IPSJ Transactions on Computer Vision and Applications Vol. 2 200–213 (Nov. 2010)

Research Paper

Video Completion for Generating Omnidirectional Video without Invisible Areas

Norihiko Kawai,[†1,†2] Kotaro Machikita,[†1]
Tomokazu Sato[†1] and Naokazu Yokoya[†1]

Omnidirectional multi-camera systems cannot capture entire fields of view because of their inability to view areas directly below them. Such invisible areas in omnidirectional video decrease the resulting realistic sensation experienced when using a telepresence system. In this study, we generate omnidirectional video without invisible areas using an image completion technique. The proposed method compensates for the change in appearance of textures caused by camera motion and searches for appropriate exemplars considering three-dimensional geometric information. In our experiments, the effectiveness of our proposed method has been demonstrated by successfully filling in missing regions in real video sequences captured using an omnidirectional multi-camera system.

1. Introduction

Telepresence systems that enable us to experience remote sites are expected to be used in various fields such as virtual sightseeing, education, and digital archiving. In these fields, omnidirectional video captured with an omnidirectional multi-camera system (OMS) composed of radially arranged cameras is used [5),6)]; however, as shown in **Fig. 1**, an ordinary OMS cannot capture entire fields of view because of the absence of a camera unit to capture areas directly below the OMS. Such invisible areas decreases the realistic sensation created by a telepresence system. In this study, we aim at generating omnidirectional video without invisible areas for offline telepresence applications by completing the missing regions directly below the OMS to achieve telepresence with highly realistic sensation.

†1 Graduate School of Information Science, Nara Institute of Science and Technology
†2 Research Fellow of the Japan Society for the Promotion of Science

Fig. 1 Unwrapped spherical panoramic image (left) and a standard perspective image generated from the panoramic image (right). The black regions on these images are caused by the system's inability to capture the area directly below it.

Numerous video completion methods have been proposed for video captured with a monocular camera. While some have successfully completed missing regions, most of the methods assume that the appearance of objects rarely changes between frames. It is difficult for such methods to successfully complete missing regions in omnidirectional video in which the appearance of textures may change drastically.

In this paper, we propose a new method to complete missing regions in omnidirectional video considering the change in appearance of textures; the proposed method is based on the following three assumptions: (1) all cameras in the given OMS are fully calibrated; (2) the ground directly below the given OMS is roughly planar; and (3) textures in the lower regions of a frame are captured from different viewpoints.

Our procedure is summarized as follows. First, we compensate for the change in appearance of textures caused by camera motion by projecting omnidirectional images onto the plane fitted to feature points around the ground; these feature points are acquired using a structure-from-motion (SFM) technique. Second, using the geometric relationship between the plane and camera motion, we identify data regions in which appropriate example textures for missing regions may exist. Third, we complete the missing regions by minimizing an energy function based on pattern similarity between the missing and data regions. Our contributions are the suggestion of a novel pipeline of the above processes for generating high-

quality omnidirectional video without invisible areas and the validation using real video sequences, which have yet to be realized in the literature.

This paper is organized as follows. Related work is reviewed in Section 2. The proposed method for generating omnidirectional video without invisible areas is described in Section 3. Experiments using two video sequences and our results are summarized in Section 4. Section 5 concludes our work and provides direction for the future.

2. Related Work

For video completion, image completion methods for a still image [1),2),4),8),9)] can be applied to each frame in a video; however, textures may discontinuously change between successive frames because these methods use only spatial information. Therefore, in video completion, we typically use not only spatial information, but also temporal information.

Video completion methods using spatial and temporal information can be classified into two categories: one that does not use the motion information of a scene and one that does. Methods that do not use motion information complete missing regions using spatial-temporal image volumes as exemplars [3),16)]. These methods define the similarity of local volumes between the missing region and the rest of the video and copy the similar local volumes or optimize a similarity-based objective function. While these methods can generate continuous textures through successive frames, an occluded scene is not always used as an exemplar, even if it appears in different frames because of the movement of the camera. As a result, the textures that do not actually exist behind the occlusion may appear in the missing regions. In addition, these methods do not consider the change in the appearance of textures caused by the camera motion. In summary, using these methods, it is difficult to successfully complete missing regions in omnidirectional video in which the appearance of textures changes between frames.

Methods that use motion information for video completion are based on the correspondence of pixels between a target and other frames. The pixel correspondences are determined based on the estimated motion of objects or that of a camera. These methods can be classified into two types: one that uses the motion of objects on an image plane [7),10)–12),18)] and one that uses the orientation of a camera [15),17)].

Using the motion of objects on an image plane, Litvin, et al. [10)] and Matsushita, et al. [11)] have proposed methods that estimate the motion of textures in missing regions. Specifically, they use optical flows across the whole image and copy the pixel values to missing regions from different frames based on the estimated motion; however, it is difficult for these methods to determine appropriate optical flows in omnidirectional video because the appearance of textures changes between successive frames and the motion of pixels in large missing regions cannot be accurately estimated by such two-dimensional interpolation. To treat relatively complicated motion in missing regions, other methods that use the motion model of objects have been proposed [7),12),18)]. In these methods, the motion of objects is modeled and the moving objects are reproduced in missing regions based on the motion model; however, the motion model is designed for cyclic motion and these methods cannot handle complex three-dimensional motion.

Using the motion of a camera, Shen, et al. [15)] and Yamashita, et al. [17)] have proposed methods that use a fixed-viewpoint pan-tilt camera. Missing regions are successfully completed using the same direction of lighting captured in neighboring frames; however, these methods cannot be applied to video captured with a freely moving camera since the fixed viewpoint is the indispensable condition to properly correspond pixels across different frames.

To complete the missing region directly below an OMS in omnidirectional video, it is essential to consider the appearance changes of textures, which has yet to be considered in the literature. Furthermore, two-dimensional optical flows cannot accurately estimate corresponding pixels in omnidirectional vision because the missing region is relatively large and the motion of corresponding pixels is quite complex because of a characteristic of optical distortion in omnidirectional video.

To address these problems, the proposed method utilizes three-dimensional information (including the motion of a camera and the structure around the missing region directly below the OMS) simultaneously estimated by a SFM technique for omnidirectional video. From the three-dimensional information, the appropriate appearance of textures can be generated. In addition, the relatively accurate correspondence of pixels between quite different frames can be determined by considering the projection model of the OMS.

3. Generation of Omnidirectional Video without Invisible Areas

3.1 Proposed Pipeline

Figure 2 shows the pipeline of the proposed method. Following the figure, the position and posture of an OMS and the three-dimensional positions of feature points are estimated using SFM for omnidirectional video [14] (A). Next, a plane for each frame is fitted to feature points around the ground (B). An image sequence projected on each of the fitted planes is generated from the omnidirectional video for compensating for the change in appearance of textures (C). A missing region in the projected image of each frame is completed frame-by-frame by minimizing an energy function based on the similarity of textures between the missing and data regions (D). In this process, first, data regions in which appropriate exemplars for missing regions may exist are determined on the projected image (D-i). The energy function is minimized by repeating two processes: a parallel search for similar textures (D-ii) and a parallel update of all pixel values (D-iii). Finally, omnidirectional video with no invisible areas is generated by re-projecting the completed images onto the spherical panoramic video (E). Processes (B), (C) and (D) are described in detail in the following subsections.

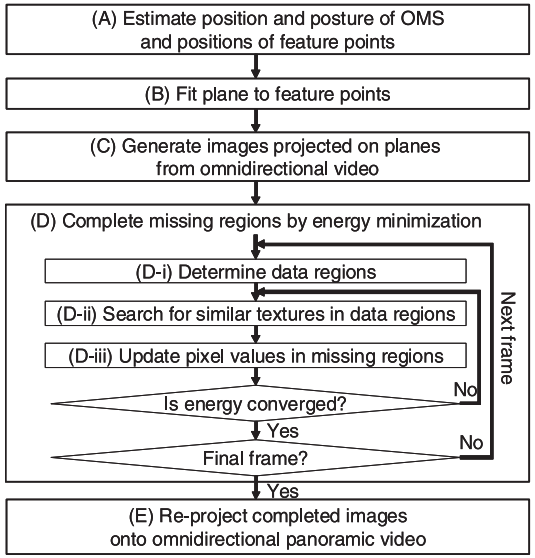


Fig. 2 Pipeline of the proposed video completion method for omnidirectional video.

3.2 Generation of Images Projected on Planes

In ordinary omnidirectional video, it is difficult to complete missing regions using originally captured textures, because the appearance of textures changes because of camera motion. In this study, we generate an image sequence in which the appearance of textures is compensated by projecting omnidirectional video to a plane of each frame. The method for generating an image projected on a plane in the f-th frame is described below, covering processes (B) and (C).

Corresponding to process (B) in Fig. 2, a plane representing the ground is fitted to feature points, where such points are selected from the points obtained through the SFM technique [14] in process (A). Here, points that satisfy the following conditions are selected: (i) the point exists in the spherical area with radius l whose center is a projection center of a representative camera unit in an OMS; and (ii) the height z of the point in the world coordinate system is $(p < z < p + m)$ (p and m are constants), as shown in **Fig. 3**. Once points are selected, the expression of the plane in the world coordinate system is calculated by the least-squares method.

Next, as shown in **Fig. 4**, an image is generated by projecting omnidirectional video to the estimated plane (process (C) in Fig. 2). Here, a missing region should be included in the generated image and the scale and orientation of textures

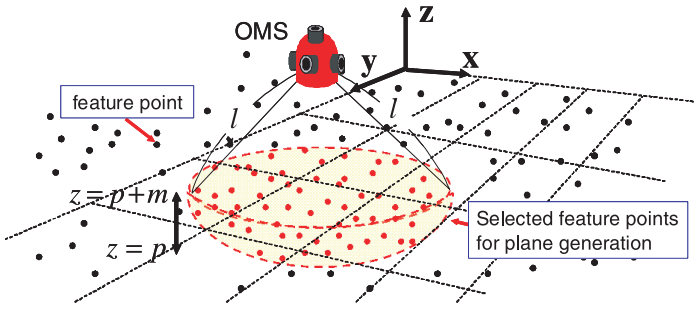


Fig. 3 Selection of feature points around a missing region.

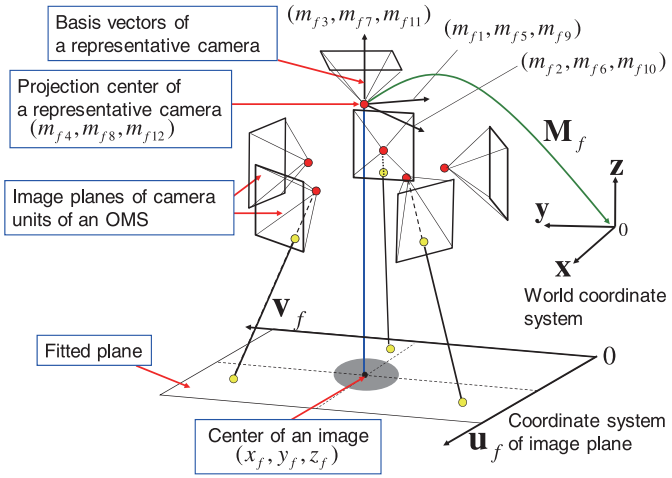


Fig. 4 Generation of an image projected on a plane.

should be invariant through all frames for image completion described below in Section 3.3. The method to generate the projected image of the f-th frame is described in the following paragraphs.

First, the center of the projected image is determined. For a missing region to be the center of the projected image, a point on the plane and just under the projection center of a representative camera is set as the center. Concretely, the transformation matrix $\mathbf{M}_f$ from the representative camera coordinate system of the f-th frame to the world coordinate system is expressed as follows:

$$\mathbf{M}_f = \begin{pmatrix} m_{f1} & m_{f2} & m_{f3} & m_{f4} \\ m_{f5} & m_{f6} & m_{f7} & m_{f8} \\ m_{f9} & m_{f10} & m_{f11} & m_{f12} \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{1}$$

The central coordinate of the projected image (x_f, y_f, z_f) can be expressed using parameter t as follows:

$$\begin{pmatrix} x_f \\ y_f \\ z_f \end{pmatrix} = \begin{pmatrix} m_{f4} \\ m_{f8} \\ m_{f12} \end{pmatrix} + t \begin{pmatrix} m_{f3} \\ m_{f7} \\ m_{f11} \end{pmatrix}. \tag{2}$$

From the expression of the plane and that of the straight line, Eq. (2), the intersection point can be calculated as the central coordinate (x_f, y_f, z_f).

Next, basis vectors of the projected image are determined. To prevent the rotation of textures in the projected image between different frames, basis vectors $(\mathbf{u}_f, \mathbf{v}_f)$ in the world coordinate system are set so as to satisfy the following equation:

$$\mathbf{u}_f \cdot \mathbf{y} = 0, \tag{3}$$

where $\mathbf{y}$ is one of the basis vectors of the world coordinate system.

Using the calculated center position (x_f, y_f, z_f) and basis vectors $(\mathbf{u}_f, \mathbf{v}_f)$, the relationship between coordinate (u_f, v_f) in the image coordinate system and three-dimensional position $(x_{fuv}, y_{fuv}, z_{fuv})$ in the world coordinate system is expressed as follows:

$$\begin{pmatrix} x_{fuv} \\ y_{fuv} \\ z_{fuv} \end{pmatrix} = \begin{pmatrix} x_f \\ y_f \\ z_f \end{pmatrix} + ru_f\mathbf{u}_f + rv_f\mathbf{v}_f, \tag{4}$$

where the size of a pixel is $r \times r$ and the center in the image is origin in the image coordinate system. Note that the scale change of textures in the projected image are also prevented by fixing the size of a pixel through all frames.

3.3 Video Completion through Energy Minimization

In process (D) of Fig. 2, we apply an image completion method for a still image to video frames, proceeding in a frame-by-frame manner. In this study, our previous image completion method for a still image [8] is employed because it can generate geometrically and optically natural textures by minimizing an energy function based on the pattern similarity that considers brightness changes of textures. Although the original image completion method for a still image defines data regions based on only the given image, we identify data regions pixel by pixel from multiple frames using geometric information to preserve the temporal continuity.

In the following subsections, we describe the definition of the energy function, a method for determining data regions (process (D-i) in Fig. 2), and a method for minimizing the energy function (processes (D-ii) and (D-iii) in Fig. 2).

3.3.1 Definition of Energy Function Based on Pattern Similarity

As shown in **Fig. 5**, a missing region in the projected image of the target (f-th) frame is completed using an energy function based on the similarity of textures

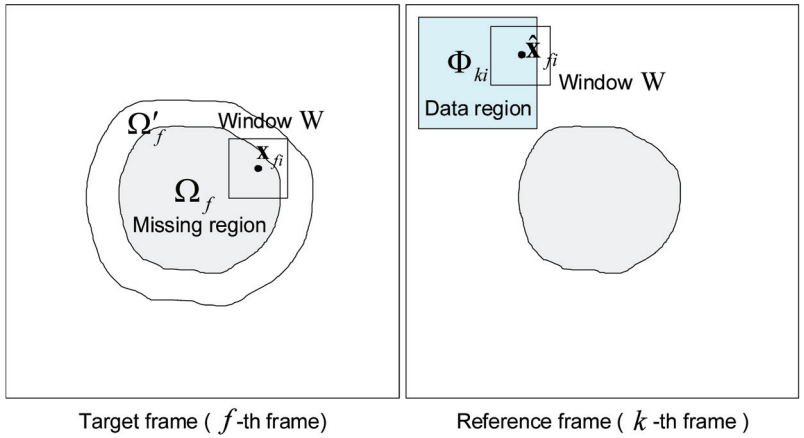


Fig. 5 Missing and data regions in projected images for completion process.

between region Ω'_f, including missing region Ω_f in the f-th frame, and data region Φ_{ki} in the reference (k-th) frame ($k \neq f$). Ω'_f is the expanded area of missing region Ω_f in which there is a central pixel, $\mathbf{x}_{fi}$, of a square window W overlapping region Ω_f. Each data region Φ_{ki} corresponding to each pixel $\mathbf{x}_{fi}$ in region Ω'_f is individually determined using the method described in Section 3.3.2. Energy function E is defined as the weighted sum of each sum of squared differences (SSD) between textures around pixel $\mathbf{x}_{fi}$ in region Ω'_f and $\hat{\mathbf{x}}_{fi}$ in data region Φ_{ki},

$$E = \sum_{\mathbf{x}_{fi} \in \Omega'_f} w_{\mathbf{x}_{fi}} SSD(\mathbf{x}_{fi}, \hat{\mathbf{x}}_{fi}), \tag{5}$$

where $w_{\mathbf{x}_{fi}}$ is the weight of pixel $\mathbf{x}_{fi}$. This weight is set as 1 if $\mathbf{x}_{fi}$ is inside region $\Omega'_f \cap \overline{\Omega_f}$ because pixel values in this region are fixed; otherwise, $w_{\mathbf{x}_{fi}} = g^{-d}$ (where d is the distance from the boundary of Ω_f and g is a constant) because pixel values around the boundary have higher confidence than those in the center of the missing region.

$SSD(\mathbf{x}_{fi}, \hat{\mathbf{x}}_{fi})$, which represents the similarity of textures around pixel $\mathbf{x}_{fi}$ and $\hat{\mathbf{x}}_{fi}$, is defined as follows:

$$SSD(\mathbf{x}_{fi}, \hat{\mathbf{x}}_{fi}) = \sum_{\mathbf{q} \in W} \{I(\mathbf{x}_{fi} + \mathbf{q}) - \alpha_{\mathbf{x}_{fi}\hat{\mathbf{x}}_{fi}} I(\hat{\mathbf{x}}_{fi} + \mathbf{q})\}^2, \tag{6}$$

where $I(\mathbf{x})$ represents the pixel value of pixel $\mathbf{x}$, $\mathbf{q}$ is a shift vector in window W, and $\alpha_{\mathbf{x}_{fi}\hat{\mathbf{x}}_{fi}}$ is the intensity modification coefficient. Note that textures around a missing region may change because of the reflection of light on the ground and the shade of the camera and operator. The intensity modification coefficient adjusts the brightness of textures in data regions to that of the missing region. In our work, $\alpha_{\mathbf{x}_{fi}\hat{\mathbf{x}}_{fi}}$ is defined as the ratio of average pixel values around pixels $\mathbf{x}_{fi}$ and $\hat{\mathbf{x}}_{fi}$ as follows:

$$\alpha_{\mathbf{x}_{fi}\hat{\mathbf{x}}_{fi}} = \frac{\sqrt{\sum_{\mathbf{q} \in W} I(\mathbf{x}_{fi} + \mathbf{q})^2}}{\sqrt{\sum_{\mathbf{q} \in W} I(\hat{\mathbf{x}}_{fi} + \mathbf{q})^2}}. \tag{7}$$

3.3.2 Determination of Data Region

In process (D-i) in Fig. 2, data regions in which appropriate exemplars for a missing region exist are determined using the geometrical pixel-by-pixel relationship between the position and posture of an OMS, estimated in process (A), and the planes, generated in process (B).

In the subsections that follow, we describe how to determine a data region (i.e., both a frame and a region) corresponding to each pixel $\mathbf{x}_{fi}$ in missing region Ω'_f in the target f-th frame. Methods to determine data regions for an initial frame and for subsequent frames are described.

Determination of Data Regions for Initial Frame

Data regions for the initial frame are determined through pixel-by-pixel consideration of the texture resolutions and the difference of frame numbers between the initial and reference frames.

First, the three-dimensional coordinate of pixel $\mathbf{x}_{fi}$ is re-projected on the image plane of a camera unit in the k-th frame. As shown in **Fig. 6**, the pixel coordinate $p_k(\mathbf{x}_{fi})$ on the k-th projected image is determined by computing the intersection point of the k-th plane with the projecting line of pixel $\mathbf{x}_{fi}$.

Next, the best frame for exemplars is determined considering the position $p_k(\mathbf{x}_{fi})$ in the projected image and the difference of frame numbers between the target and reference frames. In the projected images, the texture resolution

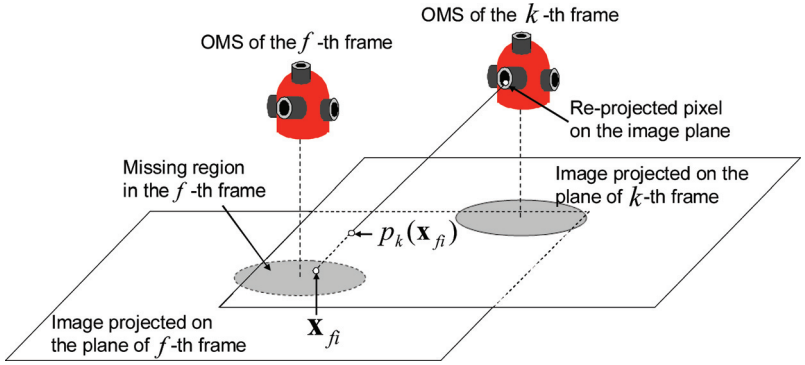


Fig. 6 Projection to the other frame.

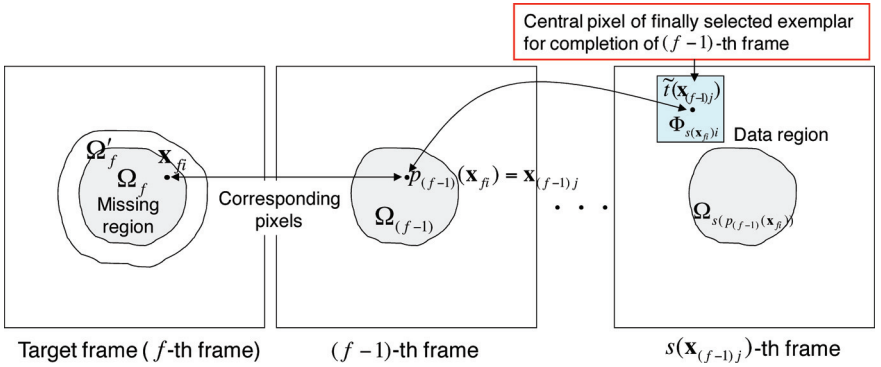


Fig. 7 Utilization of the result of the previous frame.

decreases as a pixel becomes farther from the center of the image, because textures of objects far from the camera are small in input images of an OMS. Thus, textures near the center of the image should be used as exemplars for completion. Because temporally close frames often have similar brightness values, a temporally close frame should be used. Therefore, the best frame $s(\mathbf{x}_{fi})$ is selected from candidate frames $\mathbf{K} = (k_1, \ldots, k_n)$ by applying the following equation:

$$s(\mathbf{x}_{fi}) = \underset{k \in \mathbf{K}}{\operatorname{argmin}}(\parallel p_k(\mathbf{x}_{fi}) - \mathbf{x}_{center} \parallel + \lambda |k - f|), \tag{8}$$

where candidate frames $\mathbf{K}$ are picked such that the texture pattern of the exemplar whose center is $p_k(\mathbf{x}_{fi})$ does not include the missing region. $\mathbf{x}_{center}$ is the central pixel of the k-th projected image and λ is the weight for the difference of frame numbers. The fixed square area whose center is pixel $p_{s(\mathbf{x}_{fi})}(\mathbf{x}_{fi})$ is set as data region $\Phi_{s(\mathbf{x}_{fi})i}$.

Determination of Data Regions for Subsequent Frames

In addition to the initial frame, data regions for subsequent frames are determined considering not only the resolution of textures and the difference of frame numbers but also the positions of previously selected exemplars to preserve the temporal continuity of generated textures.

In this research, the availability of the previously selected exemplars is first checked and a data regions is then determined by following steps.

(1) Pixel $p_{(f-1)}(\mathbf{x}_{fi})$ $(= \mathbf{x}_{(f-1)j})$ in the previous $((f-1)$-th) frame corresponding to $\mathbf{x}_{fi}$ is calculated based on the estimated geometry.

(2) Frame $s(\mathbf{x}_{fi})$ corresponding to $\mathbf{x}_{fi}$ is calculated by Eq. (8).

(3) If $s(\mathbf{x}_{fi})$ does not coincide with $s(\mathbf{x}_{(f-1)j})$, a data region is determined using the same approach as with the initial frame.

(4) Otherwise, the square area whose center is pixel $\tilde{t}(\mathbf{x}_{(f-1)j})$ that was selected for completion in the previous frame is set as data region $\Phi_{s(\mathbf{x}_{fi})i}$ (see **Fig. 7**).

It should be noted that the size of the data region in case (4) is set smaller than the other case, not only to preserve the temporal continuity of textures, but also decrease the computational cost.

3.3.3 Energy Minimization

We minimize energy function E from Eq. (5) by using a greedy algorithm. In our definition of E, the energy for each pixel can be treated independently if pattern pairs $(\mathbf{x}_{fi}, \hat{\mathbf{x}}_{fi})$ are fixed and the change of coefficient $\alpha_{\mathbf{x}_{fi}\hat{\mathbf{x}}_{fi}}$ in the iterative process of energy minimization is much smaller than the change of pixel values in the missing region. Thus, we repeat the following two processes until the energy converges on a minimum: a parallel search for the most similar pattern, keeping pixel values fixed (D-ii); and a parallel update of all pixel values, keeping pattern pairs fixed (D-iii). Each approach is described below.

(D-ii) Parallel Search for Similar Texture Patterns

In process (D-ii) of Fig. 2, data region Φ_{ki} $(k = s(\mathbf{x}_{fi}))$ is searched for position

$t(\mathbf{x}_{fi})$ of the most similar pattern corresponding to pixel $\mathbf{x}_{fi}$, keeping pixel values $I(\mathbf{x}_{fi})$ fixed. $t(\mathbf{x}_{fi})$ is determined pixel by pixel in parallel as follows:

$$t(\mathbf{x}_{fi}) = \hat{\mathbf{x}}_{fi} = \underset{\mathbf{x}\in\Phi_{ki}}{\operatorname{argmin}}(SSD(\mathbf{x}_{fi}, \mathbf{x})). \quad (9)$$

(D-iii) Parallel Update of Pixel Values

In process (D-iii) of Fig. 2, all pixel values $I(\mathbf{x}_{fi})$ are updated in parallel so as to minimize the energy, keeping the similar pattern pairs fixed. To calculate pixel values $I(\mathbf{x}_{fi})$, we first resolve energy E into element energy $E(\mathbf{x}_{fi})$ for each pixel $\mathbf{x}_{fi}$ in missing region Ω_f. Element energy $E(\mathbf{x}_{fi})$ can be expressed in terms of the pixel values of $\mathbf{x}_{fi}$ and $f(\mathbf{x}_{fi}+\mathbf{q})-\mathbf{q}$, and coefficient α as follows:

$$E(\mathbf{x}_{fi}) = \sum_{\mathbf{q}\in W} w_{(\mathbf{x}_{fi}+\mathbf{q})}\{I(\mathbf{x}_{fi}) - \alpha_{(\mathbf{x}_{fi}+\mathbf{q})t(\mathbf{x}_{fi}+\mathbf{q})} I(t(\mathbf{x}_{fi}+\mathbf{q})-\mathbf{q})\}^2. \quad (10)$$

The relationship between energy E and element energy $E(\mathbf{x}_{fi})$ for each pixel is then written as follows:

$$E = \sum_{\mathbf{x}_{fi}\in\Omega} E(\mathbf{x}_{fi}) + C, \quad (11)$$

where C, the energy of pixels in region $\Omega'_f \cap \overline{\Omega_f}$, is treated as a constant, because pixel values in the region and all the pattern pairs are fixed in process (D-iii). Therefore, by minimizing element energy $E(\mathbf{x}_{fi})$, total energy E can be minimized. If we assume that the change of $\alpha_{\mathbf{x}_{fi}t(\mathbf{x}_{fi})}$ is much smaller than that of pixel value $I(\mathbf{x}_{fi})$, by differentiating $E(\mathbf{x}_{fi})$ with respect to $I(\mathbf{x}_{fi})$, each pixel value $I(\mathbf{x}_{fi})$ in missing region Ω_f can be calculated in parallel as follows:

$$I(\mathbf{x}_{fi}) = \frac{\sum_{\mathbf{q}\in W} w_{(\mathbf{x}_{fi}+\mathbf{q})}\alpha_{(\mathbf{x}_{fi}+\mathbf{q})t(\mathbf{x}_{fi}+\mathbf{q})} I(t(\mathbf{x}_{fi}+\mathbf{q})-\mathbf{q})}{\sum_{\mathbf{q}\in W} w_{(\mathbf{x}_{fi}+\mathbf{q})}}. \quad (12)$$

In addition to the energy minimization algorithm described above, a coarse-to-fine approach is also employed to efficiently avoid local minima. An image pyramid is generated and energy minimization processes (D-ii) and (D-iii) are repeated from higher-level to lower-level layers. Good initial values are seeded in the lower layers by projecting results from the higher layer. In addition, the correspondences of pixels are inherited and new data regions are set so that the corresponding pixels are the center of the new data regions. This both decreases computational cost and avoids local minima.

4. Experiments and Results

In this section, we demonstrate the effectiveness of the proposed method by completing missing regions in two video sequences. For our experiments, we used a PC (CPU: Xeon 3.0 GHz, Memory: 8 GB), and as shown in **Fig. 8**, the Ladybug [13] OMS. Ladybug has six radially located camera units, their positions and postures fixed. Each camera unit can acquire 768×1,024 images. Parameters in our experiments were empirically determined and summarized in **Table 1**.

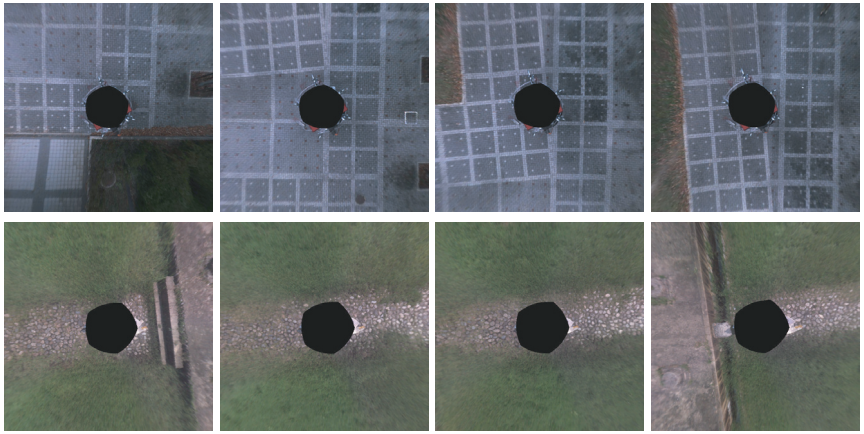
We determined a missing region in each image through the combination of two regions: (1) the region unprojected from the six input images directly below

Fig. 8 Omnidirectional multi-camera system "Ladybug".

Table 1 Experimental parameters.

For generation of panoramic and projected images			
Resolution of panoramic image	2,048 × 1,024 (pixel)		
Resolution of projected image	1,200 × 1,200 (pixel)		
Range for selecting feature points	$-1{,}000\,(\text{mm}) < z <$ height of camera $l = 6{,}000\,(\text{mm})$		
Pixel size	10 (mm) × 10 (mm)		
Weight λ	2.0		
For energy minimization			
Scale Level (Coarse-to-fine approach)	1/4	1/2	1
g in weight w	1.1		
Window size	51 × 51 (pixel)		
Data region (case (a))	11 × 11 (pixel)	5 × 5 (pixel)	5 × 5 (pixel)
Data region (case (b))	3 × 3 (pixel)	3 × 3 (pixel)	3 × 3 (pixel)

Fig. 9 First frames of input video sequences obtained by six camera units; top images are for scene (1) and bottom images are for scene (2).



1st frame 101st frame 201st frame 301st frame

Fig. 10 Projected Images; top images are for scene (1) and bottom images are for scene (2).

Ladybug; and (2) the regions projected from six input images that were manually masked, because they contain such unneeded regions as the Ladybug equipment and the shadow of the OMS operator. The following subsections describe our experiments for two video sequences in detail.

4.1 Acquisition of Input Information

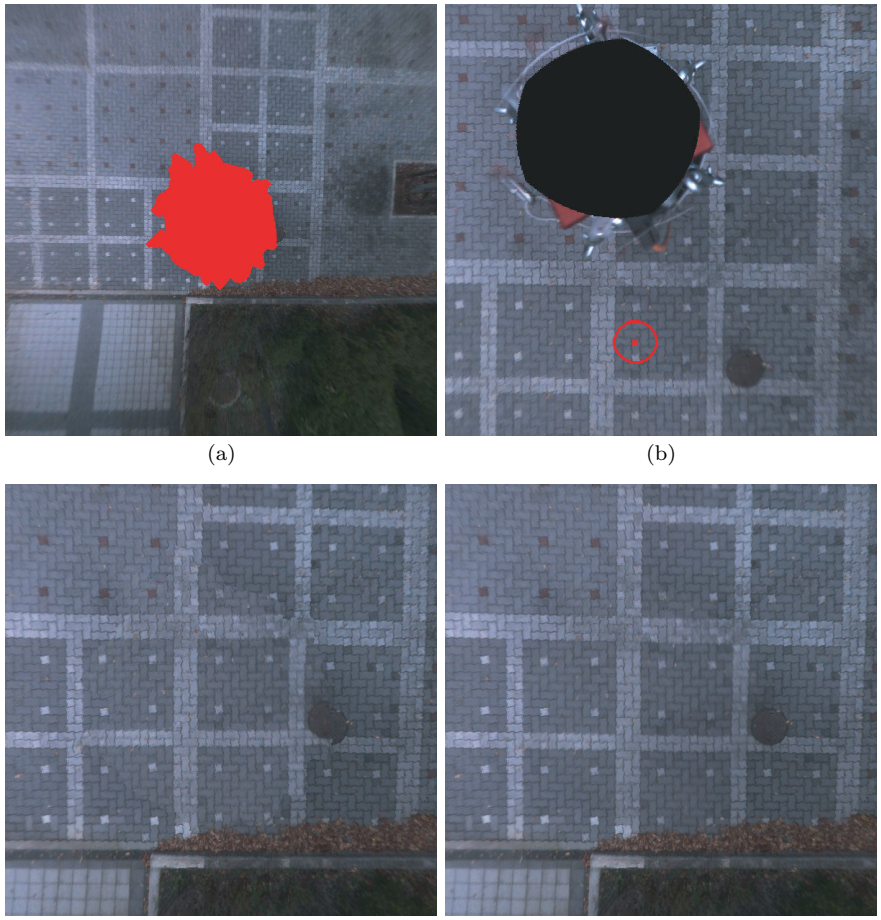
In our experiments, two video sequences were captured from Ladybug while the person on whom it was mounted was walking. Each video consists of 301 frames of omnidirectional video, totaling 1,806 images across the six cameras. **Figure 9** shows the first frames of two video sequences captured by Ladybug. In scene (1), the top six images of Fig. 9, the ground is almost planar through all frames. In scene (2), the bottom six images of Fig. 9, the ground is slightly rugged because of various stones, and there are steps near missing regions at the beginning and end of the scene.

In the following, experiments of completion for images projected on planes and a prototype telepresence system using the omnidirectional video without invisible areas are presented.

4.2 Completion for Projected Images

We generated $1{,}200 \times 1{,}200$ projected images using the method described above in Section 3.2. Of the 301 frames, four selected images of scenes (1) and (2) are shown in **Fig. 10**. The round black regions shown are unprojected regions below the Ladybug. As shown in the figure, textures of tiles in scene (1) are uniform regardless of the position of pixels; furthermore, textures of the same objects do not rotate in each frame. As a result, we confirm that appropriate exemplars were generated in the projected images. Similarly, textures of stones in scene (2) are also the same and do not rotate from frame to frame; however, the brightness of stones are different between the left and right sides of the missing regions. Furthermore, the brightness of the stones changes dramatically from frame to frame. Given that strong sunlight comes from the right, we observe that the brightness of objects has changed because of the reflection of the sunlight on the stones.

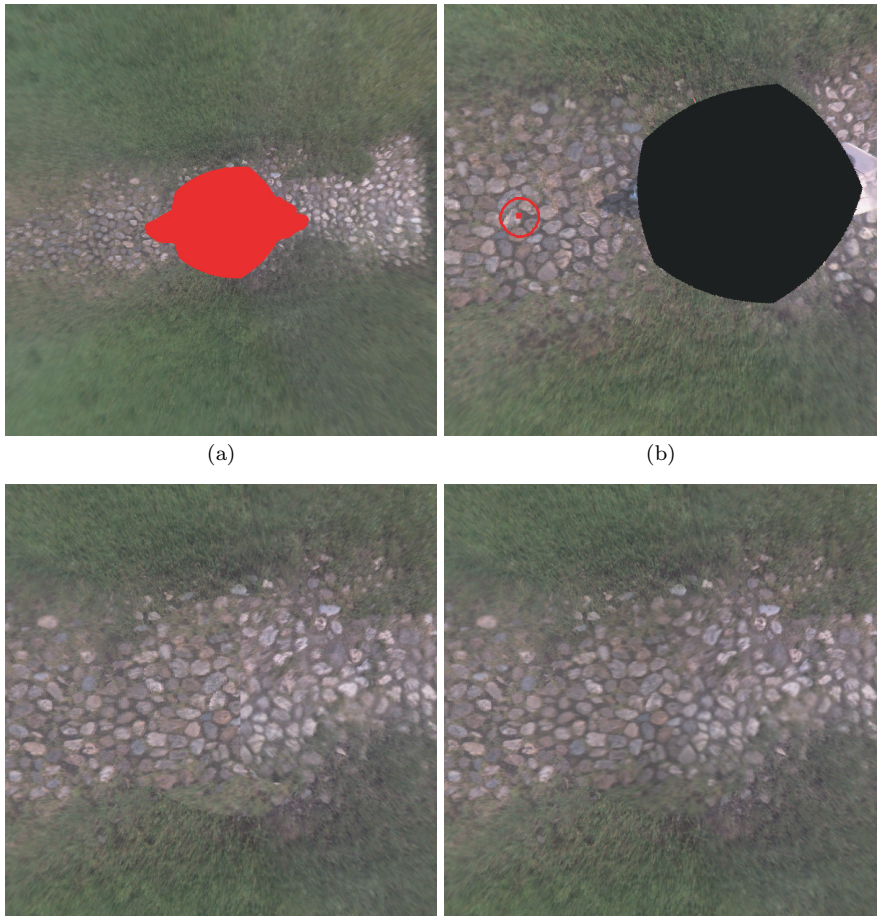
Next, a missing region in each projected image was completed. **Figures 11** and **12** show a variety of image results for the projected images of the 11th frame of scene (1) and the 71st frame of scene (2). Figures 11 (a) and 12 (a) show the target frames in which the missing region (shown in red) exists. Figures 11 (b) and 12 (b) show the central pixels of the data regions in the reference frames corresponding to pixel location $(600, 600)$ in the target frames. Figures 11 (c) and 12 (c) show the results of copying values of central pixels in data regions



(a) (b)

(c) (d)

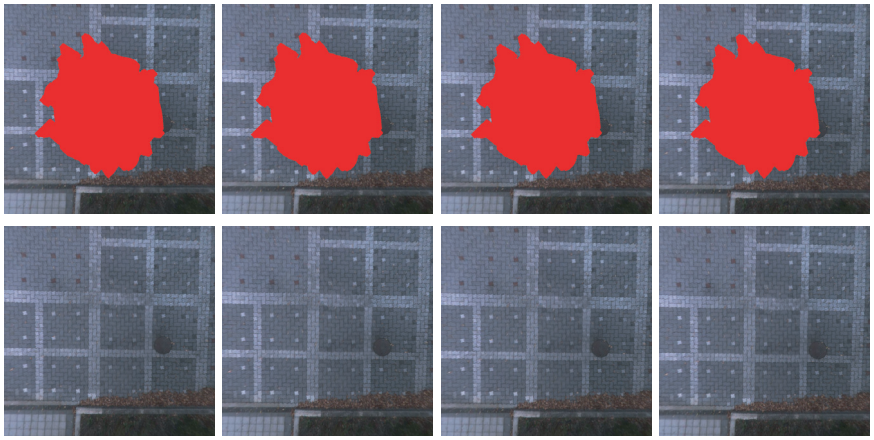
Fig. 11 Comparison of results for the 11th frame: Image (a) is the target image with a missing region; image (b) shows the center of the data region in the 67th frame corresponding to pixel $(600, 600)$ of the target image; image (c) is the resultant image produced by copying values of central pixels in data regions; and image (d) is the resultant image produced by the proposed method.



(a) (b)

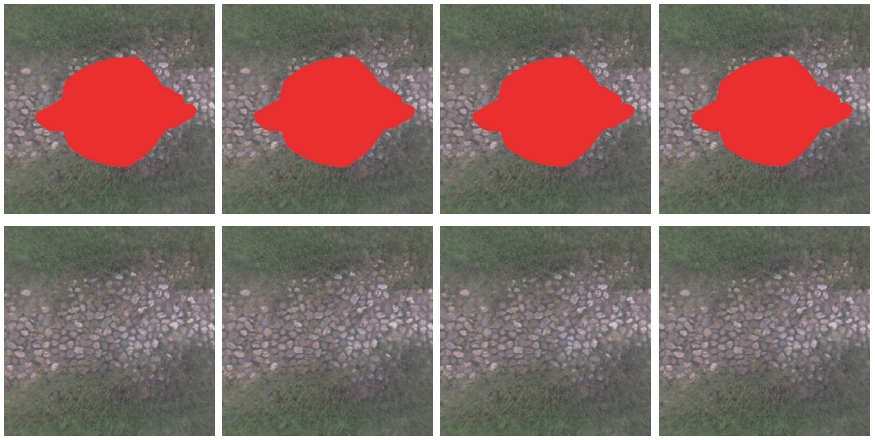
(c) (d)

Fig. 12 Comparison of results for the 71st frame: Image (a) is the target image with a missing region; image (b) shows the center of the data region in the 15th frame corresponding to pixel $(600, 600)$ of the target image; image (c) is the resultant image produced by copying values of central pixels in data regions; image (d) is the resultant image produced by the proposed method.



12th frame 13th frame 14th frame 15th frame

Fig. 13 Completion results for successive frames of scene (1).



72nd frame 73rd frame 74th frame 75th frame

Fig. 14 Completion results for successive frames of scene (2).

without the energy minimization process described above in Section 3.3. In Fig. 11 (c), the geometrical and optical disconnection of textures in the boundary of the missing region appears. We conclude that planes fitted to feature points contained some errors due to the ground not being absolutely planar. As for Fig. 12 (c), we also confirm the geometrical disconnection of textures in the middle of the missing region. In this scene, textures in the left side of the missing region are copied from earlier frames and those in the right side are copied from later frames. Therefore, the large disconnection appears. Conversely, Figs. 11 (d) and 12 (d) show resultant images in which textures are continuously connected on the boundary and middle of the missing region. As a result, natural textures are generated in the missing region. From these results, we confirm that the energy minimization process using pattern similarity considering brightness change of textures is effective for generating natural textures even if the ground is not absolutely planar.

Figures 13 and **14** show projected images of successive frames (the 12th to the 15th frames of scene (1) and the 72nd to the 75th frames of scene (2)). From these figures, the missing region in each frame of each scene was successfully completed. In addition, textures in the missing region change smoothly between successive frames and plausible videos are generated; however, as shown in **Fig. 15**, for scene (2), unnatural texture changes appeared from the 298th to the 301st frames. The completed textures are gradually distorted through these frames. We conclude that correspondences of pixels were inaccurate because the angle of the plane of each frame differed from the ground truth due to steps near the missing region. To overcome these problems in the future, we should use a more adaptive plane-fitting approach that considers the range for feature points. In addition, to obtain good results for more complex scenes, the detailed three-dimensional surface should be fitted to feature points.

We discuss the processing time. For scene (1), it took 722 seconds to complete the initial frame (i.e., determine data regions and minimize the energy function) and an average of 513 seconds to complete each subsequent frame. Similarly for scene (2), it took 776 seconds to complete the initial frame and an average of 523 seconds to complete the each subsequent frame. We observe a decrease in computational cost from the initial frame to subsequent frames, because completion results for a previous frame are reused.

4.3 Omnidirectional Telepresence without Invisible Areas

The effectiveness of the proposed method is fully realized by creating a pro-

298th frame 299th frame 300th frame 301st frame

Fig. 15 Results showing problems with our proposed method.

totype telepresence system using omnidirectional video in which missing regions are filled in with the completed images shown above in the previous section.

Top images of **Figs. 16** and **17** show omnidirectional panoramic images with a missing region; bottom images of these same figures show the omnidirectional panoramic images without invisible areas, as generated by projecting completed images (see Figs. 11 (d) and 12 (d)) onto the panoramic images. Note that these panoramic images are 2,048 × 1,024 images. **Figures 18** and **19** show example user views in the telepresence system using the above panoramic images. Comparing images with and without the missing regions, we confirm that realistic sensation is drastically increased by the proposed method; however, we also confirm that textures in the missing regions are slightly blurry. In the completion of the proposed method, the texture resolution for exemplars becomes small because the textures are captured from different viewpoints away from the camera position of the target frame. To increase the resolution of generated textures, we consider that a super-resolution method may be useful.

5. Conclusion

In this paper, we proposed a novel video completion pipeline for omnidirectional video. This pipeline compensated for the change in texture appearance and determined appropriate searching regions using three-dimensional geometric information for successful video completion.

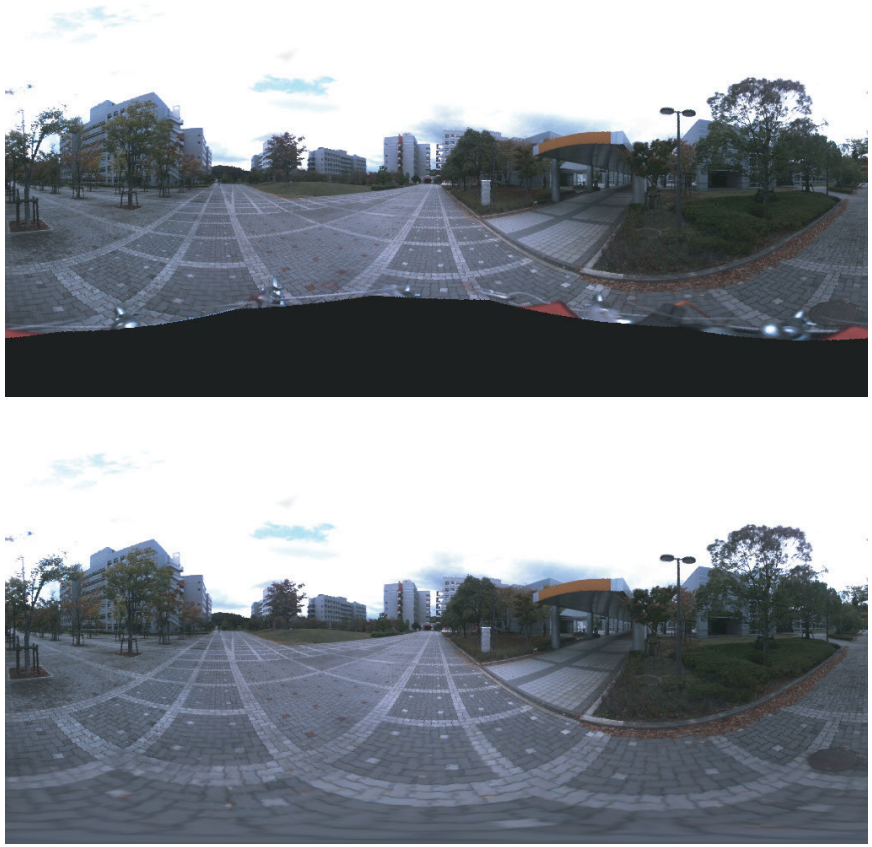


Fig. 16 Panoramic image of the 11th frame of scene (1) before and after video completion.

In experiments, missing regions in images projected on planes were successfully completed and the effectiveness of an energy minimization based on pattern similarity was demonstrated by comparing results of the proposed method with and

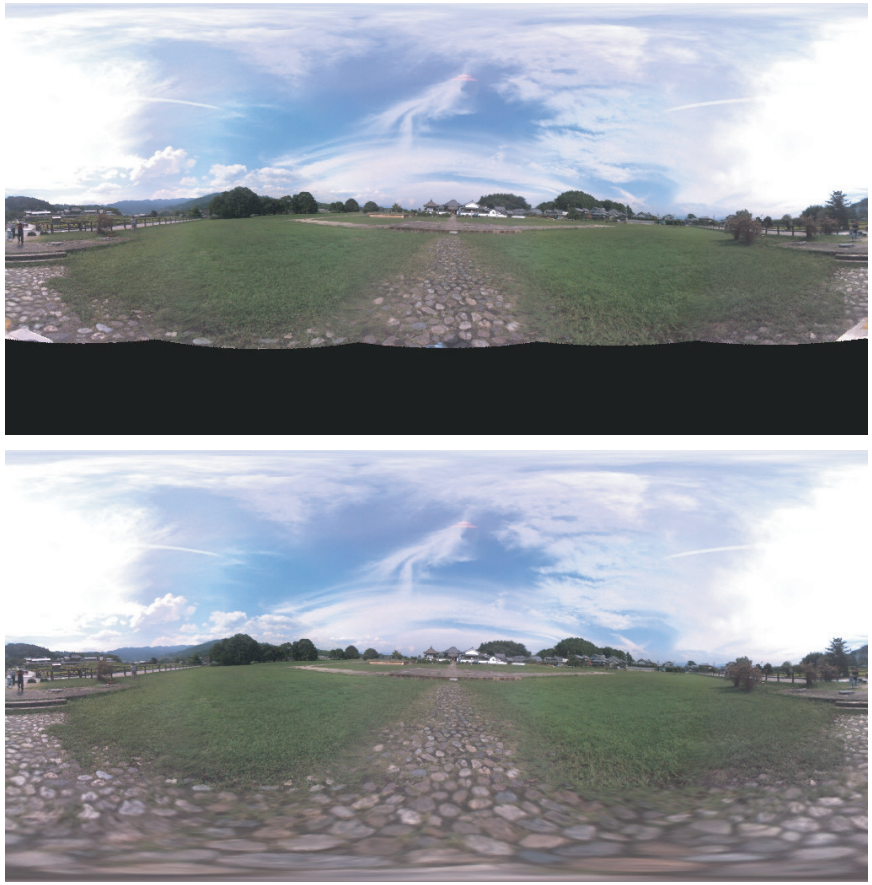


Fig. 17 Panoramic image of the 71st frame of scene (2) before and after video completion.

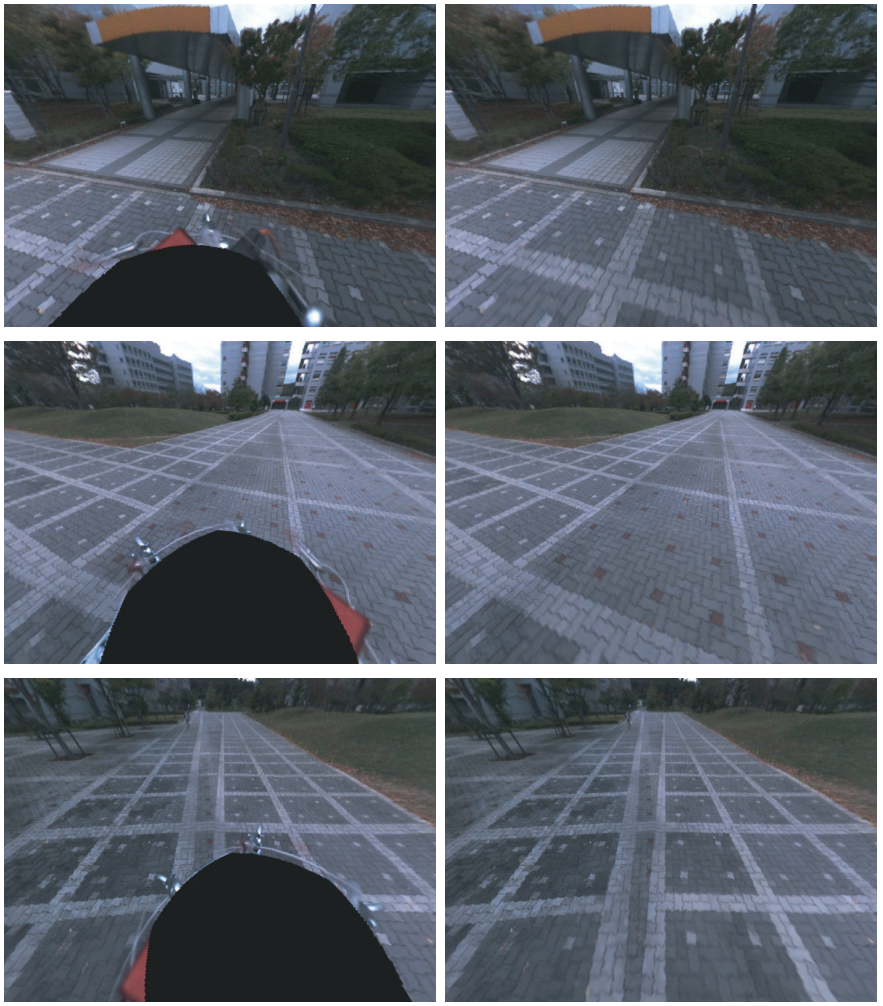


Fig. 18 Examples of looking around scene (1) using omnidirectional video with and without missing regions. Note that these are the 11th frames (top), 101st frames (middle) and 201st frames (bottom).

without an energy minimization process. In addition, we confirmed that textures in missing regions change smoothly between successive frames. Furthermore, omnidirectional telepresence without missing regions was achieved; we improved the realistic sensation in such an offline telepresence system by successfully filling in missing regions.

In the proposed method, we assume the ground around a missing region is

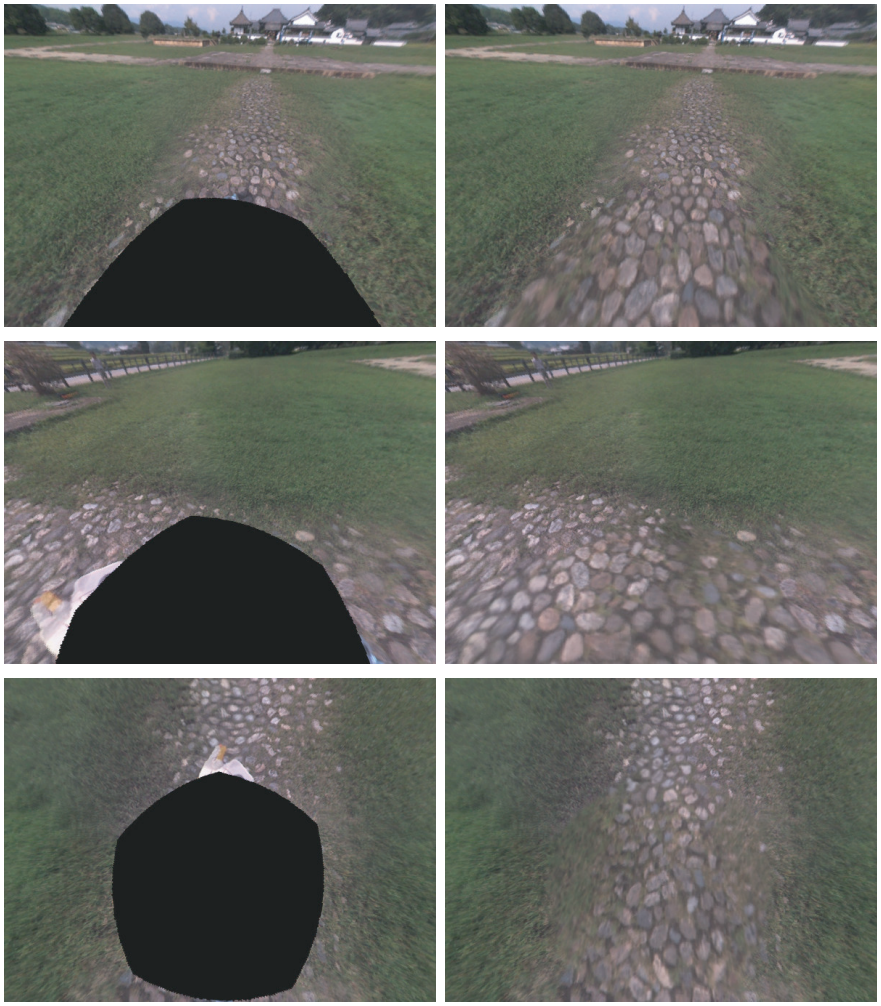


Fig. 19 Examples of looking around scene (2) using omnidirectional video with and without missing regions. Note that these are the 71st frame (top), 101st frames (middle) and 201st frames (bottom).

roughly planar. Although such an assumption is applicable to many scenes, a completion method that can be applied to a wider variety of shapes of missing regions, including stairs and rugged ground, should be developed. Furthermore, a method for increasing the resolution of textures in missing regions should be explored to reduce the blurriness of resultant images.

Acknowledgments This research was partially supported by the Ministry of Education, Culture, Sports, Science and Technology, Grant-in-Aid for Scientific Research (A), 19200016, and JSPS Fellows, 21-8045.

References

1) Bertalmio, M., Sapiro, G., Caselles, V. and Ballester, C.: Image Inpainting, *ACM Trans. on Graphics*, pp.417–424 (2000).
2) Bertalmio, M., Bertozzi, A. and Sapiro, G.: Navier-Stokes, Fluid Dynamics, and Image and Video Inpainting, *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Vol.1, pp.355–362 (2001).
3) Cheung, V., Frey, B. and Jojic, N.: Video epitomes, *Int. Journal of Computer Vision*, Vol.2, No.2, pp.141–152 (2008).
4) Criminisi, A., Pérez, P. and Toyama, K.: Region Filling and Object Removal by Exemplar-Based Image Inpainting, *IEEE Trans. on Image Processing*, Vol.13, No.9, pp.1200–1212 (2004).
5) Hori, M., Kanbara, M. and Yokoya, N.: Novel Stereoscopic View Generation by Image-Based Rendering Coordinated with Depth Information, *Proc. Scandinavian Conf. on Image Analysis*, pp.193–202 (2007).
6) Ikeda, S., Sato, T. and Yokoya, N.: Immersive Telepresence System with a Locomotion Interface Using High-resolution Omnidirectional Videos, *Proc. IAPR Conf. on Machine Vision Applications*, pp.602–605 (2005).
7) Jia, J., Tai, Y., Wu, T. and Tang, C.: Video Repairing under Variable Illumination Using Cyclic Motions, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol.28, No.5, pp.832–839 (2006).
8) Kawai, N., Sato, T. and Yokoya, N.: Image Inpainting Considering Brightness Change and Spatial Locality of Textures and Its Evaluation, *Proc. Pacific-Rim Symp. on Image and Video Technology*, pp.271–282 (2009).
9) Komodakis, N. and Tziritas, G.: Image Completion Using Global Optimization, *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp.442–452 (2006).
10) Litvin, A., Konrad, J. and Karl, W.: Probabilistic video stabilization using Kalman filtering and mosaicking, *Proc. SPIE Electronic Imaging*, Vol.5022, pp.663–674 (2003).
11) Matsushita, Y., Ofek, E., Ge, W., Tang, X. and Shum, H.: Full-Frame Video Stabilization with Motion Inpainting, *IEEE Trans. on Pattern Analysis and Machine*

Intelligence, Vol.28, No.7, pp.1150–1163 (2006).
12) Patwardhan, K., Sapiro, G. and Bertalmio, M.: Video Inpainting Under Constrained Camera Motion, *IEEE Trans. on Image Processing*, Vol.16, pp.545–553 (2007).
13) Point Grey Research Inc.: Ladybug. http://www.ptgrey.com/products/spherical.asp
14) Sato, T., Ikeda, S. and Yokoya, N.: Extrinsic Camera Parameter Recovery from Multiple Image Sequences Captured by an Omni-directional Multi-camera System, *Proc. European Conf. on Computer Vision*, Vol.2, pp.326–340 (2004).
15) Shen, Y., Lu, F., Cao, X. and Foroosh, H.: Video Completion for Perspective Camera Under Constrained Motion, *Proc. IEEE Int. Conf. on Pattern Recognition*, pp.63–66 (2006).
16) Wexler, Y., Shechtman, E. and Irani, M.: Space-Time Completion of Video, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol.29, No.3, pp.463–476 (2007).
17) Yamashita, A., Fukuchi, I., Kaneko, T. and Miura, T.: Removal of Adherent Noises from Image Sequences by Spatio-temporal Image Processing, *Proc. IEEE Int. Conf. on Robotics and Automation*, pp.2386–2391 (2008).
18) Zhang, Y., Xiao, J. and Shah, M.: Motion Layer Based Object Removal in Videos, *Proc. IEEE Workshops on Application of Computer Vision*, Vol.1, pp.516–521 (2005).

(Received January 29, 2010)
(Accepted August 5, 2010)
(Released November 15, 2010)

(Communicated by *Hajime Nagahara*)

Norihiko Kawai received his B.E. degree in informatics and mathematical science from Kyoto University in 2005. He received his M.E. and Ph.D. degrees in information science from Nara Institute of Science and Technology in 2007 and 2010, respectively. He is currently a research fellow of the Japan Society for the Promotion of Science. He is also a postdoctoral fellow at the University of California at Berkeley.

Kotaro Machikita received his B.E. degree in electrical and electronic engineering from Kobe University in 2006. He received his M.E. degree in information science from Nara Institute of Science and Technology in 2009. He has been working at SANYO Electric Co., Ltd. since 2009.

Tomokazu Sato received his B.E. degree in computer and system science from Osaka Prefecture University in 1999. He received his M.E. and Ph.D. degrees in information science from Nara Institute of Science and Technology in 2001 and 2003, respectively. He has been an assistant professor at Nara Institute of Science and Technology since 2003.

Naokazu Yokoya received his B.E., M.E., and Ph.D. degrees in information and computer science from Osaka University in 1974, 1976, and 1979, respectively. He joined Electrotechnical Laboratory (ETL) in 1979. He was a visiting professor at McGill University in 1986–1987 and has been a professor at Nara Institute of Science and Technology since 1992.