Projection Misalignment Compensation in Appearance Manipulation for Rotational Motion

Ryusuke Hayashi[1,*], Toshiyuki Amano[1]

[1]Wakayama University, 930, Sakaedani, Wakayama-shi, Wakayama, Japan

Abstract

In appearance manipulation using a projector-camera system, projection images are generated based on the observed appearance of the projection target and its response to projected light, so that the appearance of the projection target approaches a desired appearance. However, when the projection target moves, processing latency between image capture and projection causes projection misalignment. Previous studies have investigated compensation methods for misalignment caused by linear motion and translational motion in two axial directions; however, when the projection target itself undergoes rotational motion, each point on the projection target follows an arc-shaped trajectory around the rotation center, making it difficult to directly apply conventional methods based on translational motion. Therefore, this study proposes a projection misalignment compensation method using a pixel map corresponding to rotational motion, assuming that the projection target rotates around a fixed rotation center. To support arbitrary rotational velocities, three methods are compared: a linear approximation method, a multi-stage application method, and a combination method. Experimental results show that the linear approximation method using a high reference angular velocity achieves the smallest misalignment in the tangential direction, whereas the combination method using multiple reference pixel maps achieves the smallest misalignment in the radial direction. In particular, the combination method suppresses radial misalignment while maintaining relatively small tangential misalignment, confirming its effectiveness for compensating projection misalignment under rotational motion.

Keywords

appearance control, projector-camera system, projection misalignment compensation, rotational motion

1. Introduction

In recent years, spatial augmented reality (SAR) technology, which directly projects virtual information onto real-world objects and spaces using projectors, has been used in various fields, including projection mapping [1]. In SAR, real objects can be used as media for information presentation. Furthermore, by controlling the projected light, it is possible to manipulate the appearance of real objects, such as their color and material impression.

In appearance manipulation using a projector-camera system, projection images are generated based on the appearance of the projection target observed by the camera and its response to projection, thereby bringing the appearance of the projection target closer to a desired state. However, when such appearance manipulation is applied to a moving projection target, processing latency between image capture and projection causes projection misalignment. If the position of the projection target at the time of image capture differs from that at the time of actual projection, the projected light is emitted onto a position different from the intended one, which is observed as afterimages or blur. This problem occurs even when the projector-camera system is fixed, as long as the projection target moves.

To address this problem, a projection misalignment compensation method for a moving projector-camera system has been proposed [2]. In the conventional method, the correspondence between camera images and projection images before and after movement is acquired in advance as a pixel map. By considering the amount of motion that occurs during the processing latency, the projection position at a future time is estimated. Compensation for translational motion in two axial directions has also been investigated [3].

APMAR'26: The 18th Asia-Pacific Workshop on Mixed and Augmented Reality, Aug. 3-5, 2026, Taipei, Taiwan
*Corresponding author.
s276203@wakayama-u.ac.jp (R. Hayashi); amano@wakayama-u.ac.jp (T. Amano)
0009-0001-2977-4101 (R. Hayashi); 0000-0003-4146-1375 (T. Amano)

However, when the projection target moves along a curved trajectory or moves with rotational motion, pixel displacement in the image cannot be represented simply as translation in the X-axis or Y-axis direction. This is because, in translational motion, linearity can be assumed between the amount of motion and the displacement represented by the pixel map. In contrast, a pixel map representing rotational motion is expressed as displacement vectors in the tangential direction around the rotation center. Therefore, unlike translational motion, linearity cannot be assumed for an arbitrary amount of rotation.

Therefore, this study investigates projection compensation for motion including rotation. Specifically, assuming that the position of the rotation center is known, we create a pixel map corresponding to rotational motion from the correspondence between camera images and projection images before and after rotational motion. Using this pixel map, we propose a method for compensating projection misalignment under arbitrary rotational velocities. In addition, we compare three methods: a linear approximation method, a multi-stage application method, and a combination method, and evaluate projection misalignment in the tangential and radial directions.

2. Related Work

2.1. Dynamic Projection Mapping and Appearance Manipulation

Dynamic projection mapping (DPM) is a SAR technology for projecting images onto moving objects. In DPM, the shape and pose of a projection target are measured or estimated, and the projection image is deformed according to the geometry of the projection target surface, thereby presenting images or textures at geometrically aligned positions [4]. DPM is effective when the shape and motion of the projection target can be acquired with high accuracy. However, depending on the projection target and environmental conditions, it may be difficult to stably acquire the shape and pose of the projection target.

In contrast, appearance manipulation using a projector-camera system does not aim to attach a predefined texture to the projection target, but rather to bring the observed appearance closer to a desired appearance. In this approach, projection images are generated based on camera images and the response of the projection target to projected light. Therefore, the projection can be adjusted based on the observed appearance without preparing detailed textures of the projection target in advance. However, because this approach involves a series of processes, including image capture, image processing, projection image generation, and projection, projection misalignment caused by processing latency becomes a problem when the projection target or the projector-camera system moves.

2.2. Projection Misalignment Compensation Using Pixel Maps

As described in the previous section, when appearance manipulation using a projector-camera system is applied to a moving projection target, projection misalignment occurs due to the processing latency between image capture and projection. Here, processing latency refers to the total delay from image capture to actual projection onto the projection target. It includes the image capture time, image processing time, projection image transfer time, and display delay of the projector. This misalignment is observed as duplicated edges, afterimages, and blur, and prevents the desired appearance from being presented. Therefore, in order to apply appearance manipulation to moving projection targets, it is necessary to consider the motion that occurs during the processing latency and estimate the corresponding position at the time of projection.

To address this problem, Amano et al. proposed a projection misalignment compensation method using pixel maps for a moving projector-camera system [2]. A pixel map is a correspondence that stores, for each pixel in a camera image, which pixel in a projector image it corresponds to.

In the method proposed by Amano et al., as shown in Fig. 1, the correspondences among camera images, projector images, and external projector images before and after movement are acquired as

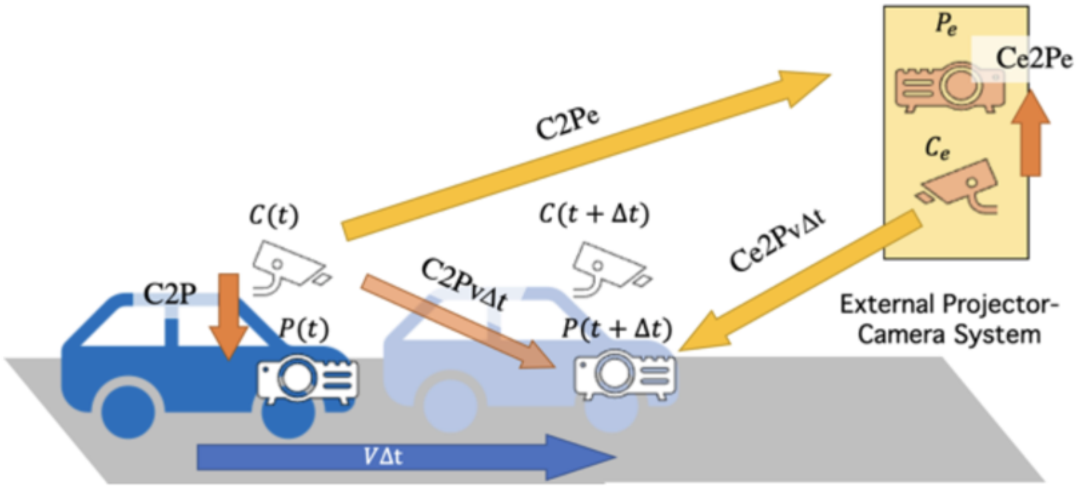


Figure 1: Pre-calibration for projection alignment correction [2]

multiple pixel maps. By composing these pixel maps, a compensation pixel map from the camera image at the time of image capture to the projector image at the time of projection is obtained.

Let t denote the time before movement, Δt denote the processing latency, and $V\Delta t$ denote the amount of motion that occurs during the processing latency. Let $C(t)$ denote the camera image at time t, $C(t+\Delta t)$ denote the camera image at time $t+\Delta t$, and $P(t+\Delta t)$ denote the projector image at the time of projection. Then, the compensation pixel map $C2P_{V\Delta t}$, which maps the camera image $C(t)$ before movement to the projector images $P(t+\Delta t)$ after the processing latency, is obtained as

$$C2P_{V\Delta t} = C_{V\Delta t}2P_{V\Delta t} \circ (C_{V\Delta t}2P_e)^{-1} \circ C2P_e \quad (1)$$

where $\circ$ denotes the composition of pixel maps, and $(C_{V\Delta t}2P_e)^{-1}$ denotes the inverse function of $C_{V\Delta t}2P_e$.

Using this compensation pixel map, a pixel observed in the camera image at the time of image capture can be transformed into the corresponding projector pixel at the time of projection after the processing latency. Let $C2P$ denote the ordinary correspondence between the camera image and the projector images at the reference position. Let $C2P_{V\Delta t}$ denote the compensation pixel map obtained at the reference velocity V, and let V' denote the actual moving velocity. Then, the compensation pixel map $C2P_{V'\Delta t}$ for an arbitrary velocity is obtained by linearly scaling the displacement at the reference velocity according to the velocity ratio as

$$C2P_{V'\Delta t} = \frac{V'}{V}(C2P_{V\Delta t} - C2P) + C2P \quad (2)$$

where $C2P_{V\Delta t} - C2P$ represents the displacement of projector pixels that occurs during the processing latency at the reference velocity V.

However, because this method uses a linear approximation based on a single reference velocity, the compensation accuracy may decrease when it is applied to velocities far from the reference velocity, or when the amount of misalignment changes nonlinearly depending on the position in the image. To address this problem, Kano et al. proposed methods for improving compensation accuracy for arbitrary velocities, including a piecewise linear approximation method using pixel maps obtained at multiple reference velocities and a method that repeatedly applies a pixel map generated for a small reference displacement [3]. The following describes the misalignment compensation method based on small displacements, which serves as the basis for the compensation method for rotational motion discussed later.

In the misalignment compensation method based on small displacements, a pixel map between camera images before and after movement is acquired for a small time interval $\Delta t'$, which is shorter than the processing latency Δt. By applying this pixel map multiple times, the method can handle an arbitrary amount of motion. Let $\mathbf{x}_0 = (x_0, y_0)^T$ denote the pixel coordinate in the camera image before movement, and let $f_{\mathbf{V}\Delta t'}$ denote the pixel map between camera images corresponding to the small displacement. Then, by repeatedly applying this pixel map, the corresponding position in the camera image after movement is obtained as

$$\mathbf{x}_{k+1} = f_{\mathbf{V}\Delta t'}(\mathbf{x}_k) \quad (k = 0, 1, \dots, n-1)$$

where n is the number of applications required to represent the actual amount of motion as an accumulation of small displacements. In this method, nonlinear misalignment in the image can be approximated step by step by reducing the displacement in each application. On the other hand, because the pixel map is applied multiple times, pixel rounding errors at each application may accumulate.

Furthermore, Kano et al. extended this concept to two-axis translational motion. Specifically, they generated pixel maps corresponding to the reference displacements along the x- and y-axes, respectively. They then obtained the compensation pixel map for arbitrary in-plane translational motion as a linear combination of these maps, weighted by the displacement amount along each axis. This enables projection misalignment compensation not only for motion in a single direction, but also for translational motion in an arbitrary direction on a plane.

As described above, Amano et al. presented a basic framework for compensating projection misalignment caused by motion during processing latency using pixel maps. Kano et al. improved the compensation accuracy for arbitrary velocities and extended the method to two-axis translational motion. However, these methods mainly target translational motion. In translational motion, each point on the projection target moves approximately in the same direction, making it relatively easy to generate a compensation pixel map by combining the displacement components along each axis.

In contrast, when the projection target itself undergoes rotational motion, each point follows an arc-shaped trajectory around the rotation center. Therefore, the moving direction and amount of motion of each pixel depend on its position relative to the rotation center. As a result, compensation based only on two-axis translational motion may not be sufficient to represent arc-shaped pixel motion or misalignment that varies according to the distance from the rotation center.

Therefore, this study targets projection misalignment compensation for a projection target rotating around a fixed rotation center, while building on existing pixel-map-based compensation methods. The proposed approach addresses situations where, unlike translational motion, the moving direction and amount of motion vary depending on the position relative to the rotation center.

3. Proposed Method

3.1. Challenges in Extending the Method to Rotational Motion

In this study, we extend the projection misalignment compensation method based on pixel maps proposed by Kano et al. from translational motion to motion including rotational motion. In this method, a compensation pixel map for an arbitrary translational motion is generated by linearly scaling the displacement of a pixel map acquired for a reference amount of motion according to the actual amount of motion.

However, in rotational motion, each point on the projection target moves along a circular trajectory around the rotation center. Therefore, the moving direction and amount of motion of each pixel vary depending on its position relative to the rotation center. In addition, when the amount of rotation becomes large, pixel motion in the image appears not as a linear displacement but as an arc-shaped motion. Therefore, simply scaling the displacement of a pixel map acquired for a reference amount of rotation, as in translational motion, may not sufficiently represent pixel motion corresponding to the actual rotational motion.

In this study, we acquire a pixel map corresponding to rotational motion from the correspondence before and after rotational motion. We also investigate methods for generating a compensation pixel map corresponding to an arbitrary rotational velocity from pixel maps acquired at reference rotational velocities. In the following, Section 3.2 describes the acquisition of a pixel map corresponding to rotational motion, and Section 3.3 describes three compensation strategies for arbitrary rotational velocities: a linear approximation method, a multi-stage application method, and a combination method.

3.2. Acquisition of a Pixel Map Corresponding to Rotational Motion

In this study, we use a pixel map that represents the coordinate correspondence between the captured image at time t and the captured image after the projection target rotates at an angular velocity V_θ during the processing latency Δt.

The pixel map corresponding to rotational motion is obtained by acquiring the correspondence between the camera image and the projector image at the reference position and the correspondence between the camera image and the projector image at the rotated position, and then composing these correspondences. The procedure for acquiring the pixel map is as follows.

1. Acquire the correspondence $C2P$ between camera pixels and projector pixels at the reference position.
2. Rotate the projection target around the rotation center by the angle $V_\theta \Delta t$, which corresponds to the rotation at angular velocity V_θ during the processing latency Δt.
3. Acquire the correspondence $C_{V_\theta \Delta t}2P$ between camera pixels and projector pixels at the rotated position.

Here, let $C2P$ denote the correspondence between the camera image and the projector image at the reference position, and let $C_{V_\theta \Delta t}2P$ denote the correspondence between the camera image and the projector image after rotation by the angle $V_\theta \Delta t$. Then, the pixel map $C2C_{V_\theta \Delta t}$, which represents the correspondence between camera images before and after rotational motion, is obtained as

$$C2C_{V_\theta \Delta t} = C2P \circ \left(C_{V_\theta \Delta t}2P\right)^{-1} \tag{3}$$

where $\circ$ denotes the composition of pixel maps, and $\left(C_{V_\theta \Delta t}2P\right)^{-1}$ denotes the inverse transformation of the correspondence between the camera image and the projector image at the rotated position. Using this pixel map, we can estimate which position in the camera image after rotation corresponds to each pixel in the camera image at the reference position.

3.3. Compensation Methods for Arbitrary Rotational Velocities

The pixel map $C2C_{V_\theta \Delta t}$ obtained in the previous section is acquired for the reference angular velocity V_θ. However, in actual appearance manipulation, the rotational velocity of the projection target does not always coincide with the reference velocity. Therefore, it is necessary to generate a compensation pixel map corresponding to an arbitrary rotational velocity $V'\theta$ using pixel maps acquired at reference rotational velocities.

In this study, we investigate the following three methods for generating a compensation pixel map corresponding to an arbitrary rotational velocity. First, we use a method that scales the displacement of a reference pixel map according to the velocity ratio. This method is referred to as the linear approximation method. Second, we use a method that repeatedly applies a pixel map corresponding to a small rotational displacement. This method is referred to as the multi-stage application method. Third, we use a method that combines multiple pixel maps acquired at different reference velocities. This method is referred to as the combination method. These methods correspond to arbitrary rotational velocities based on different strategies: linear approximation using a single reference velocity, accumulation of small displacements, and combination of multiple reference velocities.

In the following, let $\mathbf{x} = (x, y)^T$ denote a pixel coordinate in the camera image at the reference position, and let $f_{V_\theta}(\mathbf{x})$ denote the pixel map corresponding to the reference rotational velocity V_θ.

3.3.1. Linear Approximation Method

In the linear approximation method, the displacement of the pixel map acquired at the reference rotational velocity V_θ is linearly scaled according to the actual rotational velocity V'_θ. When a pixel $\mathbf{x}$ is mapped to $f_{V_\theta}(\mathbf{x})$ by the reference pixel map, the displacement at the reference velocity, $\Delta\mathbf{x}_{V\theta}$, is expressed as

$$\Delta\mathbf{x}_{V\theta}(\mathbf{x}) = f_{V_\theta}(\mathbf{x}) - \mathbf{x} \tag{4}$$

By linearly scaling this displacement by the velocity ratio V'_θ/V_θ, the pixel map $f^{\mathrm{lin}}_{V'\theta}$ corresponding to the arbitrary rotational velocity V'_θ is obtained as

$$f^{\mathrm{lin}}_{V'_\theta}(\mathbf{x}) = \mathbf{x} + \frac{V'_\theta}{V_\theta}\left(f_{V_\theta}(\mathbf{x}) - \mathbf{x}\right) \tag{5}$$

This method is easy to implement because it can handle an arbitrary rotational velocity using a single reference pixel map. On the other hand, pixel motion in rotational motion is inherently arc-shaped. Therefore, when the actual velocity differs greatly from the reference velocity, errors may occur because the arc-shaped motion is approximated by a linear displacement.

3.3.2. Multi-stage Application Method

In the multi-stage application method, a pixel map corresponding to a small amount of rotational motion is used as a reference, and this pixel map is applied multiple times to generate a pixel map corresponding to an arbitrary rotational velocity V'_θ. Let V_θ denote the reference rotational velocity. For the arbitrary rotational velocity V'_θ, we decompose it as

$$V'_\theta = nV_\theta + r \tag{6}$$

where n denotes the number of applications of the reference pixel map, and r denotes the remaining rotational velocity component. The number of applications n is given by

$$n = \left\lfloor \frac{V'_\theta}{V_\theta} \right\rfloor \tag{7}$$

and the residual component r is expressed as

$$r = V'_\theta - nV_\theta \tag{8}$$

The pixel coordinate obtained by applying the reference pixel map f_{V_θ} n times is defined as follows:

$$\mathbf{x}_0 = \mathbf{x} \tag{9}$$

$$\mathbf{x}_{i+1} = f_{V_\theta}(\mathbf{x}_i) \quad (i = 0, 1, \dots, n-1) \tag{10}$$

For the residual component r, the displacement of the reference pixel map is linearly approximated. Therefore, the pixel map $fV'^{\mathrm{small}}_\theta$ for the arbitrary rotational velocity $V'\theta$ obtained by the multi-stage application method is given by

$$f^{\mathrm{small}}_{V'_\theta}(\mathbf{x}) = \mathbf{x}_n + \frac{r}{V_\theta}\left(f_{V_\theta}(\mathbf{x}_n) - \mathbf{x}_n\right) \tag{11}$$

This method can represent pixel motion close to rotational motion by repeatedly applying a small rotational displacement. However, because the pixel map stores correspondences on a pixel-by-pixel basis, rounding errors occur at each application. Therefore, when the number of applications n becomes large, these errors may accumulate, resulting in projection misalignment or blur in the projected image.

3.3.3. Combination Method

In the combination method, multiple pixel maps acquired at different reference rotational velocities are used to generate a pixel map corresponding to an arbitrary rotational velocity. In this study, two pixel maps, $f_{V_{\theta 1}}$ and $f_{V_{\theta 2}}$, are acquired for two reference rotational velocities, $V_{\theta 1}$ and $V_{\theta 2}$, where $V_{\theta 1} > V_{\theta 2}$.

The arbitrary rotational velocity V'_θ is represented as a combination of the larger reference velocity $V_{\theta 1}$ and the smaller reference velocity $V_{\theta 2}$:

$$V'_\theta = nV_{\theta 1} + mV_{\theta 2} + r \tag{12}$$

where n denotes the number of applications of $f_{V_{\theta 1}}$, m denotes the number of applications of $f_{V_{\theta 2}}$, and r denotes the remaining rotational velocity component. First, the number of applications n of the larger reference velocity is determined as

$$n = \left\lfloor \frac{V'_\theta}{V_{\theta 1}} \right\rfloor \tag{13}$$

and the remaining velocity component V_{rem} is given by

$$V_{\text{rem}} = V'_\theta - nV_{\theta 1} \tag{14}$$

Next, the number of applications m of the smaller reference velocity is determined as

$$m = \left\lfloor \frac{V_{\text{rem}}}{V_{\theta 2}} \right\rfloor \tag{15}$$

and the residual component r is given by

$$r = V_{\text{rem}} - mV_{\theta 2} \tag{16}$$

The pixel coordinates are updated by first applying $f_{V_{\theta 1}}$ n times and then applying $f_{V_{\theta 2}}$ m times. Let the initial coordinate be $\mathbf{x}_0 = \mathbf{x}$. Then,

$$\mathbf{x}_{i+1} = f_{V_{\theta 1}}(\mathbf{x}_i) \quad (i = 0, 1, \dots, n-1) \tag{17}$$

$$\mathbf{x}_{n+j+1} = f_{V_{\theta 2}}(\mathbf{x}_{n+j}) \quad (j = 0, 1, \dots, m-1) \tag{18}$$

Finally, for the residual component r, the pixel map of the smaller reference velocity $V_{\theta 2}$ is linearly approximated. Therefore, the pixel map $f^{\text{comb}}_{V'\theta}$ for the arbitrary rotational velocity $V'\theta$ obtained by the combination method is given by

$$f^{\text{comb}}_{V'_\theta}(\mathbf{x}) = \mathbf{x}_{n+m} + \frac{r}{V_{\theta 2}} \left(f_{V_{\theta 2}}(\mathbf{x}_{n+m}) - \mathbf{x}_{n+m} \right) \tag{19}$$

This method can reduce the number of pixel-map applications compared with the multi-stage application method. In addition, compared with the single linear approximation method, it can reduce the difference between the actual rotational velocity and the reference velocity used for approximation. Therefore, this method is expected to suppress both the accumulation of rounding errors and the error caused by linear approximation.

4. Experiment

4.1. Experimental Setup

In this experiment, we first visually confirm the compensation effect using a character pattern, and then quantitatively evaluate projection misalignment using a circular checker pattern. Furthermore, based on the evaluation results, we show application examples of several types of appearance manipulation using the selected method. In the visual evaluation, we compare the results without compensation and those with each compensation method, and observe changes in afterimages, blur, and local misalignment. In the quantitative evaluation, the projection result in the stationary state is used as a reference, and the projection misalignment during rotation is evaluated by separating it into tangential and radial components.

Fig. 2 shows the experimental setup used in this study. The left image presents an overview of the experimental system, which consists of a projector, a camera, and a rotary stage. The right side shows schematic diagrams of the geometric configurations of the projector and camera. In this experiment, the projector and camera were fixed, and only the projection target was rotated on the rotary stage. A coordinate system was defined on the rotary stage, with its origin set at the center of rotation. The X- and Y-axes were defined on the stage plane, and the projection target was assumed to rotate within this plane.

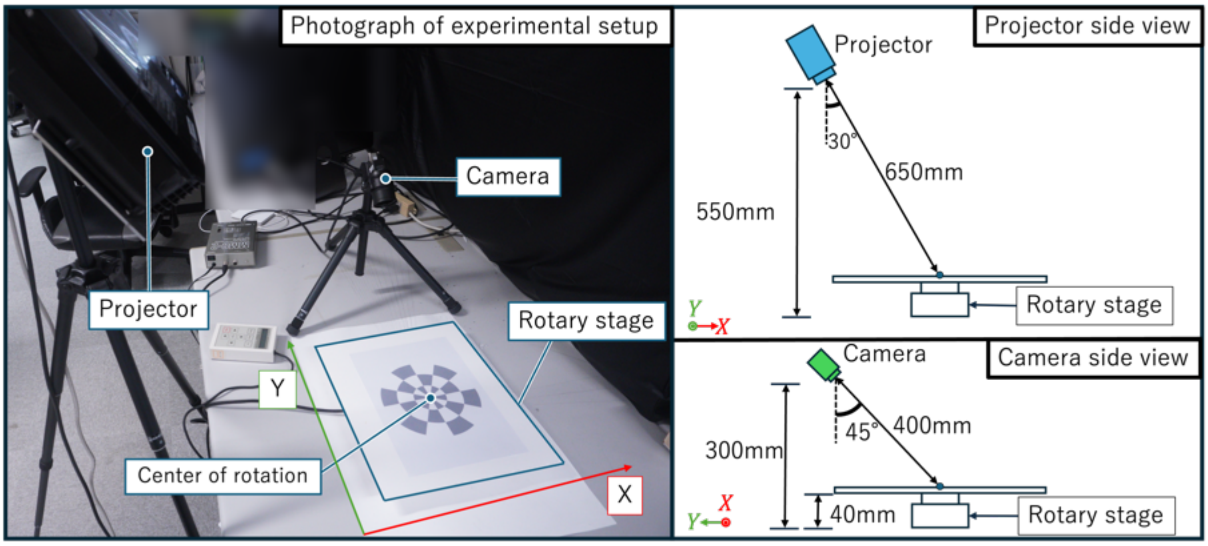


Figure 2: Experimental setup

A rotary stage (Chuo Precision Industrial 9L-185) was used to rotate the projection target. The projector (EPSON EB-W420, resolution: 1920 × 1080) was placed so as to face the center of rotation. The camera (Ximea MQ013CG-E, resolution: 1280 × 900) was fixed at a position where it could capture the projection target. As shown on the right side of Fig. 2, the positions, heights, and angles of the projector and camera were fixed throughout the experiment, while only the projection target rotated around the center of rotation. This reproduced a situation in which the relative positional relationship between the projector-camera system and the projection target changes during the processing latency.

To perform appearance manipulation, we acquired the correspondence between the camera image and the projector image at the reference position, as well as the correspondence after rotating the projection target by a known angle. Using these correspondences, we generated pixel maps corresponding to rotational motion. The generated pixel maps were then used to compensate for misalignment of the appearance manipulation results on the rotating projection target.

The compared methods were the linear approximation method, the multi-stage application method, and the combination method. In the linear approximation method, two reference rotational velocities, 7.25 deg/s and 72.50 deg/s, were used. In the multi-stage application method, 7.25 deg/s was used as the reference rotational velocity. In the combination method, 10.88 deg/s and 32.63 deg/s were used as the reference rotational velocities.

4.2. Compensation Results for Rotational Motion

In this section, we visually compare the compensation effects of the proposed method using a character pattern. Since the character pattern has clear contours, afterimages, blur, and local misalignment caused by projection misalignment can be easily observed.

Fig. 3 shows the projection results when the projection target is stationary, when it is rotated without compensation, and when each compensation method is applied. The rotation speed of the projection target in this experiment was 40.00 deg/s. Note that the values shown in the figure, 7.25 deg/s, 10.88 deg/s, 32.63 deg/s, and 72.50 deg/s, do not indicate the actual rotation speed of the projection target. Instead, they indicate the reference rotation speeds used when generating the pixel maps for each compensation method.

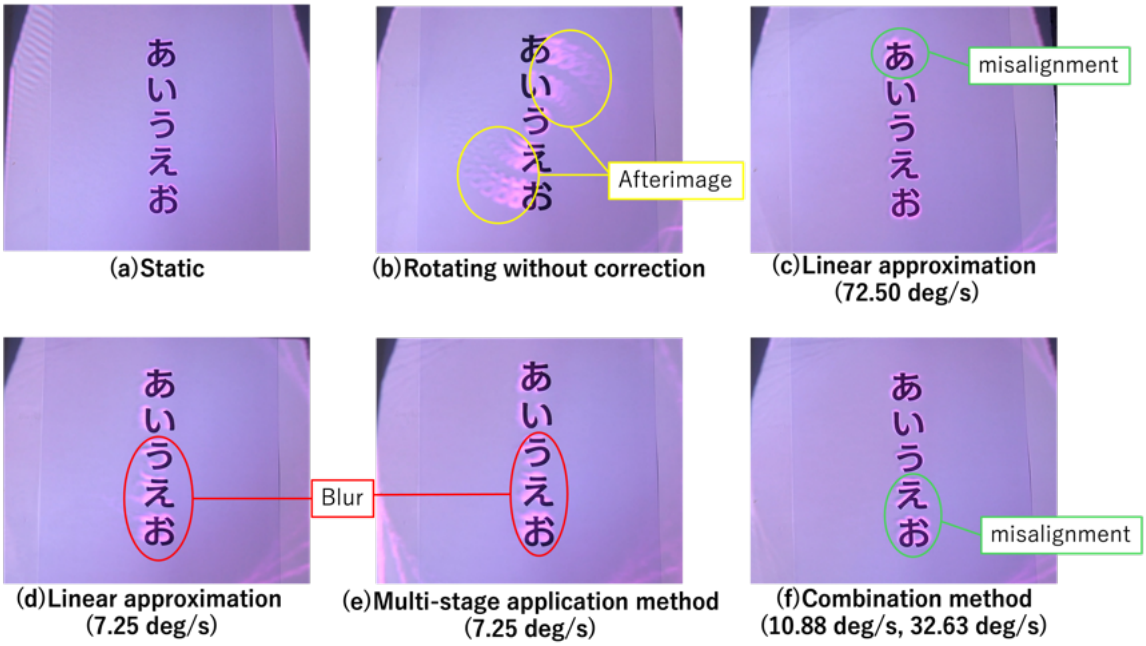


Figure 3: Comparison of projection results without compensation and with each compensation method

Fig. 3(a) shows the projection result when the projection target is stationary, where the contours of the character pattern are clearly observed. In contrast, as shown in Fig. 3(b), when the projection target is rotated without compensation, the projection image generated based on the position of the projection target at the time of image acquisition does not match the position of the projection target at the time of projection due to the rotation of the projection target. This mismatch occurs because the projection target moves during the time interval between image acquisition and projection. As a result, the projected pattern is superimposed at a position slightly shifted from the current target position, and afterimages appear around the characters. In particular, faint character patterns remain along the direction of motion, indicating the effect of projection misalignment caused by rotational motion.

In contrast, as shown in Figs. 3(c)–(f), when each compensation method is applied, the wide-range afterimages observed in Fig. 3(b) are greatly reduced. This result indicates that pixel-map-based compensation can suppress projection misalignment caused by rotational motion.

Figs. 3(c)–(f) show representative results under conditions in which the error characteristics of each compensation method can be easily observed. In Fig. 3(c), which shows the linear approximation method with a reference speed of 72.50 deg/s, local misalignment is clearly observed in part of the character pattern. This is considered to be because the linear approximation method generates the compensation image by linearly approximating the change in projection position, and therefore cannot sufficiently represent the nonlinear positional change caused by rotational motion, leaving local compensation errors.

In addition, in Fig. 3(d), which shows the linear approximation method with a reference speed of 7.25 deg/s, and Fig. 3(e), which shows the multi-stage application method with a reference speed of 7.25 deg/s, blur around the character contours is clearly observed. In particular, in the multi-stage application method, rounding errors in pixel coordinates are accumulated through the repeated application of pixel maps, which is considered to reduce the sharpness of the contours.

On the other hand, in Fig. 3(f), which shows the combination method, the afterimages observed without compensation are reduced, and the local misalignment observed in Fig. 3(c) and the contour blur observed in Figs. 3(d) and 3(e) are also relatively small. However, compared with the stationary result in Fig. 3(a), slight misalignment remains in part of the character pattern. This result suggests that combining multiple pixel maps can reduce the effects of local errors and accumulated rounding errors that occur in individual compensation methods, while quantitative evaluation is still necessary to verify the compensation accuracy.

From these results, it was confirmed that applying each compensation method greatly reduces the afterimages observed without compensation. On the other hand, local projection misalignment and contour blur may remain after compensation due to differences in the generation and application of pixel maps. In addition, even in the combination method, slight misalignment is observed compared with the stationary result. The conditions shown in Fig. 3 are representative examples for visually confirming the error characteristics of each method. Therefore, quantitative evaluation is necessary to compare the compensation accuracy of each method. In the next section, projection misalignment in each method is quantitatively evaluated by comparing the luminance peak positions at the edges between the stationary and moving conditions using an edge-enhanced image.

4.3. Evaluation of Projection Misalignment

Next, we measured the amount of projection misalignment to quantitatively evaluate the compensation accuracy of the proposed method. In this evaluation, the projection result obtained by applying appearance manipulation in the static condition was used as the reference, and the difference between this result and the projection result obtained during rotation with compensation was defined as the projection misalignment.

Specifically, we focused on the vicinity of the edges of the pattern on the projection target and measured the positions of brightness peaks generated by edge enhancement. The amount of projection misalignment was calculated from the difference between the brightness peak position in the static condition and that obtained during rotation with compensation. Fig. 4 shows the region where projection misalignment was measured.

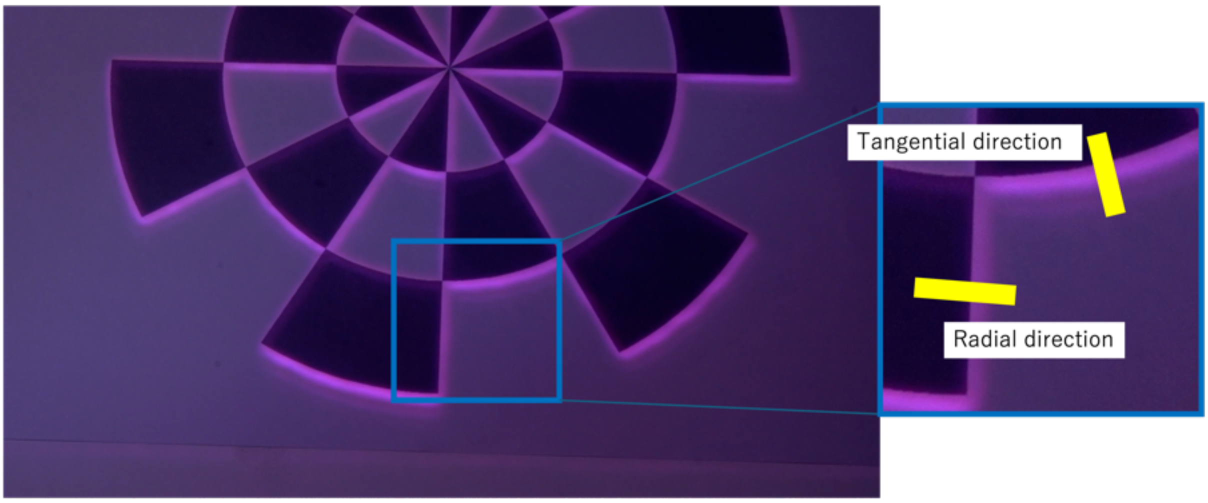


Figure 4: Measurement region for projection misalignment

In rotational motion, pixel motion appears not as simple motion in the X-axis or Y-axis direction, but as arc-shaped motion around the rotation center. Therefore, in this study, the measured projection misalignment was decomposed into tangential and radial components. This allowed us to compare which directional error each method tends to produce with respect to rotational motion.

The rotational velocity was set in 10 steps from 7.25 deg/s to 72.50 deg/s. For each velocity, projection results were acquired for the case without compensation and for the cases using each compensation method. Then, the amounts of misalignment in the tangential and radial directions were measured at each velocity, and the compensation accuracy of each method was compared.

5. Results and Discussion

5.1. Evaluation Results of Projection Misalignment

Fig. 5 shows the projection misalignment in the tangential direction on the left and the projection misalignment in the radial direction on the right. Table 1 shows the average misalignment for each method.

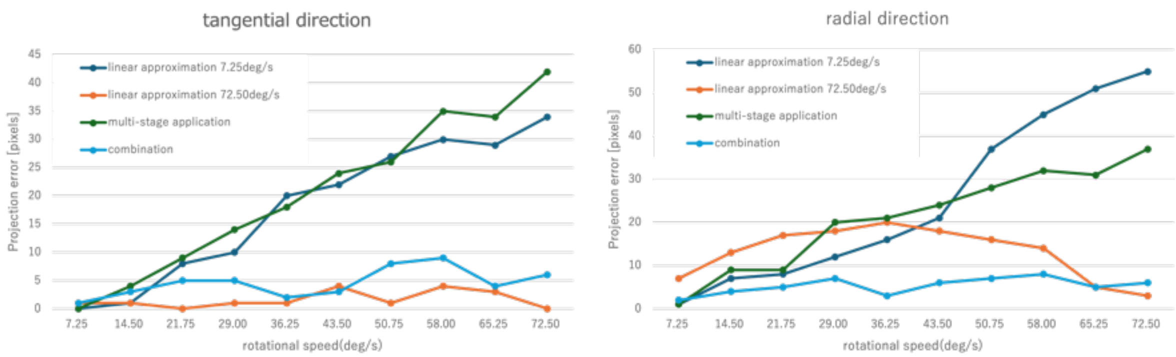


Figure 5: Projection misalignment in the tangential and radial directions

Table 1
Average projection alignment error in tangential and radial directions.

Method	Tangential direction [pixel]	Radial direction [pixel]
Combination method	4.6	5.3
Multi-stage application method	20.6	21.2
Linear approximation (7.25 deg/s)	18.1	25.3
Linear approximation (72.50 deg/s)	1.6	13.1

In the tangential direction, the linear approximation method with the reference rotational velocity of 72.50 deg/s showed the smallest average misalignment, which was 1.6 pixels. This result is considered to be due to the small scaling factor required when applying the reference pixel map. Under the high-speed conditions, the rotational displacement that occurred during one processing interval was close to the displacement represented by the reference pixel map acquired at 72.50 deg/s. Therefore, only a small amount of scaling was required, which reduced the approximation error introduced by the linear interpolation of the pixel map.

On the other hand, as shown in the left graph of Fig. 5, the linear approximation method with the reference rotational velocity of 7.25deg/s and the multi-stage application method showed increasing misalignment as the rotational speed increased. In particular, their tangential misalignment became large in the high-speed range from 50.75 deg/s to 72.50 deg/s. For the linear approximation method with the reference rotational velocity of 7.25 deg/s, the displacement obtained at a small reference velocity must be greatly enlarged when applied to high velocities, which may have increased the approximation error. For the multi-stage application method, the number of applications of the reference pixel map increases as the rotational speed becomes higher. Therefore, rounding errors at each application are considered to have accumulated, resulting in large tangential misalignment in the high-speed range.

In the radial direction, as shown in Table 1, the combination method showed the smallest average misalignment, which was 5.3 pixels. In addition, as shown in the right graph of Fig. 5, the combination method kept the radial misalignment below 10 pixels under all velocity conditions.

The linear approximation method with the reference rotational velocity of 72.50 deg/s showed a small average misalignment in the tangential direction, as shown in Table 1. However, its average radial misalignment was 13.1 pixels. The right graph of Fig. 5 also shows that the radial misalignment of this method remains larger than that of the combination method over a wide range of rotational speeds.

This is considered to be because approximating rotational motion as a linear displacement caused the difference from the actual arc-shaped pixel motion to appear as radial misalignment.

From these results, as shown in Table 1, when only the tangential direction is considered, the linear approximation method with the reference rotational velocity of 72.50 deg/s showed the smallest average misalignment. This tendency can also be observed in the left graph of Fig. 5, where this method maintained small tangential misalignment over the evaluated velocity range.

In the radial direction, as shown in Table 1, the combination method showed the smallest average misalignment. The right graph of Fig. 5 also shows that the combination method kept the radial misalignment small under all velocity conditions.

Although the tangential misalignment of the combination method was not the smallest, its average tangential misalignment was 4.6 pixels, which was smaller than those of the linear approximation method with the small reference velocity and the multi-stage application method, as shown in Table 1. Therefore, when both the tangential and radial directions are considered, the combination method is considered to be effective because it greatly suppresses radial misalignment while maintaining relatively small tangential misalignment.

5.2. Discussion of the Compensation Methods

The linear approximation method can handle arbitrary rotational velocities using a single reference pixel map. However, when it is applied to velocities that are far from the reference velocity, the displacement of the pixel map must be greatly enlarged or reduced. Therefore, the arc-shaped pixel motion in rotational motion may not be sufficiently represented, and radial misalignment is likely to occur. In fact, the linear approximation method with the reference rotational velocity of 72.50 deg/s showed the smallest average misalignment in the tangential direction, whereas its average misalignment in the radial direction was larger than that of the combination method.

In the multi-stage application method, a pixel map corresponding to a small amount of rotational motion is applied multiple times to handle arbitrary rotational velocities. Because the reference velocity is small in this method, the number of pixel-map applications increases in the high-speed range. As a result, rounding errors generated at each application are considered to have accumulated, leading to large projection misalignment.

In the combination method, multiple pixel maps acquired at different reference rotational velocities are used. Therefore, compared with a method that depends on a single reference velocity, a compensation pixel map can be generated while reducing the difference between the actual rotational velocity and the reference velocity. This is considered to have suppressed both the error caused by linear approximation and the accumulated error caused by repeated pixel-map applications in the multi-stage application method.

However, the compensation accuracy of the combination method depends on the reference velocities and the number of reference pixel maps used. Increasing the number of reference pixel maps may reduce the approximation error by allowing the use of a reference velocity closer to the actual rotational velocity, but it also increases the calibration cost and the amount of stored map data. Therefore, the trade-off between compensation accuracy and calibration cost should be further investigated.

Another limitation of this study is that the projection target is assumed to rotate around a fixed and known rotation center. In practical scenarios, the rotation center may be unknown, may shift during motion, or the target may undergo combined translational and rotational motion. To address this limitation, rotation-center estimation from optical flow observed in the camera image while keeping the projector-camera system fixed should be investigated in future work. Since pixel motion caused by rotation follows arc-shaped trajectories around the rotation center, the rotation center could be estimated from the displacement vectors of multiple pixels. However, such estimation may introduce new accuracy issues depending on the object shape, texture, and reliability of optical flow. Furthermore, by combining the proposed method with previous pixel-map-based compensation methods for translational motion, the method could be extended to targets undergoing simultaneous translation and rotation.

6. Application Examples

In this section, we demonstrate the application of the proposed projection misalignment compensation method to several types of appearance manipulation. As examples, saturation enhancement and monochrome conversion are applied to a rotating sushi image, edge enhancement is applied to a carousel image, and hue shift is applied to an SAR logo image. The following examples qualitatively demonstrate that the results of appearance manipulation can be presented while maintaining alignment with a rotating projection target. For each example, the appearance under white illumination, the manipulated result, and enlarged views of representative regions are presented to illustrate the resulting appearance changes.

6.1. Saturation Enhancement for Freshness Manipulation

Fig. 6 shows the results of applying saturation enhancement and monochrome conversion to the rotating sushi image. The figure shows the appearance under white illumination, the result of saturation enhancement with compensation, the result of monochrome conversion with compensation, and enlarged views of regions of interest.

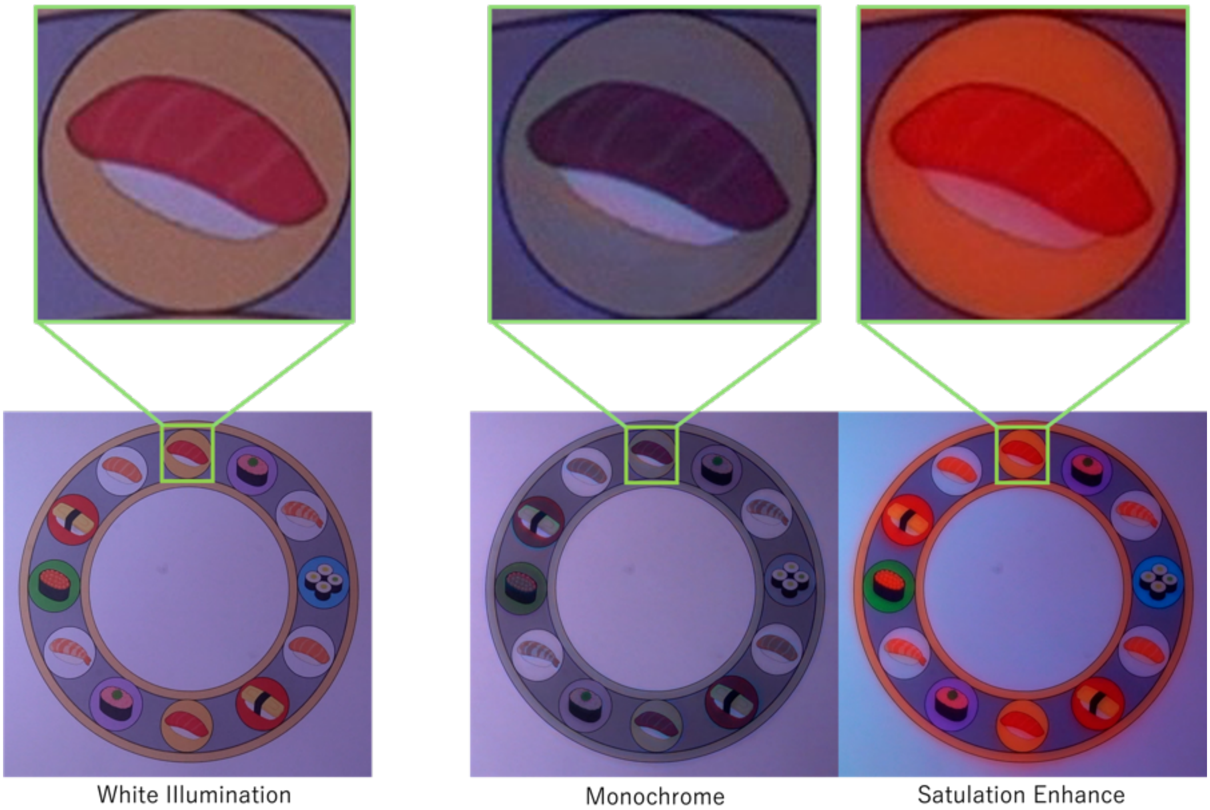


Figure 6: Examples of saturation enhancement and monochrome conversion applied to a rotating sushi image. The appearance under white illumination, the saturation-enhanced result, the monochrome conversion result, and enlarged views are shown.

In the saturation enhancement result, the red and green regions of the sushi toppings appear more vivid than in the appearance under white illumination. In particular, the enlarged views of the topping region show that the color differences are more clearly observed after saturation enhancement. This confirms that appearance manipulation that enhances color impression can be presented even on a projection target during rotational motion.

For monochrome conversion, the color of the projection target is suppressed compared with the appearance under white illumination and the saturation enhancement result, and the appearance changes to one closer to gray. However, because projection superimposes light onto the projection target surface, it is difficult to completely remove the color components originally present on the projection target and the influence of ambient light. Therefore, the monochrome conversion result is observed not as complete achromatization but as an appearance in which color components are reduced. Since color saturation is associated with the perception of freshness, these results suggest that appearance manipulation related to freshness can be maintained even for food served on a rotating table by compensating for projection misalignment during rotational motion.

6.2. Edge Enhancement for Watercolor-Like Stylization

Fig. 7 shows the result of applying edge enhancement to a carousel image. Fig. 7(a) shows the appearance of the entire carousel image under white illumination and the enlarged region around the horse contours used for comparison. Fig. 7(b) shows the results of applying edge enhancement with multiple colors to the enlarged region shown in Fig. 7(a).

Under white illumination, the contours, legs, and decorative lines of the horses are observed relatively faintly. In contrast, when edge enhancement is applied, as shown in Fig. 7(b), the projected colors are superimposed around the contours and legs of the horses, making the shape and fine details of the projection target easier to recognize. Furthermore, the projected colors added around the contours are perceived not only on the contour lines themselves, but also as if they spread into the inner regions. Therefore, although the colors are mainly added around the contours, the colors of the horse bodies and the background appear to change as well. This is considered to be caused by the watercolor effect, in which colors added near contours are perceived as spreading into the inside of the projection target.

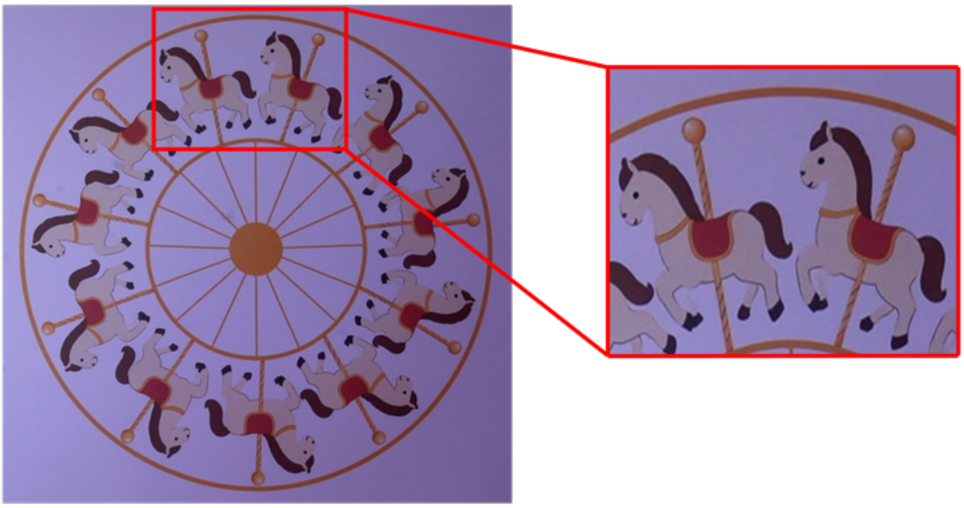
In addition, changing the color used for edge enhancement changes not only the visibility of the contours but also the overall impression of the image. These results show that the proposed method can present appearance manipulation on a rotating target while suppressing projection misalignment and can produce a watercolor-like visual effect. This example suggests that the proposed framework may be applicable to stylized appearance manipulation for objects whose visual impression is difficult to modify such as brown hair of the pony by conventional projection alone.

6.3. Color Shift for Design Support

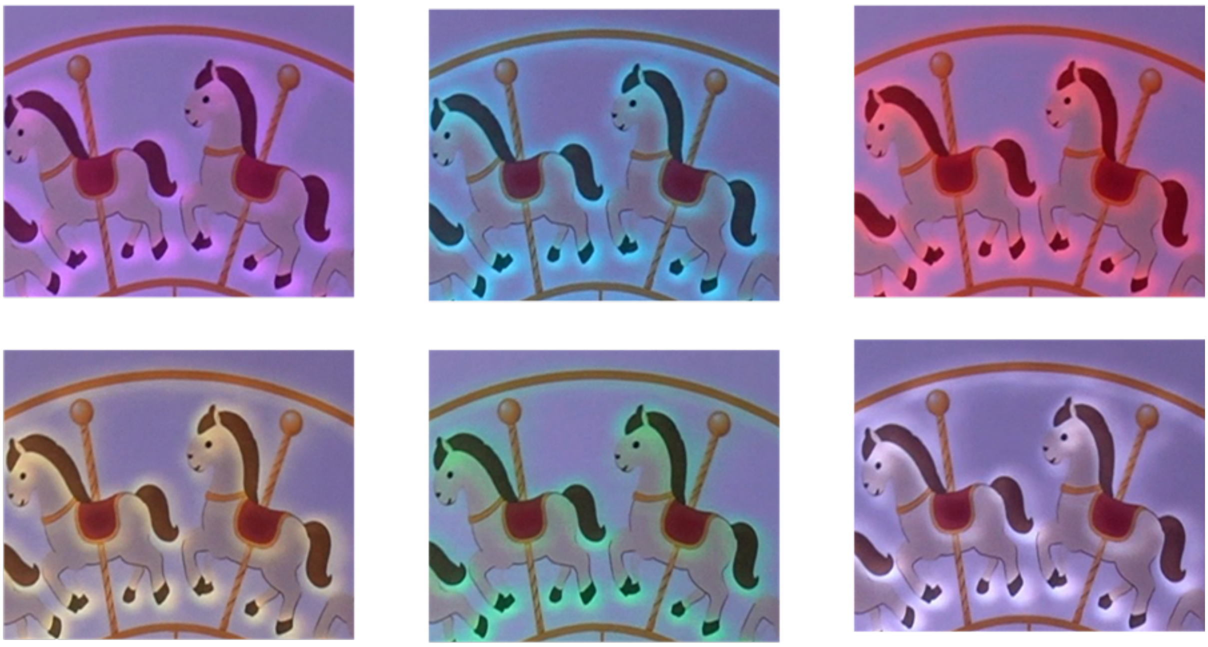
Fig. 8 shows the result of applying hue shift to the SAR logo image. Fig. 8(a) shows the appearance under white illumination and the results of applying positive and negative hue shifts during rotational motion in chronological order. Each image shows the projection result from time $t1$ to $t4$, and it can be seen that the logo moves along an arc-shaped trajectory over time. Fig. 8(b) shows enlarged views of a part of Fig. 8(a) and indicates the region where projection misalignment occurs.

As shown in Fig. 8(a), the positive hue shift changes the color of the SAR logo to a purplish color, whereas the negative hue shift changes it to a cyan color. Although the position and orientation of the logo change from $t1$ to $t4$, the hue-shifted projection results are superimposed on the inner regions of the SAR characters. This indicates that, even when the projection target moves along an arc-shaped trajectory, the proposed compensation method can present appearance manipulation results on surface regions such as the inside of the characters.

On the other hand, as shown in Fig. 8(b), projection misalignment is partially observed near the contour of the logo. In particular, when the positive hue shift is applied, the projection misalignment near the contour is relatively noticeable. This is considered to be because the purplish projected color



(a)White Illumination



(b) Enlarged view

Figure 7: Example of edge enhancement applied to a carousel image. (a) Appearance under white illumination and the enlarged region. (b) Edge enhancement results using multiple colors for the enlarged region.

obtained by the positive hue shift has a relatively large color difference from the background under white illumination and from the original logo color. Therefore, even a slight misalignment near the boundary is easily perceived as color leakage outside the contour. In contrast, in the negative hue shift, the projected color becomes cyan, which has a relatively small color difference from the original logo color and the background. As a result, even when a similar amount of misalignment occurs, it is less visually noticeable.

At the contour region, even a slight displacement between the boundary of the projection target and that of the projected image can be visually noticeable as color leakage or boundary mismatch. In addition, in rotational motion, the moving direction and amount of motion of each pixel vary depending on its position relative to the center of rotation. Therefore, compensation errors are likely to be observed,

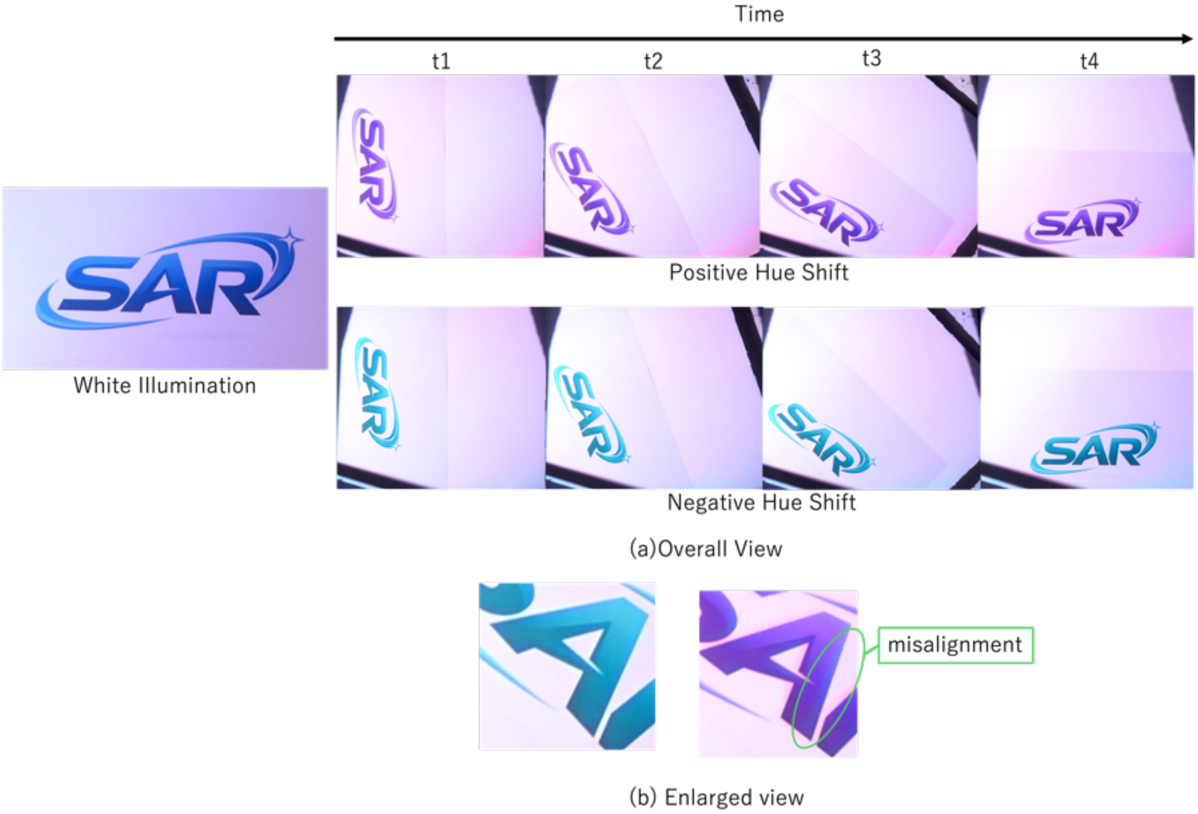


Figure 8: Example of hue shift applied to the SAR logo image. (a) Appearance under white illumination and results of applying positive and negative hue shifts during rotational motion. (b) Enlarged views of a partial region and the region where projection misalignment occurs.

especially around contours and boundary regions.

Since hue shift can be used to present different color variations of an object, these results suggest that the proposed method can support product display applications in which the appearance of an object is dynamically changed while it is displayed on a rotating turntable.

Through these application examples, we demonstrated that the proposed projection method can support various types of appearance manipulation for rotating targets, including manipulations related to color impression, stylized visual effects, and color design. However, slight projection misalignment can be visually noticeable in contour regions, indicating that improving compensation accuracy around boundary regions remains a future issue.

7. Conclusion

In this study, we investigated a method for compensating projection misalignment caused by processing latency for the purpose of appearance manipulation on a projection target undergoing rotational motion. Specifically, we targeted the case in which the projection target rotates around a fixed rotation center, and generated a pixel map corresponding to rotational motion from the correspondence between camera images and projection images before and after rotational motion. Furthermore, to support arbitrary rotational velocities, we compared three methods: a linear approximation method, a multi-stage application method, and a combination method.

In the experiment, the projector-camera system was fixed, and the projection target placed on a rotating stage was rotated to evaluate projection misalignment caused by rotational motion of the projection target. The results showed that, in the tangential direction, the linear approximation method with the reference rotational velocity of 72.50 deg/s showed the smallest average misalignment, whereas in the radial direction, the combination method showed the smallest average misalignment. The combination method also greatly suppressed radial misalignment while maintaining relatively small tangential misalignment. These results suggest that, when both the tangential and radial directions are considered, the method that combines multiple pixel maps with different reference velocities is effective for compensating projection misalignment under rotational motion. Furthermore, as application examples using the combination method, we demonstrated monochrome conversion and saturation enhancement for a rotating sushi image, edge enhancement for a carousel image, and hue shift for a logo image. These results confirmed the possibility of presenting several types of appearance manipulation on a projection target undergoing rotational motion.

In future work, we will extend the proposed method to motion with an unknown or time-varying rotation center and to combined translational and rotational motion. For motion with an unknown rotation center, we will investigate rotation-center estimation based on optical flow observed in the camera image. For combined motion, we will examine how the proposed rotational compensation can be integrated with previous pixel-map-based compensation methods for translational motion by separating the observed displacement into translational and rotational components.

Acknowledge

This work was supported by JSPS KAKENHI Grant Numbers JP24K02979, JP25H01886.

Declaration on Generative AI

During the preparation of this work, the author(s) used ChatGPT in order to: Grammar and spelling check, language editing, and improvement of clarity and readability. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.

References

[1] R. Raskar, G. Welch, K.-L. Low, D. Bandyopadhyay, Shader lamps: Animating real objects with image-based illumination, in: Proceedings of the 12th Eurographics Workshop on Rendering Techniques, 2001, pp. 89–102.

[2] T. Amano, T. Kagawa, Projection alignment correction for in-vehicle projector-camera system, in: International Conference on Artificial Reality and Telexistence and Eurographics Symposium on Virtual Environments, 2021.

[3] T. Kano, T. Amano, Projection alignment correction for appearance manipulation under two-axis motion, in: Meeting on Image Recognition and Understanding, 2022.

[4] D. Bandyopadhyay, R. Raskar, H. Fuchs, Dynamic shader lamps: Painting on movable objects, in: Proceedings of the IEEE and ACM International Symposium on Augmented Reality, 2001, pp. 207–216.