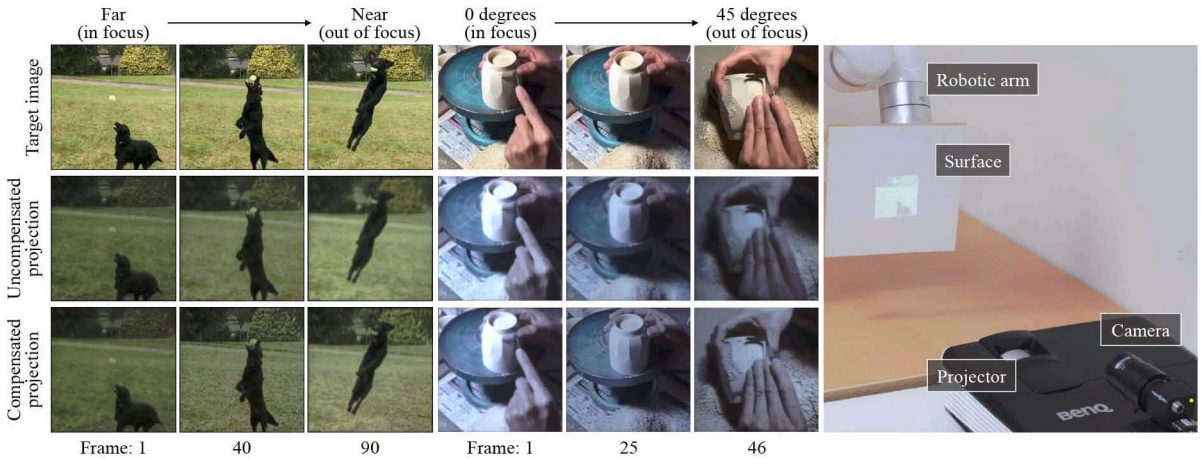
Online Projector Deblurring Using a Convolutional Neural Network

Yuta Kageyama, Daisuke Iwai, *Member, IEEE*, and Kosuke Sato, *Member, IEEE*



(a) Translation movement (b) Rotation movement (c) Experimental setup

Fig. 1. The proposed projector deblurring technique compensates movie contents for defocus blur artifacts using a deep neural network for dynamic projection mapping. The left and middle images show the target images (top) as well as the projected results of the target images (middle) and of the compensation images (bottom) that were computed by the proposed network. The projection surface was (a) translated from far (in focus) to near (out of focus) or (b) rotated from 0 degrees (in focus) to 45 degrees (out of focus) along the yaw axis by using a robotic arm shown in the right image. We observed that the details of the target images were preserved in the projected results of the proposed technique but were missing in the projected results of the direct projection of the target images.

Abstract—Projector deblurring is an important technology for dynamic projection mapping (PM), where the distance between a projector and a projection surface changes in time. However, conventional projector deblurring techniques do not support dynamic PM because they need to project calibration patterns to estimate the amount of defocus blur each time the surface moves. We present a deep neural network that can compensate for defocus blur in dynamic PM. The primary contribution of this paper is a unique network structure that consists of an extractor and a generator. The extractor explicitly estimates a defocus blur map and a luminance attenuation map. These maps are then injected into the middle layers of the generator network that computes the compensation image. We also propose a pseudo-projection technique for synthesizing physically plausible training data, considering the geometric misregistration that potentially happens in actual PM systems. We conducted simulation and actual PM experiments and confirmed that: (1) the proposed network structure is more suitable than a simple, more general structure for projector deblurring; (2) the network trained with the proposed pseudo-projection technique can compensate projection images for defocus blur artifacts in dynamic PM; and (3) the network supports the translation speed of the surface movement within a certain range that covers normal human motions.

Index Terms—Projector deblurring, dynamic projection mapping, deep neural network

1 Introduction

Projection mapping (PM) is a major approach to realize spatial augmented reality (SAR), which seamlessly merges the real and cyber worlds by optically superimposing computer-generated graphics onto physical surfaces [8, 15]. It has been applied in various fields, such as medicine [37], industrial design [44], online conferencing [23], office work [25, 30], and entertainment [26, 31]. Due to the recent advances of high-speed and low-latency projector hardware [49], the latest research tends toward dynamic PM, where the geometrical relationship between a projection target and a projector changes during the projection [5, 32, 35, 42, 45]. In addition, there is an emerging optical see-through, head-mounted display that applies a dynamic PM framework, where a head-mounted screen is projected by a pan-tilt projector [22]. To control the appearance of the target surface, accurate alignment of the projected image onto the target is crucial; and thus, geometric registration of the projector has been the prime technical challenge [8]. On the other hand, defocus blur is also a critical problem in dynamic PM. Because the lens aperture of a projector is normally designed to be large (i.e., with a small f-stop) to achieve bright projection, a projector's depth of field (DoF) is narrow. When a target moves and its distance from the projector changes, the projected result is significantly blurred. Therefore, defocus blur compensation is as important as geometric registration in dynamic PM.

Defocus blur is mathematically modeled as the convolution of a projection image and a point spread function (PSF). The PSF of each projector pixel represents its impulse response or the intensity distribution of its projected result. Past studies realized projector deblurring by computing the projection images using deconvolution [9, 13, 38]. To achieve effective defocus blur compensation, these techniques require accurate PSF estimation. Because the PSF varies according to the distance between the projector lens and the projection surface, it varies spatially in PM, where the projection surface is potentially nonplanar.

- *Yuta Kageyama, Daisuke Iwai, and Kosuke Sato are with Graduate School of Engineering Science, Osaka University. E-mail: kageyama@sens.sys.es.osaka-u.ac.jp, {daisuke.iwai, sato}@sys.es.osaka-u.ac.jp.*
- *Daisuke Iwai is with PRESTO, Japan Science and Technology Agency.*

Manuscript received 6 Sept. 2021; revised 3 Dec. 2021; accepted 7 Jan. 2022. Date of publication 15 Feb. 2022; date of current version 29 Mar. 2022. Digital Object Identifier no. 10.1109/TVCG.2022.3150465

Previous studies estimated the PSFs by projecting a calibration pattern, which is either a spatial code pattern such as a dot pattern [9, 13, 52] or an original target image [28, 38], and capturing the projected results using a camera. The estimated PSFs are valid for as long as the projector and the surface do not move. If they move, however, the PSFs must be estimated again. Therefore, in dynamic PM, the calibration pattern must be frequently projected onto the surface, which significantly degrades the augmented reality experience of users. While many approaches to mitigating the defocus blur in PM have been proposed (see Sect. 2.1), to the best of our knowledge, no technical solution has been presented yet for the projector defocus problem in dynamic PM.

This paper proposes a projector deblurring technique that does not require projection of the calibration pattern even when the projection surface moves. As the first attempt towards the projector deblurring in dynamic PM, we start with a simple assumption that the surface is uniformly white and completely diffuse. The key insight exploited in the technique is that the geometric relationship between the projector and the projection surface does not vary significantly within a video frame (i.e., 1/60 sec in most current video projectors). Therefore, our method computes the compensation image of the current frame using the projected result of the previous frame captured with a camera. Specifically, we applied a deep convolutional neural network (CNN) to generate the compensation image. As the prime contribution of this research, we devised an effective network structure for projector deblurring, which has two parts: an extractor and a generator. In each frame, the extractor, which consists of two subnetworks, takes a pair of the projection image of the previous frame and its projected result as the input. The first subnetwork estimates a defocus blur map that represents how much each projector pixel is defocused on the surface. The second subnetwork estimates a luminance attenuation map that represents the degree of reduction of the captured luminance of the projected result compared to that of the target luminance due to the inverse square law of light intensity. These two maps are then injected into the middle layers of the generator network, which takes the original target image of the current frame as the input and computes the compensation image to be projected at the current frame. We also propose to synthesize physically plausible training data to avoid laborious and time-consuming projection data collection. In particular, our pseudo-projection technique generates the projected results by simulating the defocus blur based on the thin-lens model and by simulating the luminance reduction based on the inverse-square law with respect to the depth. Considering actual PM scenarios where the geometric registration of the projector and the camera is potentially inaccurate, we also incorporated warping of the generated image into our pseudo-projection framework. Through a simulation-based comparison, we show that the proposed network structure is more suitable for compensating dynamic projection contents for defocus blur than a simple, single-network structure. Using a physical projector-camera system, we demonstrate that our projector deblurring technique can compensate for the defocus blur in an actual dynamic PM scenario.

To summarize, our primary contributions from this study are as follows:

- We introduce a CNN-based projector deblurring technique that generates a compensation image for defocus blur artifacts in the projected result in a dynamic PM scenario without requiring offline PSF estimation;
- We find that the combination of extractor subnetworks and a generator subnetwork is the effective structure for projector deblurring, where the outputs of the extractor (the defocus blur map and the luminance attenuation maps) are incorporated into the middle layers of the generator;
- We design a pseudo-projection framework to synthesize physically plausible training data, considering inaccurate geometric registration between the projector and the camera; and
- We demonstrate the projector deblurring achieved by the proposed system through a physical dynamic PM experiment.

Details on the implementation can be found at the GitHub repository[1].

2 Related Studies

There are two major research topics related to this study: projector deblurring and deep learning for radiometric compensation in PM. In this section, we introduce previous studies on these topics and state our contributions compared to them.

2.1 Projector deblurring

Previous PM techniques that tackled the defocus blur artifacts can be categorized into single-projector and multiple-projectors approaches. The goal of the single-projector approach is to generate a compensation image that, when projected, will closely resemble the target image, which is not blurred. To this end, the single-projector techniques solve the inverse problem of projector defocus, which is modeled by a convolution between a projection image and the pixel-dependent PSFs [11]. Researchers have explored several deconvolution approaches. For instance, Brown et al. [9] and Oyamada and Saito [38] proposed compensation techniques based on the Wiener filter. Although these were computationally efficient, their compensation results tended to suffer from ringing artifacts. Other researchers achieved projector deblurring with fewer artifacts by applying a constrained optimization technique [52] or an inverse light transport matrix technique [50], though they were computationally expensive. Kageyama et al. balanced the trade-off between the deblurring accuracy and the computational complexity using a deep neural network (DNN) [28]. Grosse et al. also balanced the trade-off by applying a coded aperture to the projector optics, which preserves the high-frequency components of a projected image more than do normal circular apertures and, consequently, reduces the ringing artifacts caused by Wiener filtering [13].

In these aforementioned techniques, the pixel-dependent PSFs must be estimated for the deconvolution. The estimation is done by projecting either dot patterns [9, 13, 50, 52] or original images (i.e., target images) [28, 38] in advance. Therefore, unblurred images cannot be continuously displayed on a moving projection surface, where the PSFs vary in time. One possible solution is to apply a traditional fast PSF estimation method and deblurring technique [3, 27] using the projection information of the previous frame and the target image of the current frame. This idea is similar to our proposed method but may not work well due to the misalignment between the projector and the camera pixels in actual dynamic PM scenarios. These limitations had been overcome by applying a fast focal sweep projection technique with an electrically focus tunable lens (ETL), by which the PSF of each pixel becomes uniform over a wide depth range [24]. Another solution also applied the ETL by controlling it to keep focusing on a moving target whose position was tracked by a depth sensor [47]. In these techniques, the PSFs need not be estimated online even when the projection surface moves. However, to the best of our knowledge, there is currently no ETL device that can balance a sufficiently quick response for the focus control with a sufficiently large aperture for image projection.

The multiple-projectors approach achieves all in-focus projection by positioning multiple projectors such that their projected images overlay each other and their DoFs cover the entire projection surface. In early methods, a projector is selected for each surface area whose PSF preserves the high-spatial-frequency components of the target image [6, 34]. Recent studies have proposed more advanced techniques in which the projection images for the multiple projectors are jointly optimized so that the appearance of the overlaid projected results resembles the target image [4, 41]. The image quality of the projected result is better in the multiple-projectors techniques than in the single-projector techniques. However, a multi-projection system significantly increases the cost and requires complex geometric and photometric calibrations. Thus, its application field is still very limited.

[1] https://github.com/kagechan5/Online-Projector-Deblurring-Using-a-Convolutional-Neural-Network

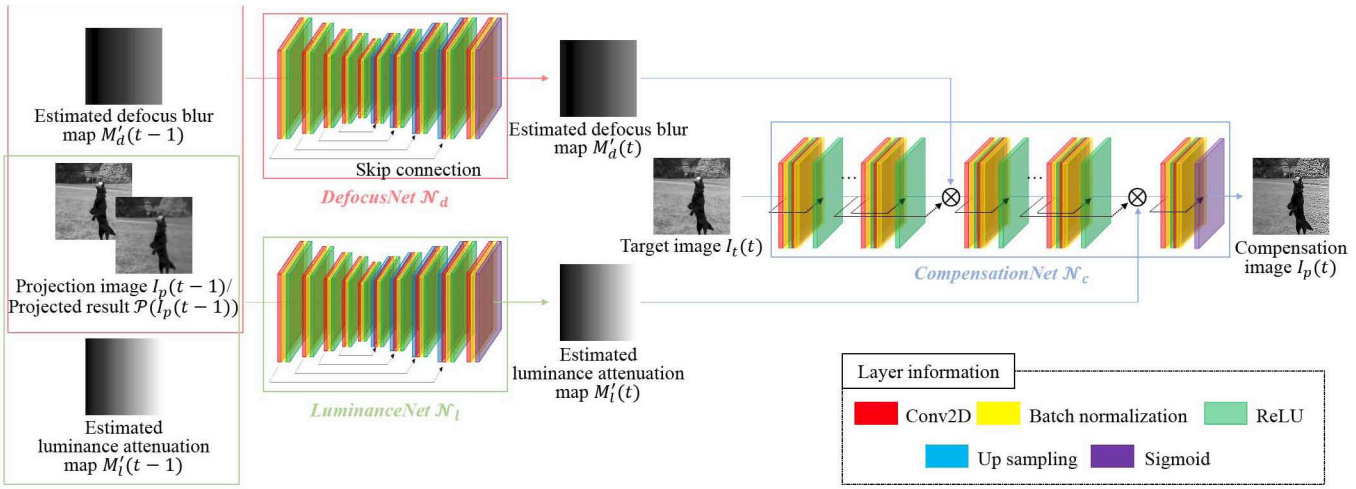


Fig. 2. An overview of our online blur compensation framework that has two parts: an extractor and a generator. The two subnetworks of the extractor, DefocusNet and LuminanceNet, have the same U-Net-like structure [39] but do not share the learnable weights. From the information in the previous frame $t-1$, the DefocusNet estimates the defocus blur map $M'_d(t)$, and LuminanceNet estimates the luminance attenuation map $M'_l(t)$. These maps represent the degree of defocus blur and luminance attenuation per pixel, respectively. The generator's subnetwork generates the compensated projection image $I_p(t)$ from the target image $I_t(t)$ by incorporating the maps into the middle layers of the network.

2.2 Deep learning for radiometric compensation

A projected result on an arbitrary surface appears different from its original image. The color is modulated by the texture of the surface, and the high-spatial-frequency components are reduced by the subsurface scattering, interreflection, and defocus blur. Radiometric compensation techniques have been intensively explored in the last two decades to mitigate the image quality degradation in PM [7, 15]. The latest trend in this research field is the application of deep learning technologies. Huang et al. proposed an end-to-end projector photometric compensation network [19]. By projecting and capturing hundreds of texture images on a projection surface, the network implicitly learns the per-pixel complex reflectance property of the surface. Once trained, the network generates a projection image from an original image such that the projected result does not suffer from photometric distortions caused by the spatially varying surface reflectance properties. The network was improved to compensate for both photometric and geometric distortions [18, 21] as well as for global illumination effects [20]. These networks outperform classical technologies with regard to radiometric compensation. However, they need to be trained for each projection surface, in which multiple textures need to be projected in advance.

Kageyama et al. proposed an end-to-end compensation network for projector deblurring [28]. Once the weights of the network are trained, they can be used for any projection surface. However, the network still requires projection of the target image and capture of its projected result before it can compute the compensation image for each surface. Therefore, it does not support dynamic projection contents (e.g., movies) and cannot properly work for dynamic projection surfaces. In summary, the previous radiometric compensation techniques based on deep learning are not suitable for dynamic PM scenarios.

2.3 Our contribution

The prime contribution of this paper is the realization of a projector deblurring technique in a single-projector approach that works in dynamic PM. Based on an observation that the depth of the projection surface does not significantly change within a video frame (i.e., 1/60 sec), our technique synthesizes the compensation image using the projected information in the previous frame. We designed a DNN structure that synthesizes a compensation image from the target image of the current frame as well as from the projection image of the previous frame and its projected result. In particular, we found that the combination of two network modules (an extractor and a generator) has better compensation performance than a simple single-network structure. We also propose a pseudo-projection technique for synthesizing a projected result from a projection image and a depth map of a surface in order to train the network. We show that the network weights trained by the synthesized data are useful for projector deblurring in physical setups.

3 Projector Deblurring Network

We propose a DNN that synthesizes a projection image to compensate for defocus blur even in dynamic PM scenarios. This section describes our network and our loss function, which are designed to minimize the difference between a target image and the projected result of the compensation image.

3.1 Overview

We assume that a projection surface is (1) uniformly white and completely diffuse, (2) observed with a camera, and (3) within the camera's DoFs. Although various optical phenomena can degrade the image quality of a projected result, we found from our preliminary investigation that two of these optical phenomena are dominant factors in the assumed situation. The first optical phenomenon is the projector's defocus blur, which attenuates the high-spatial-frequency components of the projected result according to the distance from the focal plane. The second optical phenomenon is the attenuation of the luminance of the captured projected result according to the distance of the surface from the camera (i.e., according to the inverse square law of light). We designed our network to mitigate the image quality degradation caused by the defocus blur in the projected result without suffering from the luminance attenuation artifacts.

Figure 2 shows the whole structure of our proposed projector deblurring network. In our preliminary investigation, we observed that the projection surface did not significantly move within each video frame in most of the dynamic PM scenarios. Thus, we estimated the extent of the occurrence of the defocus blur and the luminance attenuation of the projected result in the previous frame, and we used that information to generate the compensation image for the current frame. We applied a network structure with two parts, an extractor and a generator, rather than a single network to explicitly estimate the defocus blur and the luminance attenuation. Specifically, the extractor had two subnetworks, one of which estimates the amount of defocus blur, and the other, of luminance attenuation for each projector pixel. The estimated defocus blur map and the luminance attenuation map were then injected into the generator subnetwork, which synthesized the projection image that compensated for the image degradation. We explicitly separated the network structure so that the texture of the previous frame would not affect the compensation image in the current frame.

Note that we designed our network to compensate for defocus blur caused by spatially varying PSFs, and it properly worked even in a case wherein two consecutive video frames were not similar (e.g., a scene change occurred between them). Without loss of generality, we assumed that the projector and the camera shared the same field of view (FoV). This assumption allowed us to model the projector's PSF without having to consider its distortion on a freeform or tilted surface due to the different perspectives of the two devices. We achieved the FoV sharing in an actual projector-camera setup by applying a beam-splitter or by geometrically transforming the captured image using the pose relationship between them.

3.2 Extractor

The two subnetworks in the extractor independently estimate the defocus blur and the luminance attenuation. We refer to the subnetwork for the defocus blur estimation as DefocusNet ($\mathcal{N}_d$), and that for the luminance attenuation, as LuminanceNet ($\mathcal{N}_l$). As shown in Fig. 2, these subnetworks are independent to enable them to extract different features from each network. They apply the same U-Net [39] structure but do not share the weights and are not connected to each other.

Each subnetwork takes as input a projection image and its projected result in the preceding frame. Considering the interframe consistency, we also feed each subnetwork with its previous output. Then, DefocusNet and LuminanceNet output the defocus blur map and the luminance attenuation map, respectively. As a reasonable assumption that was applied in most previous studies [1, 24, 34], we considered the PSF of a projected pixel a Gaussian function. Then, each pixel of the defocus blur map represents the variance of the PSF of the corresponding projected pixel. A captured projected pixel becomes darker when the pixel is farther from the camera (i.e., based on the inverse square law of light). Therefore, each pixel of the luminance attenuation map represents the distance of the corresponding projected pixel from the camera. Then, these subnetworks are modeled as in the following equations:

$$M'_d(t) = \mathcal{N}_d(I_p(t-1), \mathscr{P}(I_p(t-1)), M'_d(t-1)), \quad (1)$$

$$M'_l(t) = \mathcal{N}_l(I_p(t-1), \mathscr{P}(I_p(t-1)), M'_l(t-1)), \quad (2)$$

where M'_d and M'_l are the estimated defocus blur map and the luminance attenuation map, respectively; I_p is the projection image and $\mathscr{P}(I_p)$ represents its projected result; and t and $t-1$ indicates the frame numbers.

We trained DefocusNet and LuminanceNet to accurately estimate the amount of defocus blur and luminance attenuation in the projected result for each projector pixel. Suppose that the ground truth map of the PSF variances of the projected pixels and of their distances from the camera are M_d and M_l, respectively. Then, we used the following $\mathscr{L}_d$ and $\mathscr{L}_l$ as the loss functions in the training of DefocusNet and LuminanceNet, respectively:

$$\mathscr{L}_d = \mathscr{L}(M'_d(t), M_d(t)), \quad (3)$$

$$\mathscr{L}_l = \mathscr{L}(M'_l(t), M_l(t)), \quad (4)$$

$$\mathscr{L}(i,j) = \mathscr{L}_{\ell_1}(i,j) + \lambda_1 \mathscr{L}_{TV}(i,j), \quad (5)$$

where $\mathscr{L}_{\ell_1}(i,j)$ is a loss function that uses the ℓ_1 norm of the differences between i and j. In addition, we applied the total variation [40] loss $\mathscr{L}_{TV}(i,j)$ for the regularization of the estimated maps. λ_1 is a coefficient that balances the two functions.

3.3 Generator

Given the estimated defocus blur map M'_d and the luminance attenuation map M'_l, we generated a compensation image from the target image I_t using the generator subnetwork. We denote the compensation image as I_p because it is the projection image of the next frame. We refer to the network as CompensationNet ($\mathcal{N}_c$), and we use ResNet [17] as its backbone network. To use the per-pixel information of the two maps in the compensation image generation, we applied an attention mechanism [12] for incorporating the maps into CompensationNet. Specifically, we injected M'_d and M'_l into the middle layers of CompensationNet with the Hadamard product rather than giving them to the first layer of the network, as shown in Fig. 2. Thus, we modeled the network as in the following equation:

$$I_p(t) = \mathcal{N}_c(I_t(t), [M'_d(t)], [M'_l(t)]), \quad (6)$$

where the brackets represent the injection of the estimated maps into the middle layers.

We trained CompensationNet to generate a compensation image whose projected result resembles the target image for human observers. Therefore, the following function can be considered the loss in the training:

$$\mathscr{L}_{ssim}(\mathscr{P}(I_p(t)), I_t(t)), \quad (7)$$

where $\mathscr{L}_{ssim}(i,j)$ represents the structural similarity (SSIM) [48] loss that computes the difference between i and j in the human visual perception space. However, due to the limited peak luminance of a current projector device, the full luminance range of the target image is not always reproducible in a projected result [14, 46]. To mitigate the clipping errors caused by this limitation, we linearly normalized the target image pixel values into the reproducible range. Thus, we used the following $\mathscr{L}_c$ as the loss function in the training of CompensationNet:

$$\mathscr{L}_c = \mathscr{L}_{ssim}(\mathscr{P}(I_p(t)), kI_t(t)), \quad (8)$$

where k ($0 \leq k \leq 1$) represents the normalization constant that is uniformly applied to the pixel values over the target image, and k is the minimum luminance attenuation factor (i.e., the largest amount of attenuation) in the captured image.

3.4 Training strategy

A straightforward (or naïve) method of training the proposed network is to update all the weights in the network by flowing the data depicted in Fig. 2. Specifically, the estimated maps from DefocusNet and LuminanceNet (i.e., M'_d and M'_l) are directly injected into CompensationNet in the training. However, we expect this method to be unstable and the weights not to converge in a reasonable time frame. In the early stage of the training, DefocusNet and LuminanceNet did not output correct maps. Thus, updating the weights of CompensationNet did not make much sense. Therefore, we applied another method that separately trained the three networks. Specifically, we used the ground truth of the defocus blur map M_d and the luminance attenuation map M_l to train CompensationNet instead of the estimated maps, M'_d and M'_l (Fig. 3).

4 Dataset Synthesis

To train the proposed network, we had to prepare a large set of target images I_t, the ground truth of the defocus blur maps M_d, and that of the luminance attenuation maps M_l. In addition, the training required the projected results of the projection images I_p. However, it was impractical to obtain them using actual PM and capturing setups with a large number of projection surfaces of various shapes. Therefore, we synthesized the dataset from a set of target images and a set of depth images that represented the shapes of the projection surfaces. We performed the dataset synthesis in a virtual space with a virtual projector and a virtual camera. Based on the assumption described in Sect. 3.1, the virtual projector and the virtual camera had the same FoV.

4.1 Computational model of projector blur

As shown in Fig. 4, based on a thin-lens model, a light point emitted from the imaging plane of a projector is observed as a circle on a projection surface located v away from the projector lens. The diameter of the blur circle b can be computed using geometrical similarity as:

$$b = |d(\frac{v}{s} - 1)|, \quad (9)$$

where d and s are the diameter of the projector's aperture and the focusing distance, respectively. A pixel on the imaging plane is not an ideal point light source; therefore, the PSF of the projected pixel cannot

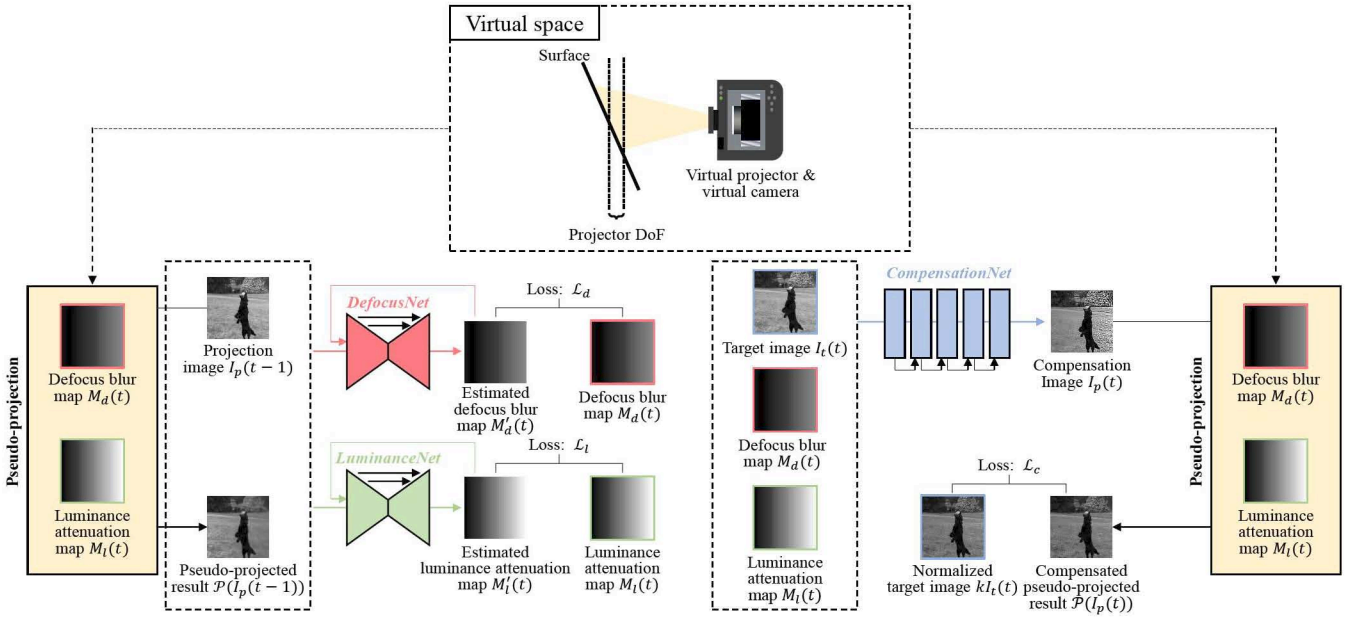


Fig. 3. Our training strategy, which separated the training of the subnetworks rather than jointly optimizing the entire network. In our training of DefocusNet and LuminanceNet, we applied pseudo-projection, as described in Sect. 4, which requires the defocus blur map and the luminance attenuation map. These maps were generated by the surface depth map (dashed arrows).

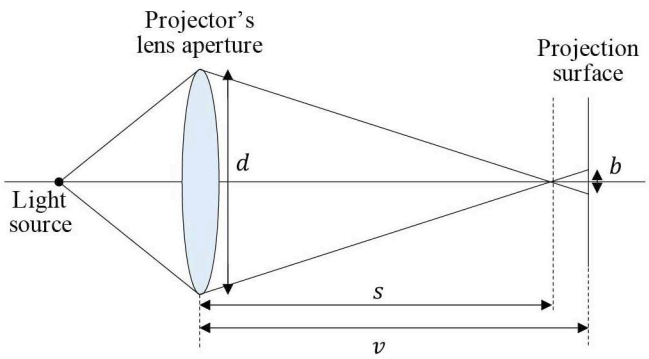


Fig. 4. Thin-lens model for computing the PSF of a projected pixel.

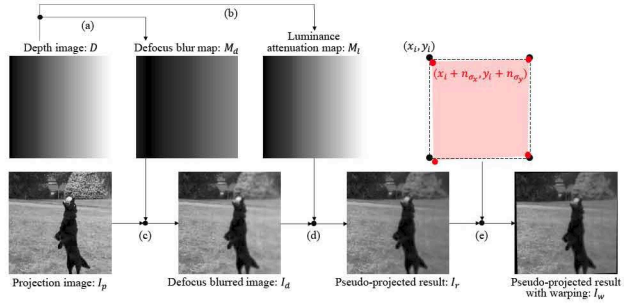


Fig. 5. The process flow of the proposed pseudo-projection technique.

be represented as a pillbox function but is generally approximated as the following Gaussian function [1, 24, 34]:

$$PSF(r,b) = \frac{2}{\pi b^2}\exp(-\frac{2r^2}{b^2}), \tag{10}$$

where r is the distance from the blur center. When an image I_p is projected from a virtual projector onto a nonplanar surface, the projected appearance I_r captured by a virtual camera can be computed using the following convolution:

$$I_r = I_p \otimes PSF + n, \tag{11}$$

where $\otimes$ is the convolution operator and n is a Gaussian noise.

4.2 Synthesis of defocus blur map and luminance attenuation map

We generated the defocus blur map M_d from the depth map of a projection surface that represented the distance from the virtual projector to the surface at each projector pixel (Fig. 5(a)). Specifically, based on Eq. 9, we computed the defocus blur map as follows:

$$M_d(x,y) = |d(\frac{\lambda_2 D(x,y) + \lambda_3}{s} - 1)|, \tag{12}$$

where $D(x,y)$ is the depth value ($0 \le D(x,y) \le 1$) at the projector pixel coordinate of (x,y). The focusing distance s is randomly selected from the predefined range $\lambda_4 \le s \le \lambda_5$.

We generated the luminance attenuation map M_l also from the depth map (Fig. 5(b)). As described in Sect. 3.2, each pixel of the luminance attenuation map $M_l(x,y)$ represents the distance of the corresponding projected pixel from the camera. Therefore, we computed it as follows:

$$M_l(x,y) = \lambda_6 D(x,y) + \lambda_7, \tag{13}$$

where λ_6 and λ_7 are the predefined scaling factor and the bias, respectively.

4.3 Pseudo-projection for synthesis of projected results

We developed a pseudo-projection technique to synthesize the projected results, considering both the defocus blur and the luminance attenuation. Suppose that I_p is a projection image; then the defocus blurred image I_d can be computed using Eq. 11 as:

$$I_d(x,y) = (I_p \otimes PSF(M_d))(x,y) + n_\sigma(x,y), \tag{14}$$

where n_σ is a Gaussian noise with the standard deviation value of σ (Fig. 5(c)).

Next, we applied the luminance attenuation to I_d to obtain the projected result I_r. According to our assumption of the FoV sharing between the virtual projector and the virtual camera, the apparent size of each projected pixel captured by the virtual camera does not change with respect to the depth variation of the projection surface. On the other hand, the luminous flux that is emitted from the projected pixel and incident into the virtual camera's lens is inversely proportional to the square of the distance from the pixel. Thus, the luminance of the projected pixel captured by the virtual camera attenuates according to the inverse square law. Therefore, the luminance attenuation factor at each pixel is $\frac{1}{(M_l(x,y))^2}$. Suppose the pseudo-projection operator is denoted as $\hat{\mathscr{P}}$; then the projected result is generated as:

$$I_r(x,y) = \hat{\mathscr{P}}(I_p(x,y)) = \frac{1}{(M_l(x,y))^2} I_d(x,y). \tag{15}$$

Figure 5(d) shows this process. Note that the normalization constant in the computation of the loss of CompensationNet is $k = \min_{x,y} \frac{1}{(M_l(x,y))^2}$ (see Eq. 8).

4.4 Warping in pseudo-projection

The pseudo-projection process has been implemented so far assuming the FoV of the virtual projector and that of the virtual camera are identical. However, achieving the perfect alignment in an actual projector-camera setup is difficult [2]. If the projected results in a dataset are synthesized as described above and used to train the proposed network, a slight misalignment in an actual setup potentially causes significant artifacts in the compensation result. Therefore, we propose the use of a geometric warping technique to simulate the misalignment and to incorporate it into the pseudo-projection process.

Specifically, we applied a homography transformation to warp the synthesized projected result I_r, which was originally a rectangle (Fig. 5(e)). Suppose that the coordinate values of the four corners are (x_i, y_i) $(i = 1,2,3,4)$. Then, we warp the projected result so that the coordinate values after warping are $(x_i + n_{\sigma_x}, y_i + n_{\sigma_y})$. The warped image is clipped by the original rectangle area. We fill non-existing pixels with the pixel value of zero (i.e., black). We denote the warping process as $\mathscr{W}$. Thus, the whole process of obtaining the pseudo-projected result with warping I_w can be written as:

$$I_w = \mathscr{W}(I_r) = \mathscr{W}(\hat{\mathscr{P}}(I_p)). \tag{16}$$

5 Experiment

We evaluated the proposed network both in a simulation and a physical dynamic PM setup. In this section, the details of the training are described, followed by the simulation experiment that was conducted to evaluate the validity of the proposed network and the training strategy. Then, the results of the physical experiments, which were conducted to check if the proposed network would work in an actual dynamic PM scenario, are introduced.

5.1 Training details

To train our network, we prepared 20,000 videos of 21 video frames, each of which was generated by connecting three different video files randomly selected from Vimeo-90K dataset [51], where each video file consists of seven frames. We connected the different video files to ensure that each video sequence contained rapid changes between the connected frames. Then we randomly selected 2,748 depth images from the NYU Depth Dataset [36], which were used to represent the shapes of projection surfaces. The videos and the depth images were scaled and clipped into 256×256 pixels. Each video file was paired with one of the depth images. Thus, we prepared 20,000 pairs of video files and depth images.

To optimize the network, we utilized Adam [29] with a learning rate of 1e-3 and momentum parameters of $\beta_1 = 0.9$ and $\beta_2 = 0.999$. All learnable parameters were initialized using the method of He et al. [16]. The 20,000 pairs were divided into 1,000 mini-batches, each of which contained 20 pairs. Because training using a large set of video files normally takes a relatively long time, it was not feasible to repeat our training for multiple epochs. Therefore, the epoch number of our training was one. We trained our network using a shared workstation (GPU: NVIDIA RTX A6000, GPU memory: 48 GB, CPU: Intel Xeon Platinum 8260, CPU memory: 768 GB). The training took 630 min. The predefined parameters were set as $\lambda_1 = 0.2$, $\lambda_2 = 5$, $\lambda_3 = 1$, $\lambda_4 = 1$, $\lambda_5 = 3$, $\lambda_6 = 0.2$, and $\lambda_7 = 1$. The standard deviation values of the Gaussian noise were set as $\sigma = 3$ and $\sigma_x = \sigma_y = 3$.

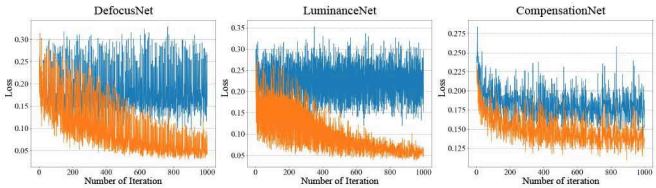


Fig. 6. Comparison of the proposed training method with the naïve training method. The orange lines are the loss values of the proposed method, and the blue lines, of the naïve method.

Note that all of the following experiments, except that in Sect. 5.2, were conducted using fixed parameters that had been trained in the aforementioned settings.

5.2 Validation of training strategy

As described in Sect. 3.4, we propose to separately train the three subnetworks, DefocusNet, LuminanceNet, and CompensationNet, rather than jointly updating all the weights of the entire network (i.e., as done in the naïve training method). In the naïve training method, it is not guaranteed that DefocusNet and LuminanceNet provide the correct defocus blur and luminance attenuation maps, respectively, at the early stage of the training. On the other hand, the correct maps are always injected into CompensationNet in the proposed training method. Therefore, we hypothesize that the proposed training method can train the subnetworks more efficiently than can the naïve training method. We experimentally tested this hypothesis by comparing the loss values between the naïve training method and the proposed training method.

Figure 6 shows the loss values at each iteration in the training of the three subnetworks using the naïve method and the proposed method. We see that the loss values did not decrease but only fluctuated over the iterations when the naïve training method was applied. On the other hand, the proposed training method decreased the loss values of all three subnetworks. Therefore, we confirmed that our hypothesis is correct; and thus, the proposed training strategy is valid.

5.3 Validation of network structure

The most important feature of the proposed network is its divided structure into an extractor and a generator. We designed the proposed structure to use the target image of the previous frame and its projected result only to estimate the defocus blur map and the luminance attenuation map. If we did not explicitly separate the network structure, the possibility that the texture of the previous frame would affect the compensation image in the current frame would increase. Therefore, we evaluated our proposed network structure by comparing it with a simpler one that computes the compensation image without the explicit estimations of the defocus blur map and the luminance attenuation map. Specifically, considering the versatile property of ResNet, we applied it as we did CompensationNet for the compared simple network, which takes the target image of the current frame, that of the previous frame, and its projected result as the inputs and generates a compensation image. We trained the simple network using the same dataset and pseudo-projection technique that we used to train the proposed network.

We compared the proposed and simple networks in the simulation using the pseudo-projection technique. Ten video files for the comparison were randomly selected from another dataset (Moment in Time

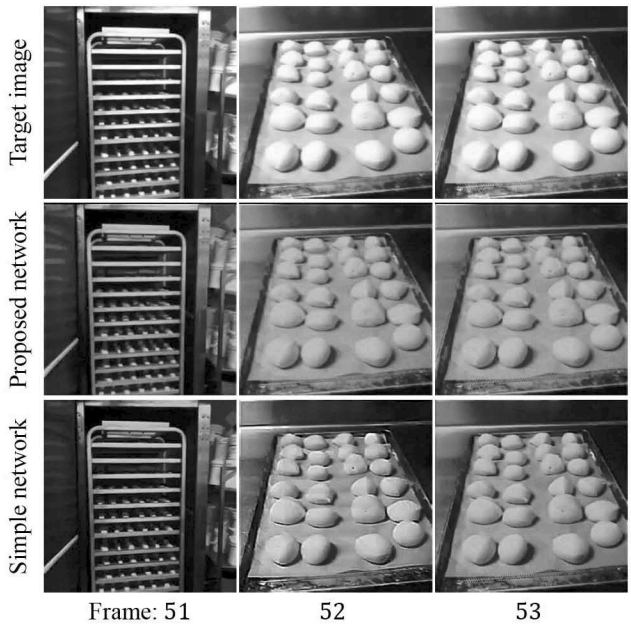


Fig. 7. The pseudo-projected results of the images compensated by the proposed and simple networks. There is a scene change between the 51st and 52nd frames.

Dataset [33]). All the selected video files were scaled and clipped into 256×256 pixels. Each video file had 90 frames. We paired each of the video files with a randomly selected depth image from the NYU Depth Dataset, which represents the shape of a projection surface. We tested each network by repeating the following process from the frame number of $t = 1$ to 90 for each video file. First, we fed the video frame of t as the target image $I_t(t)$, the projection image of the previous frame $I_p(t-1)$, and its pseudo-projected result $I_r(t-1) = \hat{\mathscr{P}}(I_p(t-1))$ into the tested network. In cases where the tested network was the proposed network, we also fed the defocus blur map $M'_d(t-1)$ and the luminance blur map $M'_l(t-1)$ estimated in the previous frame to the network. The tested network generated the compensation image $I_p(t)$. Second, we applied the proposed pseudo-projection technique using the paired depth image to the compensation image to synthesize its projected result $I_r(t) = \hat{\mathscr{P}}(I_p(t))$. We then computed the SSIM value of the pseudo-projected result compared to the normalized target image (see Eq. 8). Note that in the first frame, we used the video frame of $t = 1$ as $I_p(t-1)$ and $I_r(t-1)$, and we used a uniform black image as $M'_d(t-1)$ and $M'_l(t-1)$.

Figure 7 shows a part of the pseudo-projection results of the three video frames where a scene change occurred. In the result of the simple network, the silhouette of the shelf in the 51st frame remained in the 52nd frame. On the other hand, such artifacts are not visible in the result of the proposed network. This difference can be quantitatively confirmed in the SSIM values of the pseudo-projected results compared to the target images. We averaged the SSIM values of all the frames of all the video files. The pseudo-projected results of the proposed network (ave. SSIM: 0.820) were more similar to the target images than those of the simple network (ave. SSIM: 0.807) and those of a normal projection where the target images were directly pseudo-projected (ave. SSIM: 0.790). Therefore, we confirmed that the proposed network structure is more suitable for projector deblurring than the simple one.

5.4 Dynamic PM in actual setup

We designed our network to compensate for the defocus blur in an actual dynamic PM scenario. In particular, our pseudo-projection technique in the training applies warping to the virtually projected result, considering the potential misregistration in an actual projector-camera setup, as described in Sect. 4.4. We evaluated the compensation performance of the proposed network and the efficacy of the warping process using an actual PM system.

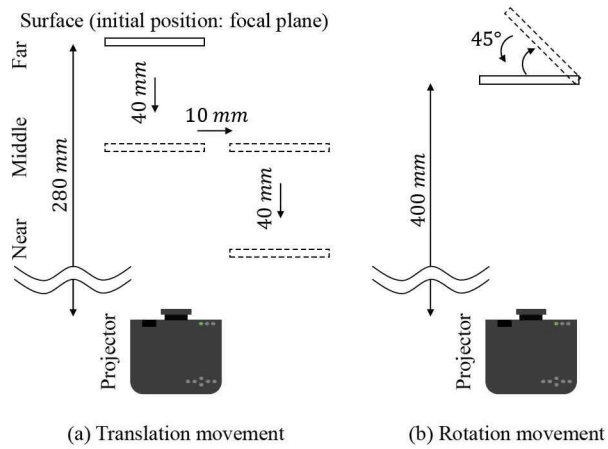


Fig. 8. The translation and rotation movements of the projection surface applied in the experiment.

5.4.1 Experimental setup

We built a physical projector-camera setup, as shown in Fig. 1. We used a DLP projector (BenQ TH682ST, 60mm lens aperture) and an industrial CMOS camera (FLIR FL3-U3-13S2C-CS). The projection surface was a flat, diffuse surface whose pose was controlled by a robot arm (UFACTORY xArm 7) so that the same sequence of the surface poses could be repeated in different conditions. Because the computation of a compensation image took 71.7 ms (Sect. 5.3), real-time frame-by-frame projector deblurring, which requires completion of the computation within 1/60 s, was difficult to perform in the current setup. Therefore, we merely emulated it using the robotic arm by slowly moving the surface. We performed a manual calibration to obtain the geometric relationships among the projector, the camera, and the surface, by which we were able to geometrically transform the captured image of the projected result such that the camera and the projector shared the same FoV.

We randomly selected three video clips from the Moment in Time Dataset used in Sect. 5.3 (90 frames, 256×256 pixels). These videos were projected onto the moving surface in two conditions: compensated and uncompensated. In the first condition, we computed the projection images by entering the video frames into the proposed network and geometrically transforming the compensated images to align them with the surface. In the second condition, the original video frames were directly geometrically transformed to compute the projection images.

We prepared two types of movements for the projection surface: translation and rotation (Fig. 8). For the translation movement, the surface was initially placed 280 mm away from the projector such that the surface was perpendicular to the projector's optical axis. Then, the surface was translated towards the projector along its optical axis (i.e., from far to middle) at the speed of 1 mm per frame for 40 frames. Then, the surface was translated in the horizontal direction at the same speed for 10 frames, during which no depth variation occurred. The surface was then translated towards the projector (i.e., from middle to near) again at the same speed for 40 frames. During the experiment, the projector's focusing distance was fixed at 280 mm from the projector lens (i.e., far). For the rotation movement, the surface was placed 400 mm away from the projector such that the surface was perpendicular to the projector's optical axis. The surface was rotated around the yaw axis at the angular velocity of 1 angle per frame for 45 frames. Then, it was rotated back at the same speed for 45 frames. During the experiment, the projector's focusing distance was fixed at 400 mm from the projector lens.

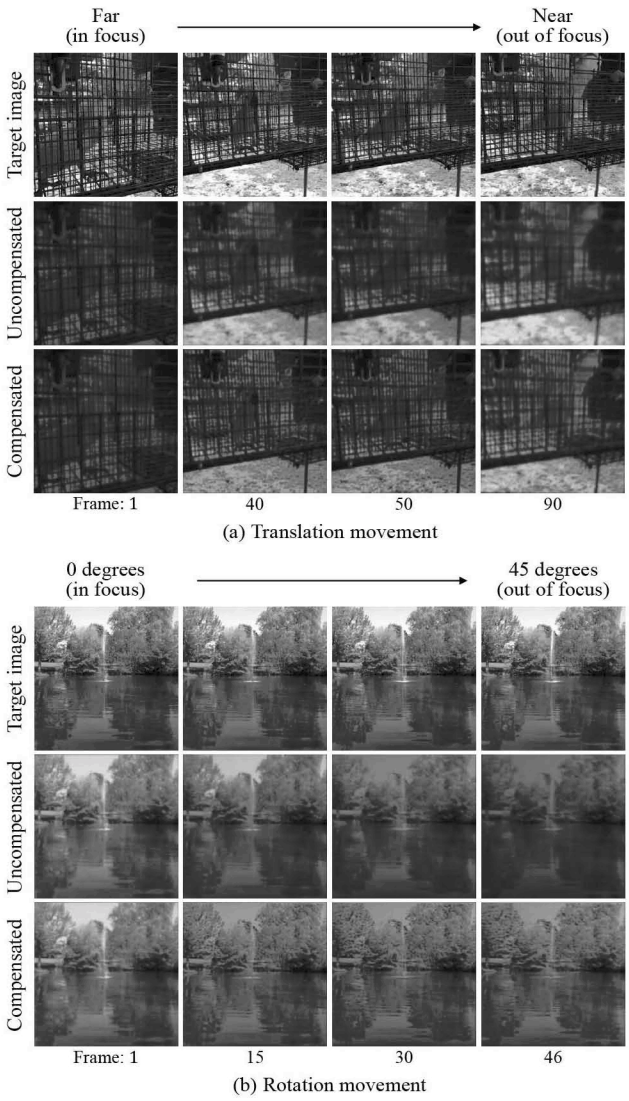


Fig. 9. The projected results of the uncompensated and compensated projection images onto the projection surface moved by the robotic arm according to the (a) translation and (b) rotation movements.

5.4.2 Results

Figure 9 shows a part of the projected results. The comparison of the results in the compensated and uncompensated shows that the high-frequency components (texture details) were preserved when the proposed compensation technique was applied, while those were missing in the uncompensated condition. Figure 10 shows the SSIM values of the projected results compared to the corresponding target video frames. The SSIM values in the compensated condition were higher than those in the uncompensated condition in both the translation and rotation movements. The average SSIM values over all the video frames of all the video files in the translation movement were 0.531 in the compensated condition and 0.438 in the uncompensated condition. Those in the rotation movement were 0.804 in the compensated condition and 0.722 in the uncompensated condition. Therefore, we can quantitatively confirm that the proposed network successfully compensated for the defocus blur in an actual PM system.

5.4.3 Validation of warping process in pseudo-projection

We saw the effect of the warping process on the estimated defocus blur and luminance attenuation maps. We trained our network without the warping process in the pseudo-projection technique and conducted the dynamic PM experiment using the same video files and projection

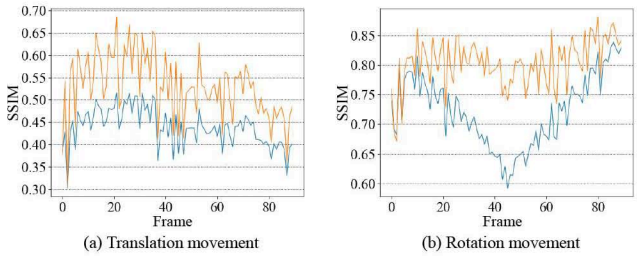


Fig. 10. The SSIM values of the projected results shown in Fig. 9 compared to the corresponding target images (orange: compensated condition; blue: uncompensated condition).

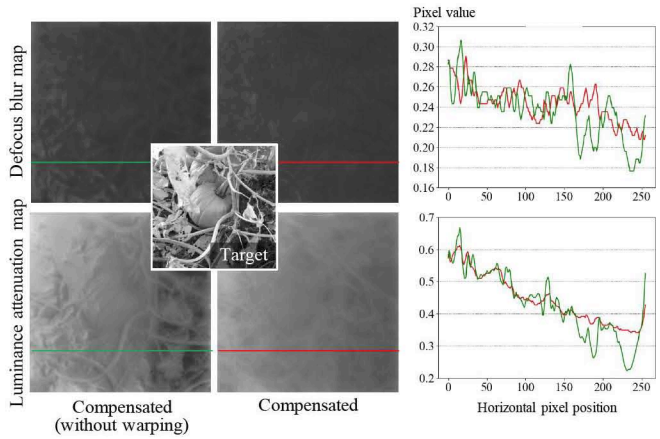


Fig. 11. The estimated defocus blur map and the luminance attenuation map from the network that was trained without the warping process in the pseudo-projection technique, and those from the proposed network. The right graphs show the pixel values along the green and red lines.

surface. We called this the "compensated (without warping)" condition. Figure 11 compares the estimated defocus blur and luminance attenuation maps in the rotation movement between the compensated condition and the compensated (without warping) condition. Because the surface was flat and rotated such that the projected image appeared focused at its right end and was getting defocused towards the left end, the intensities of the defocus blur map and the luminance attenuation map should be linearly decreasing from left to right. In the figure, we see that the target texture is prominent in both the maps of the compensated (without warping) condition. This artifact was caused by the inaccurate geometric registrations of the actual projector and camera, and thus, these two devices did not perfectly share the FoVs. The network trained without the warping process did not assume such situation, and thus, produced the artifacts. On the other hand, we can observe that the proposed network was less affected by the misregistration, and the texture of the target image is less visible in the maps. The right graphs quantitatively show this trend. The red lines (the compensated condition) decrease more smoothly from 0 to 255 on the horizontal pixel coordinate than the green lines (the compensated (without warping) condition).

5.4.4 Full-color dynamic PM

Full-color images can be compensated using the proposed network independently for each color channel. The full-color compensation image was generated by concatenating the compensation images of the three color channels. Figure 1 shows a part of the projected results captured by the camera. We can see that the high-frequency components were preserved in the results of the *comp* condition, while those of the corresponding video frames were missing in the results of the *uncomp* condition. Therefore, we experimentally confirmed that the proposed network could compensate full-color images for the defocus

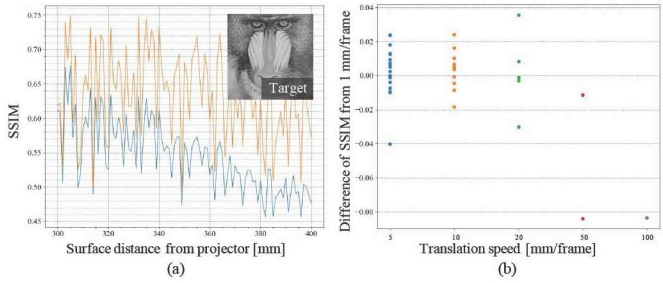


Fig. 12. Results of the test of the robustness of the proposed network against fast surface movement. (a) The SSIM values of the projected results of the original target image (blue) and those of the projected results of the compensated images when the translation speed was 1 mm/frame (orange). (b) The difference in the SSIM values between the translation speed of 1 mm/frame and the other speeds.

blur artifacts.

5.5 Robustness against fast movement

We developed the proposed technique by assuming that a projection surface does not move much within the period of a video frame (1/60 s). Thus, we experimentally evaluated how robust the proposed network is against a fast projection surface motion. We used the same PM system described in Sect. 5.4. The projection surface was translated along the projector's optical axis from 300 mm from the projector to 400 mm. We changed the translation speed to 1, 5, 10, 20, 50, and 100 mm/frame. The projector's focusing distance was fixed at 300 mm from the projector lens throughout this experiment. To compare the projected results of the same target images at different translation speeds, we used a static image (Mandrill) as the target image in this experiment.

Figure 12(a) plots the SSIM values of the projected results of the original target image and those of the compensated images in the translation speed of 1 mm/frame. Both SSIM values decreased according to the distance from the focusing distance (= 300 mm) due to the defocus blur artifact. The fluctuation in the SSIM values was caused by the misregistration of the projector and the camera to the surface. Figure 12(b) shows the difference in the SSIM values between the projected results of the translation speed of 1 mm/frame and those of the other translation speeds. The number of plots differed across the translation speeds because a slower speed decreases the surface positions to be projected (e.g., 20 positions in a 5mm/frame, 10 positions in a 10mm/frame, ..., 1 position in a 100mm/frame). We can observe that the similar compensation performances were achieved when the translation speed was 5, 10, and 20 mm/frame, while they were mostly lower when the speed was 50 and 100 mm/frame. These results indicate that the proposed technique robustly works at various surface speeds. Specifically, it worked when the translation speed of a projection surface was slower than 20 mm/frame (= 1,200 mm/s) in theory, which covers normal human hand motions. This result is valid only for the current PM setup; and thus, we consider more general cases. When the distance between a projection surface and a projector becomes shorter, the PSF changes more notably with the same depth difference of the surface. Because we used a relatively short distance in the current experimental setup, the PSF changed more quickly than in general dynamic PM scenarios. Therefore, our technique can support faster than 1,200 mm/s projection surface movement in a more normal dynamic PM setup, where the projection surface is placed farther than the current experimental setup.

6 Discussion

Although we achieved our research goal, our technique still has several technical limitations. First, the computational time required to generate the compensation image using the proposed network was 71.7 ms in the current setup. Therefore, we used the robotic arm to emulate real-time frame-by-frame compensation and projection by slowing down the surface movement. Among conventional techniques, simple (but inaccurate) projector deblurring methods work faster than a CNN-based method [28]. Previous studies showed that the acceptable latency for dynamic PM must be less than 10 ms [32,35,49]; otherwise, observers cannot perceive that a projected image is attached to a projection surface. However, this level of latency has not been achieved even by the simple methods. Therefore, reducing the computational time of projector deblurring to the required level while achieving satisfactory compensation performance is still an open issue in dynamic PM research. Second, the currently implemented network works for 256×256-pixel images. On the other hand, commercially available displays support a 4K resolution (i.e., 3,840×2,160 pixels). However, increasing the pixel size to 4K in the training of the proposed network is not feasible due to the limited size of the GPU memory.

A large part of a natural image consists only of low-spatial-frequency components; and thus, projector deblurring does not significantly change the image quality of the projected result of such a low-frequency part. Therefore, we could improve our method in terms of its computational cost and memory usage by applying the compensation network selectively to only the image areas that contain a large amount of high-spatial-frequency components. We may further speed up the inference process and reduce memory usage by incorporating factorization of two-dimensional PSF into two one-dimensional PSFs [43]. Another solution is the application of the latest multi-scale separable network, which has been proven to deblur a 4K video at the video rate of 35 fps [10]. We believe that projector deblurring for larger image size in dynamic PM is an important topic for future study.

Third, we developed our technique based on the simple assumption that the projection surface is uniformly white and diffuse. However, in reality, physical surfaces are more complex, such as those with spatially varying reflectance properties that include specular reflections. In addition, image quality degradation is caused not only by defocus blur and luminance attenuation but also by various global illumination effects that include interreflection, subsurface scattering, and diffraction. The proposed network does not support these degrading factors. A couple of previous techniques can compensate for all the artifacts [20, 50], but they require actual projection of a large number of calibration patterns. One of these studies demonstrated that a DNN could generate compensation images for complex artifacts. Therefore, for our future study, we find it interesting to combine our approach and the previous technique of optimizing projection images to jointly compensate for all the image degradation factors in dynamic PM.

7 Conclusion

This paper presented a DNN that can compensate for defocus blur in dynamic PM. The primary contribution of this paper is its development of a unique network structure that consists of an extractor and a generator. The extractor explicitly estimates a defocus blur map and a luminance attenuation map, which are then injected into the middle layers of the generator network that computes the compensation image. We also proposed a pseudo-projection technique for synthesizing physically plausible training data, considering not only the defocus blur and the luminance attenuation but also the geometric misregistration that potentially happens in actual PM use. We conducted a simulation and actual PM experiments and confirmed that: (1) the proposed network structure was more suitable for projector deblurring than a simple structure; (2) the network trained with the proposed pseudo-projection technique could compensate projection images for defocus blur and luminance attenuation artifacts in dynamic PM; and (3) the network supported the translation speed of a projection surface within a certain range that covers normal human motions. In our future study, we will test our method in more complex environments, such as one with clear depth discontinuities. We will also conduct a user study to evaluate how much the proposed network improves the projected image quality compared to the simple network in the perceptual space.

Acknowledgments

This work was supported by JSPS KAKENHI Grant Numbers JP20H05958 and JST, PRESTO Grant Number JPMJPR19J2, Japan.

REFERENCES

[1] D. G. Aliaga, Y. H. Yeung, A. Law, B. Sajadi, and A. Majumder. Fast high-resolution appearance editing using superimposed projections. *ACM Trans. Graph.*, 31(2), Apr. 2012.

[2] T. Amano. Projection center calibration for a co-located projector camera system. In *2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 449–454, 2014.

[3] A. Beck and M. Teboulle. Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. *IEEE transactions on image processing*, 18(11):2419–2434, 2009.

[4] A. Bermano, P. Brüschweiler, A. Grundhöfer, D. Iwai, B. Bickel, and M. Gross. Augmenting physical avatars using projector-based illumination. *ACM Trans. Graph.*, 32(6), Nov. 2013.

[5] A. H. Bermano, M. Billeter, D. Iwai, and A. Grundhöfer. Makeup lamps: Live augmentation of human faces via projection. *Computer Graphics Forum*, 36(2):311–323, 2017.

[6] O. Bimber and A. Emmerling. Multifocal projection: A multiprojector technique for increasing focal depth. *IEEE Transactions on Visualization and Computer Graphics*, 12(4):658–667, 2006.

[7] O. Bimber, D. Iwai, G. Wetzstein, and A. Grundhöfer. The visual computing of projector-camera systems. *ACM SIGGRAPH 2008 classes*, pp. 1–25, 2008.

[8] O. Bimber and R. Raskar. *Spatial Augmented Reality: Merging Real and Virtual Worlds*. A. K. Peters, Ltd., Natick, MA, USA, 2005.

[9] M. S. Brown, P. Song, and T.-J. Cham. Image pre-conditioning for out-of-focus projector blur. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 2, pp. 1956–1963. IEEE, 2006.

[10] S. Deng, W. Ren, Y. Yan, T. Wang, F. Song, and X. Cao. Multi-scale separable network for ultra-high-definition video deblurring. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 14030–14039, 2021.

[11] P. Favaro and S. Soatto. A geometric approach to shape from defocus. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(3):406–417, 2005.

[12] H. Fukui, T. Hirakawa, T. Yamashita, and H. Fujiyoshi. Attention branch network: Learning of attention mechanism for visual explanation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10705–10714, 2019.

[13] M. Grosse, G. Wetzstein, A. Grundhöfer, and O. Bimber. Coded aperture projection. *ACM Transactions on Graphics (TOG)*, 29(3):1–12, 2010.

[14] A. Grundhöfer and D. Iwai. Robust, error-tolerant photometric projector compensation. *IEEE Transactions on Image Processing*, 24(12):5086–5099, 2015.

[15] A. Grundhöfer and D. Iwai. Recent advances in projection mapping algorithms, hardware and applications. *Computer Graphics Forum*, 37(2):653–675, 2018.

[16] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pp. 1026–1034, 2015.

[17] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

[18] B. Huang and H. Ling. Compennet++: End-to-end full projector compensation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7165–7174, 2019.

[19] B. Huang and H. Ling. End-to-end projector photometric compensation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[20] B. Huang and H. Ling. Deprocams: Simultaneous relighting, compensation and shape reconstruction for projector-camera systems. *IEEE Transactions on Visualization and Computer Graphics*, 27(5):2725–2735, 2021.

[21] B. Huang, T. Sun, and H. Ling. End-to-end full projector compensation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[22] Y. Itoh, T. Kaminokado, and K. Akşit. Beaming displays. *IEEE Transactions on Visualization and Computer Graphics*, 27(5):2659–2668, 2021.

[23] D. Iwai, R. Matsukage, S. Aoyama, T. Kikukawa, and K. Sato. Geometrically consistent projection-based tabletop sharing for remote collaboration. *IEEE Access*, 6:6293–6302, 2018.

[24] D. Iwai, S. Mihara, and K. Sato. Extended depth-of-field projector by fast focal sweep projection. *IEEE transactions on visualization and computer graphics*, 21(4):462–470, 2015.

[25] D. Iwai and K. Sato. Document search support by making physical documents transparent in projection-based mixed reality. *Virtual Reality*, 15(2):147–160, Jun 2011.

[26] B. R. Jones, H. Benko, E. Ofek, and A. D. Wilson. Illumiroom: peripheral projected illusions for interactive experiences. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 869–878, 2013.

[27] N. Joshi, R. Szeliski, and D. J. Kriegman. Psf estimation using sharp edge prediction. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8. IEEE, 2008.

[28] Y. Kageyama, M. Isogawa, D. Iwai, and K. Sato. Prodebnet: projector deblurring using a convolutional neural network. *Optics Express*, 28(14):20391–20403, 2020.

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

[30] K. Matsushita, D. Iwai, and K. Sato. Interactive bookshelf surface for in situ book searching and storing support. In *Proceedings of the 2nd Augmented Human International Conference*, 2011.

[31] M. R. Mine, J. van Baar, A. Grundhofer, D. Rose, and B. Yang. Projection-based augmented reality in disney theme parks. *Computer*, 45(7):32–40, 2012.

[32] L. Miyashita, Y. Watanabe, and M. Ishikawa. Midas projection: Markerless and modelless dynamic projection mapping for material representation. *ACM Trans. Graph.*, 37(6), Dec. 2018.

[33] M. Monfort, A. Andonian, B. Zhou, K. Ramakrishnan, S. A. Bargal, T. Yan, L. Brown, Q. Fan, D. Gutfreund, C. Vondrick, et al. Moments in time dataset: one million videos for event understanding. *IEEE transactions on pattern analysis and machine intelligence*, 42(2):502–508, 2019.

[34] M. Nagase, D. Iwai, and K. Sato. Dynamic defocus and occlusion compensation of projected imagery by model-based optimal projector selection in multi-projection environment. *Virtual Reality*, 15(2-3):119–132, 2011.

[35] G. Narita, Y. Watanabe, and M. Ishikawa. Dynamic projection mapping onto deforming non-rigid surface using deformable dot cluster marker. *IEEE Transactions on Visualization and Computer Graphics*, 23(3):1235–1248, 2017.

[36] P. K. Nathan Silberman, Derek Hoiem and R. Fergus. Indoor segmentation and support inference from rgbd images. In *Proceedings of European Conference on Computer Vision*, pp. 746–760. Springer, 2012.

[37] H. Nishino, E. Hatano, S. Seo, T. Nitta, T. Saito, M. Nakamura, K. Hattori, M. Takatani, H. Fuji, K. Taura, and S. Uemoto. Real-time navigation for liver surgery using projection mapping with indocyanine green fluorescence: Development of the novel medical imaging projection system. *Annals of Surgery*, 267(6):1134–1140, 2018.

[38] Y. Oyamada and H. Saito. Focal pre-correction of projected image for deblurring screen image. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8. IEEE, 2007.

[39] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation, 2015.

[40] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: nonlinear phenomena*, 60(1-4):259–268, 1992.

[41] C. Siegl, M. Colaianni, L. Thies, J. Thies, M. Zollhöfer, S. Izadi, M. Stamminger, and F. Bauer. Real-time pixel luminance optimization for dynamic multi-projection mapping. *ACM Trans. Graph.*, 34(6), Oct. 2015.

[42] T. Sueishi, H. Oku, and M. Ishikawa. Lumipen 2: Dynamic projection mapping with mirror-based robust high-speed tracking against illumination changes. *Presence: Teleoperators and Virtual Environments*, 25(4):299–321, 2016.

[43] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826, 2016.

[44] T. Takezawa, D. Iwai, K. Sato, T. Hara, Y. Takeda, and K. Murase. Material surface reproduction and perceptual deformation with projection mapping for car interior design. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 251–258, 2019.

[45] D. Tone, D. Iwai, S. Hiura, and K. Sato. Fibar: Embedding optical fibers in 3d printed objects for active markers in dynamic projection mapping. *IEEE Transactions on Visualization and Computer Graphics*, 26(5):2030–2040, 2020. doi: 10.1109/TVCG.2020.2973444

[46] D. Wang, I. Sato, T. Okabe, and Y. Sato. Radiometric compensation in

a projector-camera system based properties of human vision system. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Workshops*, 2005.

[47] L. Wang, H. Xu, S. Tabata, Y. Hu, Y. Watanabe, and M. Ishikawa. High-speed focal tracking projection based on liquid lens. In *ACM SIGGRAPH 2020 Emerging Technologies*. Association for Computing Machinery, New York, NY, USA, 2020.

[48] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Trans. Image Process.*, 13(4)(4):600–612, 2004.

[49] Y. Watanabe, G. Narita, S. Tatsuno, T. Yuasa, K. Sumino, and M. Ishikawa. High-speed 8-bit image projector at 1,000 fps with 3 ms delay. In *The International Display Workshops*, pp. 1064–1065, 2015.

[50] G. Wetzstein and O. Bimber. Radiometric compensation through inverse light transport. In *15th Pacific Conference on Computer Graphics and Applications (PG'07)*, pp. 391–399, 2007.

[51] T. Xue, B. Chen, J. Wu, D. Wei, and W. T. Freeman. Video enhancement with task-oriented flow. *International Journal of Computer Vision (IJCV)*, 127(8):1106–1125, 2019.

[52] L. Zhang and S. Nayar. Projection defocus analysis for scene capture and image display. In *ACM SIGGRAPH 2006 Papers*, pp. 907–915. 2006.