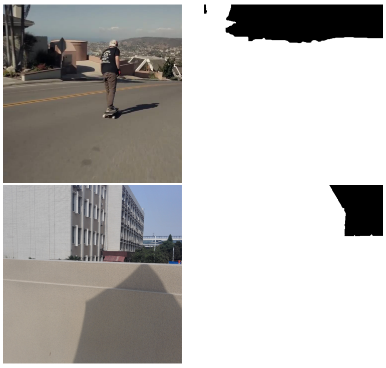
Estimating the albedo map on the image after removing shadows is also effective in keeping the texture and color of the same material relatively constant, as shown in Fig. 12. Figure 13 shows a comparison of the results of rendering the scene using the estimated albedo or original image under different light source conditions.

The results show that the relit image using the estimated albedo map is blurred since the map has lost some texture detail. On the other hand, better results could be obtained by directly relighting the original image. This result is because the estimated albedo map is not accurate enough. For example, as shown in Fig. 12, the right side of a building is always darker than the front side. Although it is expected that accurate albedo estimation would



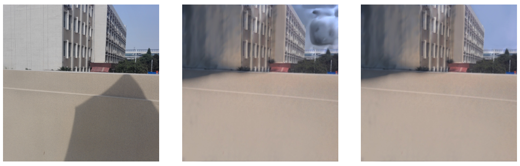
(a) Origin images (b) Shadow removed

Figure 9: Shadow removal using ViSha and STCGAN.



(a) Origin images (b) Sky masks

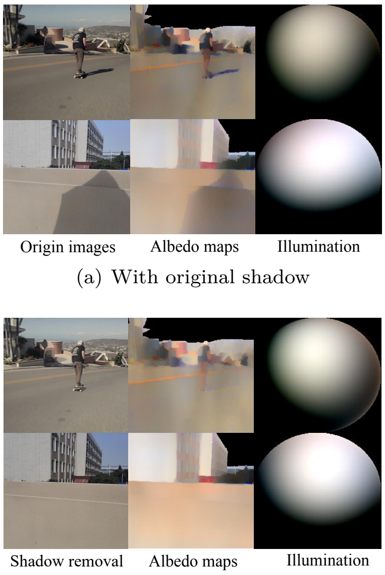
Figure 10: Results of sky detection by PSPNet.



(a) Origin images (b) Relit w/ sky (c) Relit w/o sky

Figure 11: Relighting results with origin sky and without origin sky.

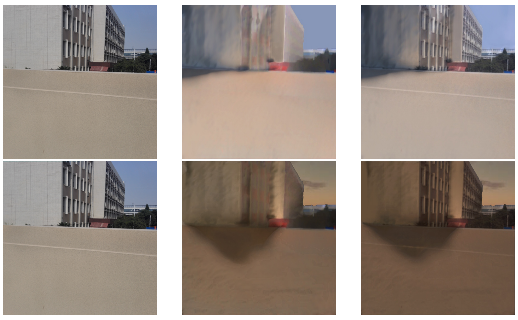
positively impact relighting, we confirmed that current albedo estimation is often inadequate for this task.



(a) With original shadow

(b) With shadow removal

Figure 12: Results of albedo maps estimation.



(a) Origin images (b) Relit w/ albedo (c) W/o albedo

Figure 13: Comparison of relit results with albedo maps.

5. Limitations

In this work, image relighting is mainly based on a simple rendering pipeline process. Compared to neural rendering, it has a high degree of control, such that the lighting effects can be largely controlled by knowing the scene geometry. However, such conventional rendering requires a great deal of prior intelligence to guide the computation of realistic lighting. For example, conditions such as the smoothness of the material are not considered in this study. Instead, neural rendering focuses on generating and processing shadow maps to achieve lighting effects, which significantly reduces the amount of

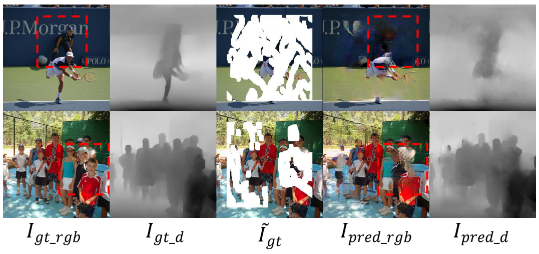


Figure 14: Failure cases of our image inpainting network.

prior information input.

Furthermore, since the estimated depth map is directly converted to a model in this study, the computation of shadows is inaccurate. For example, shadows of the same thickness as the object are not generated. However, this can be solved by using a neural network for shadow map generation due to its nature [31].

For inpainting, the network still has limitations. In some results, complex textured areas cannot be inpainted well. Also, the network cannot inpaint well when large areas are lost as shown in Fig. 14.

6. Conclusion

We proposed a method to relight the scene in an image while removing unwanted objects. The input RGB image and the mask image created by the user are the input to our system. First, the depth image is generated by using a neural network. Next, the RGBD is masked with the mask image and is inpainted using our proposed neural network. A mesh model is reconstructed from the inpainted depth image for the relighting process. In some scenes, cast shadow removal, sky region removal, and albedo estimation are selectively performed to suppress the effects of existing light sources. Through these processes, various types of relighting can be achieved from a single photograph while excluding the color and shape of unwanted objects. Future work includes improving the performance of inpainting and considering the use of neural network-based methods for relighting.

References

[1] P. Milgram, F. Kishino, A taxonomy of mixed reality visual displays, IEICE TRANSACTIONS on Information and Systems 77 (1994) 1321–1329.

[2] C. Zhang, T. Chen, A survey on image-based rendering - representation, sampling and compression, Signal Processing: Image Communication 19 (2004) 1–28.

[3] Y. Horry, K.-I. Anjyo, K. Arai, Tour into the picture: using a spidery mesh interface to make animation from a single image, in: SIGGRAPH, 1997, pp. 225–232.

[4] C. Loscos, G. Drettakis, L. Robert, Interactive virtual relighting of real scenes, IEEE Transactions on Visualization and Computer Graphics 6 (2000) 289–305.

[5] E. Eisemann, F. Durand, Flash photography enhancement via intrinsic relighting, ACM Transactions on Graphics 23 (2004) 673–678.

[6] T. Sun, Z. Xu, X. Zhang, S. Fanello, C. Rhemann, P. Debevec, Y.-T. Tsai, J. T. Barron, R. Ramamoorthi, Light stage super-resolution: continuous high-frequency relighting, ACM Transactions on Graphics 39 (2020) 1–12.

[7] M. El Helou, R. Zhou, S. Susstrunk, R. Timofte, Ntire 2021 depth guided image relighting challenge, in: CVPR, 2021, pp. 566–577.

[8] A. Sanin, C. Sanderson, B. C. Lovell, Shadow detection: A survey and comparative evaluation of recent methods, Pattern recognition 45 (2012) 1684–1695.

[9] M. Bertalmio, G. Sapiro, V. Caselles, C. Ballester, Tour into the picture: using a spidery mesh interface to make animation from a single image, in: SIGGRAPH, 2000, pp. 417–424.

[10] S. Esedoglu, J. Shen, Digital inpainting based on the mumford–shah–euler image model, European Journal of Applied Mathematics 13 (2002) 353–370.

[11] H. Li, W. Luo, J. Huang, Localization of diffusion-based inpainting in digital images, IEEE transactions on information forensics and security 12 (2017) 3050–3064.

[12] A. Criminisi, P. Pérez, K. Toyama, Region filling and object removal by exemplar-based image inpainting, IEEE Transactions on Image Processing 13 (2004) 1200–1212.

[13] Y. Wexler, E. Shechtman, M. Irani, Space-time completion of video, IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (2007) 463–476.

[14] T. Ruzic, A. Pizurica, Context-aware patch-based image inpainting using markov random field modeling, IEEE transactions on image processing 24 (2015) 444–456.

[15] D. Ding, S. Ram, J. J. Rodríguez, Image inpainting using nonlocal texture matching and nonlinear filtering, IEEE Transactions on Im-

age Processing 28 (2018) 1705–1719.

[16] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, A. A. Efros, Context encoders: Feature learning by inpainting, in: CVPR, 2016, pp. 2536–2544.

[17] Y. Zeng, J. Fu, H. Chao, B. Guo, Learning pyramid-context encoder network for high-quality image inpainting, in: CVPR, 2019, pp. 1486–1494.

[18] H. Liao, G. Funka-Lea, Y. Zheng, J. Luo, S. Kevin Zhou, Face completion with semantic knowledge and collaborative adversarial learning, in: ACCV, Springer, 2018, pp. 382–397.

[19] C. Zheng, T.-J. Cham, J. Cai, Pluralistic image completion, in: CVPR, 2019, pp. 1438–1447.

[20] W. Liu, X. Chen, J. Yang, Q. Wu, Robust color guided depth map restoration, IEEE Transactions on Image Processing 26 (2016) 315–327.

[21] T.-J. Mu, J.-H. Wang, S.-P. Du, S.-M. Hu, Stereoscopic image completion and depth recovery, The Visual Computer 30 (2014) 833–843.

[22] R. Fujii, R. Hachiuma, H. Saito, Joint inpainting of rgb and depth images by generative adversarial network with a late fusion approach, in: ISMAR Adjunct, 2019.

[23] Z. Li, N. Snavely, Megadepth: Learning single-view depth prediction from internet photos, in: CVPR, 2018, pp. 2041–2050.

[24] K. Nazeri, E. Ng, T. Joseph, F. Z. Qureshi, M. Ebrahimi, Edgeconnect: Generative image inpainting with adversarial edge learning, arXiv preprint arXiv:1901.00212 (2019).

[25] J. Johnson, A. Alahi, L. Fei-Fei, Perceptual losses for real-time style transfer and super-resolution, in: ECCV, Springer, 2016, pp. 694–711.

[26] Z. Chen, L. Wan, L. Zhu, J. Shen, H. Fu, W. Liu, J. Qin, Triple-cooperative video shadow detection, in: CVPR, 2021, pp. 2715–2724.

[27] J. Wang, X. Li, J. Yang, Stacked conditional generative adversarial networks for jointly learning shadow detection and shadow removal, in: CVPR, 2018, pp. 1788–1797.

[28] H. Zhao, J. Shi, X. Qi, X. Wang, J. Jia, Pyramid scene parsing network, in: CVPR, 2017, pp. 2881–2890.

[29] Y. Yu, W. A. Smith, Inverserendernet: Learning single image inverse rendering, in: CVPR, 2019, pp. 3155–3164.

[30] G. Liu, F. A. Reda, K. J. Shih, T.-C. Wang, A. Tao, B. Catanzaro, Image inpainting for irregular holes using partial convolutions, in: ECCV, 2018, pp. 85–100.

[31] D. Griffiths, T. Ritschel, J. Philip, Outcast: Outdoor single-image relighting with cast shadows, in: Computer Graphics Forum, volume 41, Wiley Online Library, 2022, pp. 179–193.