# Outdoor Omnidirectional Video Completion via Depth Estimation by Motion Analysis

Carlos Morales*, Shintaro Ono*, Yasuhide Okamoto*, Menandro Roxas*, Takeshi Oishi* and Katsushi Ikeuchi†
*The University of Tokyo, Japan
†Microsoft Research Asia, China

***Abstract*—Video completion aims to track, remove, and fill in unwanted regions (*holes*) of a video sequence. Holes have to be filled-in consistently to create a visually pleasant video output. Challenges arise when big holes propagate along several frames (*large spatiotemporal holes*) in outdoor videos with variant illumination and structured background. In those cases even forefront video completion approaches based on optical flow fail to complete the holes correctly as 3D information is required to keep the structure of the scene and a wider field of view is needed to handle the large spatiotemporal holes. To overcome these limitations, we propose a novel omnidirectional video completion framework based on depth estimation. First, we recover the depth of the scene from a pixel motion model constrained by known camera pose. The depth map is further improved by a structure-aware refinement. The refined depth map is then employed for color propagation into the holes. We perform a set of experiments to evaluate our approaches for preliminary depth recovery, depth refinement, and color propagation. Our results confirm that the proposed framework generates accurate preliminary depth maps, improves the depth quality maintaining the structure of the scene, and outperforms state-of-the-art optical-flow-based video completion approach in terms of accuracy and visual appeal.**

## I. Introduction

Omnidirectional video a.k.a. "360° video" has become mainstream over the recent years due to its important applications in the academia and the industry. Objects, people, or missing frame regions may be unwanted in outdoor captured videos. Video completion techniques have been applied to remove such undesired video portions (hereafter refered as *holes*) in a way that the synthesized video is visually pleasant to the human eye. Visual coherence between filled-in holes and the unsynthesized video regions require three consistencies. A geometric consistency to keep the structure of the scene, an appearance consistency in terms of color and texture, and a temporal consistency to coordinate motion of pixels. However, achieving visual coherence when big holes propagate along many frames is challenging, specially in illumination-variant scenes with structured backgrounds.

Several works on video completion are available in the literature. They can be classified into 1) perspective-view video inpainting, 2) perspective-view video completion, and 3) omnidirectional video completion.

1) *Perspective-view video inpainting*: Holes are filled-in extending image inpainting methods ( [1], [2], [3], [4], [5]) based on structure propagation. These completion techniques ( [6], [7]) can handle dynamic foregrounds but are limited to fill in small holes.
2) *Perspective-view video completion*: These techniques can fill in much larger holes via image completion methods ( [8], [9]) based on texture synthesis or optical flow. Jia et al. [10] filled in holes by merging source fragments with known parts of target fragments based on color similarities. Wexler et al. [11] used non-parametric sampling to fill in holes via global spatio-temporal optimization and 3D patches. Recently, Roxas et al. [12] proposed a spatio-temporally consistent method to fill in large spatiotemporal holes with compelling results. They utilized an iterative optimization approach that simultaneously solves for optical flow minimization and color propagation. Limitations of these techniques arise when dealing with holes in structured backgrounds.
3) *Omnidirectional video completion*: Large spatiotemporal holes are filled-in exploiting the wider field of view. Flores et al. [13] used perspective views acquired from two omnidirectional images to warp them via homography and do pixel replacement using probability maps. Their method can fill in big holes but fails to consistently complete backgrounds with non-planar regions. Kawai et al. [14] used 3D reconstruction from off-the-shelf Structure from Motion [15] and Multiple-View Stereo [16] to align frames and propagate known pixels into holes. Their method handles structured backgrounds but requires a highly accurate 3D model of the scene.

In this paper we do not focus in the hole tracking issue, rather we concentrate on the problem of filling in large spatiotemporal holes in outdoor scenes with static, structured backgrounds. For this purpose, we propose a novel omnidirectional video completion framework based on depth estimation from pixel motion analysis. First, pixel motion along multiple frames is modeled based on known camera pose. A preliminary depth map of the scene is recovered from the pixel motion model. The depth map is then refined preserving the structure of the scene. Finally, the refined depth is used for filling holes by color propagation.

The main contributions of this paper are three-fold: 1) A novel way of depth recovery from pixel motion analysis that accurately estimates the depth from a frame sequence with known camera pose. 2) A structure-aware refinement approach that significantly improves the quality of depth maps [17]. 3) A depth-based color propagation scheme that improves over the previous state of the art [12] in terms of visual coherence.

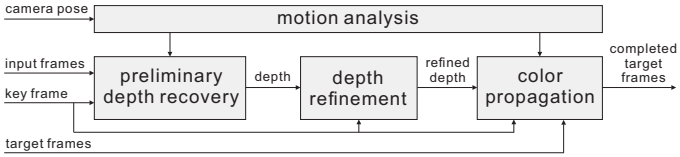



Fig. 1: Method overview.

## II. Method Overview

Figure 1 shows the overview of our video completion approach. Input data is the omnidirectional camera pose as well as the user-defined *target frames*, *input frames*, and *key frame*. Target frames are all the frames containing holes. We assume that the occluded region in the target frame can be observed from other camera positions. From this condition, the input frames are a set of frames where those occluded regions are visible. The key frame is the closest input frame to the target frames. The problem addressed in this work is to estimate the depth at the key frame using the input frames and then propagate the color information from the key frame into the holes using the estimated depth. For this purpose, our approach consists of the following four stages:

1) *Motion analysis*: Assuming a known camera pose, the pixel motion (*spatiotemporal pixel trajectory*) along the input frames is modeled so that it only depends of a time-invariant pixel depth. The time-invariant depth is defined as the distance from the scene to a *virtual camera* with fixed pose. The virtual camera is chosen depending of the camera pose model. We assume the camera pose can be modeled by two types of curves: circle or line. Then the virtual camera is placed at the imaginary center of curvature of the camera path in case the path is circular and is placed at the key frame pose in case the path is a line. This setting simplifies the parametric equations that describe the pixel motion from camera motion.
2) *Preliminary depth recovery*: The depth at the key frame is recovered by solving for an energy function that accounts for a weighted RGB color variation along the spatiotemporal pixel trajectory. Such weights handle frames with high energy and also consider the relevance of frames based on their temporal distance from the key frame.
3) *Depth refinement*: The recovered preliminary depth map is then refined considering structure preservation via a joint filter that leverages the benefits of an edge-aware enhanced intensity image used as guide.
4) *Color propagation*: The refined depth is used to propagate pixel colors along the spatiotemporal pixel trajectory from the key frame into the holes at target frames.

## III. Video Completion Framework

### A. Motion Analysis

To map a 3D point to a pixel point in an omnidirectional video frame, the camera projection is modeled using a unit sphere where the camera is located in the center $C$ of the sphere, see Fig. 2. For each camera position at time $t$={t1, t2, t3}, a 3D point $P$ in world coordinates X-Y-Z is projected onto the unit sphere at point $p_t$. Mapping $p_t$ to an equirectangular format will lead to a 2D pixel motion. Let denote the pixel position $p_t = (x_t, y_t)$ in image coordinates with $x \in [0; w-1]$ and $y \in [0; h-1]$, where $w$ and $h$ are the width and length in pixels, respectively, of the equirectangular frame. Thus, such 2D pixel motion can be modeled by

$$x_t = \begin{cases} -\frac{w-1}{2\pi}\phi_t, & \text{if } \phi_t < 0 \\ (w-1)\left(1 - \frac{\phi_t}{2\pi}\right), & \text{otherwise} \end{cases}, \qquad y_t = \frac{h-1}{\pi}\theta_t \tag{1}$$

where $\theta_t$ and $\phi_t$ are the zenith and azimuth angles, respectively, of Fig. 2.

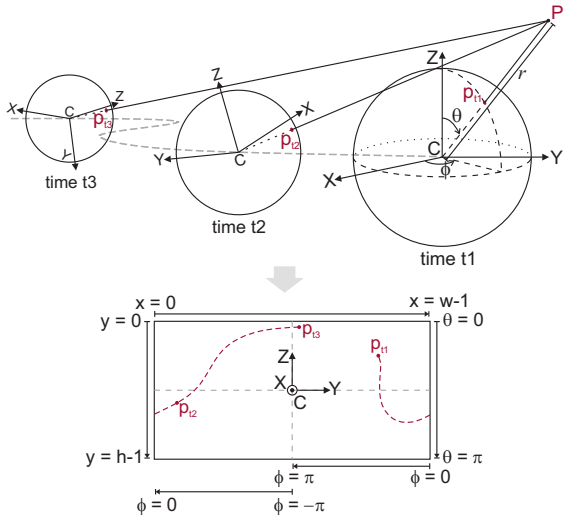



Fig. 2: Pixel motion from camera motion. A 3D point $P$ is projected to a moving camera at points $p_t$ for times $t$ ={t1, t2, t3}. Then the 2D pixel motion of $p_t$ is modeled in equirectangular format.

In a static scene the pixel motion in the omnidirectional frame can be calculated from the camera pose, the angles $\theta$ and $\phi$, and the depth $r$ at each time $t$. Modeling the pixel motion in terms of $r_t$ is inconvenient for video completion since the depth of a pixel would need to be modeled for each camera position. In this work, we propose to model the pixel position at time $t$ in terms of a time-invariant depth $s$ as

$$x_{t,s} = \begin{cases} -\frac{w-1}{2\pi}\phi(x_{t_k}, y_{t_k}, t-t_k, s), & \text{if } \phi < 0 \\ (w-1)\left(1 - \frac{\phi(x_{t_k}, y_{t_k}, t-t_k, s)}{2\pi}\right), & \text{otherwise} \end{cases}, \qquad y_{t,s} = \frac{h-1}{\pi}\theta(x_{t_k}, y_{t_k}, t-t_k, s) \tag{2}$$

where $t_k$ corresponds to the time counter for the key frame. The depth $s$ is measured from the 3D point $P$ to a fixed virtual camera position, as described in the previous section. This virtual relocation of the camera pose allows the simplification of the tedious parametric model of $\phi(\cdot)$ and $\theta(\cdot)$.

### B. Preliminary Depth Recovery

The goal in this stage is to recover a preliminary depth map $s$ of a key frame at $t_k$ from $N_{\text{IN}}$ input frames $\boldsymbol{I} = \{I_c^{(1)}, I_c^{(2)}, ..., I_c^{(k)}, ..., I_c^{(N_{\text{IN}})}\}$ corresponding to times $\boldsymbol{T} = \{T^{(1)}, T^{(2)}, ..., T^{(k)}, ..., T^{(N_{\text{IN}})}\}$, where $c \in \{r, g, b\}$ stands for the RGB channels. For this purpose, our approach combines (2) with the assumption that the color of a pixel $i$ along its spatiotemporal trajectory varies slightly. We estimate the pixel depth $s_i$ of the image region $\Omega$ that minimizes the color variation along its pixel trajectory by solving

$$\min_{s_i \in \Omega} \zeta_i \sum_{t \in \boldsymbol{T}} \tau_t E_{t,s_i}, \tag{3}$$

where $E_{t,s_i}$ is a color variation energy function, $\tau_t$ is a weighting factor that balances the color variation due to changing illumination by giving more weight to frames near the key frame, and $\zeta_i$ is a weighting factor that handles the effect of outliers. We define $E_{t,s_i}$, $\tau_t$, and $\zeta_i$ as

$$E_{t,s_i} = \sum_{c \in \{r,g,b\}} \Big( I_c^{(t)}(x_{t,s_i}, y_{t,s_i}) - MED(\boldsymbol{I}_{c,i}) \Big)^2, \tag{4}$$

$$\tau_t = 1 \Big/ \left( 1 + \left( \frac{T^{(k)} - t}{T^{(N_{\text{IN}})} - T^{(1)}} \right)^2 \right), \tag{5}$$

$$\zeta_i = 1 \Big/ \Big( 1 + \sum_{c \in \{r,g,b\}} \left( MAD(\boldsymbol{I}_{c,i}) \right)^2 \Big), \tag{6}$$

where $I_c^{(t)}(\cdot)$ stands for the pixel color at time $t$ and position explained in (2), and $\boldsymbol{I}_{c,i}$ is the array composed by the intensity values of pixel $i$ along its spatiotemporal trajectory. In this paper $MED(\cdot)$ and $MAD(\cdot)$ denote the median and the median absolute deviation, respectively, of a 1D array.

To estimate the preliminary depth map $s$ by solving (3) whilst avoiding unwanted local minima, we employ a discrete solution that runs a number of $N_s$ depth steps and evaluates which discrete depth value provides the global minimum.

### C. Depth Refinement

Due to the nature of our pixel-based depth recovery approach of (3), the preliminary depth map $s$ at the key frame needs a posterior refinement to correct high depth variations (scattered holes), wrong contours, and subtle depth variations where the depth map should be smooth. To handle those issues, we propose the following three-phase depth refinement approach with structure preservation.

1) The scattered holes in $s$ are detected by

$$\bar{s}_i = |s_i - MED(s)| / MAD(s), \tag{7}$$

$$s_i = \begin{cases} \text{hole}, & \text{if } s_i < \epsilon_1 \text{ and } \bar{s}_i < \epsilon_2 \\ s_i, & \text{otherwise} \end{cases}, \tag{8}$$

where $\epsilon_1$ and $\epsilon_2$ are threshold constants. A refined depth map $s$ is then obtained inpainting the holes by Partial Differential Equations (PDEs).

2) The RGB information of the key frame $I_c$ is enhanced to remove high frequency noise, reinforce edges, and alleviate coarse textures. The processed key frame $\hat{I}_c$ is the solution to the following optimization problem for edge-aware smoothing:

$$\min_{\hat{I}_c} \sum_{i \in \Omega} (\hat{I}_{c,i} - I_{c,i})^2 + \lambda \left( \alpha_{c,i} |\nabla_x \hat{I}_c|_i^2 + \beta_{c,i} |\nabla_y \hat{I}_c|_i^2 \right), \tag{9}$$

where $\Omega$ is the 2D region of all pixels in $I_c$, $\lambda$ is a trade-off factor, and $\nabla \cdot$ denotes the gradient operator. We define the smoothness factors $\alpha_{c,i}$ and $\beta_{c,i}$ as

$$\begin{aligned} \alpha_{c,i} &= \left( \frac{|\nabla_x \hat{I}_r|_i^2}{3\sigma_r^2/\sigma_c^2} + \frac{|\nabla_x \hat{I}_g|_i^2}{3\sigma_g^2/\sigma_c^2} + \frac{|\nabla_x \hat{I}_b|_i^2}{3\sigma_b^2/\sigma_c^2} \right)^{-1} \\ \beta_{c,i} &= \left( \frac{|\nabla_y \hat{I}_r|_i^2}{3\sigma_r^2/\sigma_c^2} + \frac{|\nabla_y \hat{I}_g|_i^2}{3\sigma_g^2/\sigma_c^2} + \frac{|\nabla_y \hat{I}_b|_i^2}{3\sigma_b^2/\sigma_c^2} \right)^{-1}, \end{aligned} \tag{10}$$

where $\sigma_c$ stands for the standard deviation of the $c$-channel of $I_c$. Equation (9) is computed solving the linear system under the Weighted Least Squares (WLS) framework as in [18].

3) Wrong contours and subtle depth variations in $s$ are corrected by a structure-aware filter. A final refined depth map $\hat{s}$ is obtained using $\hat{I}_c$ as guide to filter $s$ via a Joint Weighted Median Filter (JWMF). Following a general JWMF framework, for $j = \{1, 2, 3, ..., n\}$ ordered depth values $s_j^{(i)}$ belonging to the window $\Omega_i$ centered at pixel $i$, the corrected pixel depth is calculated as

$$\hat{s}_i = s_{\min\ k} \text{ s.t. } \sum_{j=1}^{k} w_{ij} \geq \frac{1}{2} \sum_{j=1}^{n} w_{ij}, \tag{11}$$

where $w_{ij}$ are positive weights. We use a Gaussian weight given by $w_{ij} = \exp\big( \| \hat{I}_i - \hat{I}_j \|^2 / 2\sigma^2 \big)$, where $\sigma$ is a filtering factor. In this work the computation of (11) is sped up using the framework proposed by [19].

### D. Color Propagation

Given the refined depth map $\hat{s}^{(k)}$ of a key frame $k$, the spatiotemporal trajectory of a pixel $i$ is calculated using (2). Then the color of pixel $i$ at $I_c^{(k)}$ is propagated to the holes in target frames $\ell$ with RGB intensities $\boldsymbol{J} = \{J_c^{(1)}, J_c^{(2)}, ..., J_c^{(\ell)}, ..., J_c^{(N_{\text{target}})}\}$ by following its pixel trajectory. The resulting completed frames $J_c^{(k \to \ell)}$ suffer from three main issues. Some regions of $J_c^{(k \to \ell)}$ may remain uncompleted (remaining holes) due to the nature of forward color propagation of (2) and the occlusion issues of large spatiotemporal videos. Second, propagated pixels in $J_c^{(k \to \ell)}$ will be wrongly placed when $\hat{s}^{(k)}$ is not reliable. Last, color inconsistencies around the hole boundaries of $J_c^{(k \to \ell)}$ will appear due to the variant illumination in outdoor scenes. To handle those issues, we propose the following depth-based color propagation scheme.

1) $J_c^{(k \to \ell)}$ is obtained propagating $I_c^{(k)}$ according to (2).
2) In order to detect the remaining holes and the wrongly placed pixels at $J_c^{(k \to \ell)}$, the depth $\hat{s}^{(k \to \ell)}$ at the target frame is obtained propagating $\hat{s}^{(k)}$ according to (2).

**Algorithm 1** The proposed framework for video completion

**Input:** Camera pose, target frames $\boldsymbol{J} = \{J_c^{(1)} ... J_c^{(N_{\text{target}})}\}$, input frames $\boldsymbol{I} = \{I_c^{(1)} ... I_c^{(N_{\text{IN}})}\}$, and key frame $k$

**Preliminary depth map recovery**:

**for** all pixel $i \in I_c^{(k)}$ **do**
  $s_i^{(k)} \leftarrow$ depth that minimizes color variation of $i$ along $\boldsymbol{I}$ according to (3)
**end for**

**Depth map refinement**:

$s^{(k)} \leftarrow$ PDE inpainting of scattered holes at $s^{(k)}$ detected using (8)
$\hat{I}_c^{(k)} \leftarrow$ structure-aware enhancement of $I_c^{(k)}$ using (9)
$\hat{s}^{(k)} \leftarrow$ filtering of $s^{(k)}$ guided by $\hat{I}_c^{(k)}$ according to (11)

**Color propagation**:

**for** $\ell = 1$ to $N_{target}$ **do**
  $J_c^{(k\to\ell)} \leftarrow$ propagation of $I_c^{(k)}$ to target frame $\ell$ using $\hat{s}^{(k)}$ according to (2)
  $\hat{s}^{(k\to\ell)} \leftarrow$ propagation of $\hat{s}^{(k)}$ to $\ell$ using (2)
  $J_c^{(k\to\ell)} \leftarrow$ PDE inpainting of missing and wrongly propagated pixels at $J_c^{(k\to\ell)}$ using $\hat{s}^{(k\to\ell)}$, (12) and (14)
  $J_c^{(k\to\ell)} \leftarrow$ Poisson blending of $J_c^{(k\to\ell)}$ with $J_c^{(\ell)}$
**end for**

**Output:** Completed frames $\boldsymbol{J'} = \{J_c^{(k\to1)} ... J_c^{(k\to N_{target})}\}$

3) Holes at $J_c^{(k\to\ell)}$ are caught as

$$J_{c,i}^{(k\to\ell)} = \begin{cases} \text{hole}, & \text{if } \hat{s}_i^{(k\to\ell)} \text{ is uncompleted} \\ J_{c,i}^{(k\to\ell)}, & \text{otherwise} \end{cases}, \quad (12)$$

and a processed $J_c^{(k\to\ell)}$ is then obtained inpainting the detected holes by PDEs.

4) The wrongly placed pixels at $J_c^{(k\to\ell)}$ due to the inaccuracy of $\hat{s}^{(k)}$ are detected as

$$\check{s}_i^{(k\to\ell)} = |\hat{s}_i^{(k\to\ell)} - MED_{2D}(\hat{s}^{(k\to\ell)})|, \quad (13)$$

$$J_{c,i}^{(k\to\ell)} = \begin{cases} \text{hole}, & \text{if } \hat{s}_i^{(k\to\ell)} > \epsilon_3 \\ \text{hole}, & \text{if } \check{s}_i^{(k\to\ell)} > \epsilon_4 \\ J_{c,i}^{(k\to\ell)}, & \text{otherwise} \end{cases}, \quad (14)$$

where $MED_{2D}(\cdot)$ is a 2D median filter, and $\epsilon_3$ and $\epsilon_4$ are threshold constants. A refined $J_c^{(k\to\ell)}$ is then obtained inpainting the detected holes by PDEs.

5) Finally, the new $J_c^{(k\to\ell)}$ is blended to the hole region of $J_c^{(\ell)}$ using Poisson Blending, thus producing a completed frame without color inconsistency issues.

The proposed color propagation approach is then applied to all target frames, generating the final result $\boldsymbol{J'} = \{J_c^{(k\to1)}, ..., J_c^{(k\to N_{target})}\}$. The complete video completion framework is explained in Algorithm 1.

## IV. Experimental Results

We performed quantitative and qualitative evaluations of our video completion framework. For all the experiments, the implementation was done in MatLab on a personal computer (OS: Windows 7; CPU: Corei7 2.93GHz; RAM: 16GB). All intensity and depth images were normalized from 0 to 1. The constants were $N_s = 100$, $\epsilon_1 = 0.05$, $\epsilon_2 = 3$, $\epsilon_3 = 0.9$, $\epsilon_4 = 0.05$, $\lambda = 0.01$, $n = 121$, and $\sigma = 15$. In this section the subscript $GT$ refers to ground truth.

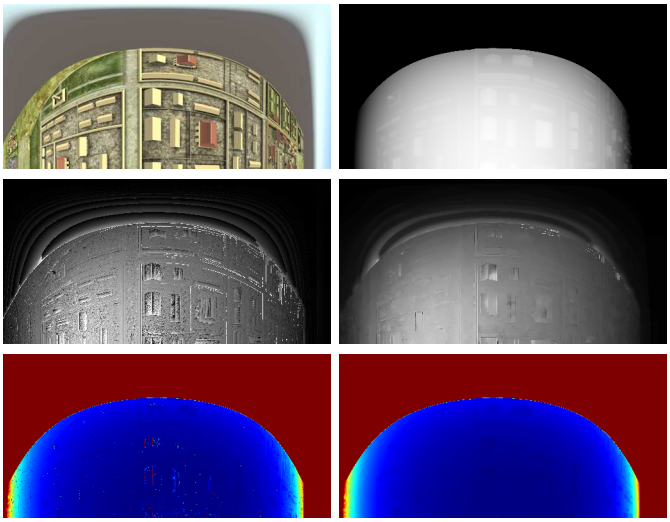


Fig. 3: Evaluation of our overall depth estimation approach with synthesized scenes. Top row: Key frame (top-right part of omnidirectional image format) and ground truth depth map. Middle row: our preliminary and refined depth maps. Bottom row: *jet* color map of the error (blue for $0\%$ and red for $100\%$) between the ground truth and the preliminary and refined depth maps.

### A. Evaluation of overall depth estimation approach

Our approach for depth estimation was evaluated using synthesized scenes. We tested our method with a camera motion composed by a translation of 0.43 meters per frame and a rotation of 1 degree per frame. We used $N_{\text{IN}} = 9$ input frames $\boldsymbol{I} = \{I_c^{(1)}, I_c^{(2)}, ..., I_c^{(k=9)}\}$ for the computation. The results of the evaluation are shown in Fig. 3. The error measure for the preliminary recovered depth map $s^{(k)}$ and the final refined depth map $\hat{s}^{(k)}$ are $|s^{(k)} - s_{GT}^{(k)}|/s_{GT}^{(k)}$ and $|\hat{s}^{(k)} - s_{GT}^{(k)}|/s_{GT}^{(k)}$, respectively. We found that the proposed preliminary depth recovery approach and the refinement approach performed accurately. We measured the root-mean-square error (RMSE) in the depth map region where the ground truth was accurate, that is, all the depth map except the sky and gray area (big red area in the colormap visualization). The RMSE of $s^{(k)}$ and $\hat{s}^{(k)}$ using $s_{GT}^{(k)}$ as reference was 0.1079 and 0.0978, respectively.

### B. Evaluation of depth refinement approach

We evaluated our refinement approach using depth maps obtained by algorithms of [17]. We applied our refinement approach to 16 algorithms of each image set *Tsukuba*, *Venus*, *Teddy*, and *Cones*. Algorithms were ordered for each image

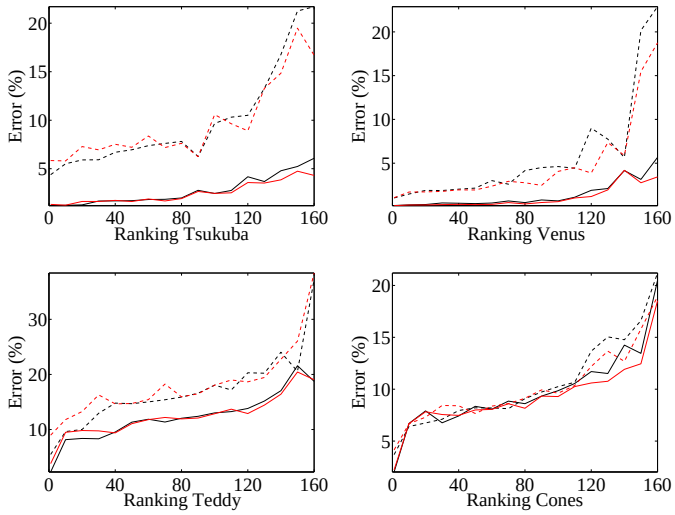



Fig. 4: Evaluation of our depth refinement approach on [17]. Algorithms are ranked based on their error for each image set. The error is the percent of bad matching pixels on *all* (solid line) and *disc* (dashed line). Black lines and red lines correspond to the error before and after refinement.

set by ranking from 1 to 160. The results of the evaluation are shown in Fig. 4. The error used for the ranking assessment was the percent of bad matching pixels (for absolute error greater than $1\%$) on all pixels in the image (*all*) and the visible pixels near the occluded regions (*disc*). Our best results for each image set are illustrated in Fig. 5. We found that the proposed method improved the accuracy of the depth maps for most of the evaluated algorithms. The edges of the depth maps were successfully corrected and aligned with the corresponding ground truth depth map. The most significant improvements were obtained for low-ranked algorithms that need more refinement, while slight degeneration was produced for top-ranked algorithms due to over refinement.

### C. Evaluation of video completion approach

We evaluated our proposed approach on real outdoor scenes captured by a Ladybug 3 camera. The 15 minute long video was down sampled to 3 FPS for convenience. The camera motion was composed by a translation of 0.044 meters per frame and a rotation of 0.05 degrees per frame. We used $N_{\text{IN}} = 10$ input frames $\boldsymbol{I} = \{I_c^{(k=140)}, I_c^{(160)}, ..., I_c^{(320)}\}$ for the computation. The original hole-free target frames $\boldsymbol{J}_{GT} = \{J_{c,GT}^{(0)}, J_{c,GT}^{(1)}, ..., J_{c,GT}^{(139)}\}$ were used as ground truth. Large spatiotemporal synthetic holes that cover almost $25\%$ of the full omnidirectional frame were introduced to the original target frames, resulting in frames $\boldsymbol{J} = \{J_c^{(0)}, J_c^{(1)}, ..., J_c^{(\ell)}, ..., J_c^{(139)}\}$. The goal of the experiment was to fill in the holes of $\boldsymbol{J}$. The estimated depth before and after refinement is shown in Fig. 6. The completed results are shown in Fig. 7. The proposed approach generated compelling completed outputs with visual coherence. We compared our results with the state-of-the-art optical-flow based video completion method of [12]. We measured the absolute error as $\sum_{c\in\{r,g,b\}} |J_{c4}^{(\ell)} - J_{c,GT}^{(\ell)}|/3$ and found that the proposed approach provides accurate structure-aware completed results even when holes span around 45 s from the key frame. In contrast, results obtained by [12] failed to keep the structure of the scene inside the challenging large spatiotemporal holes of the experiment.

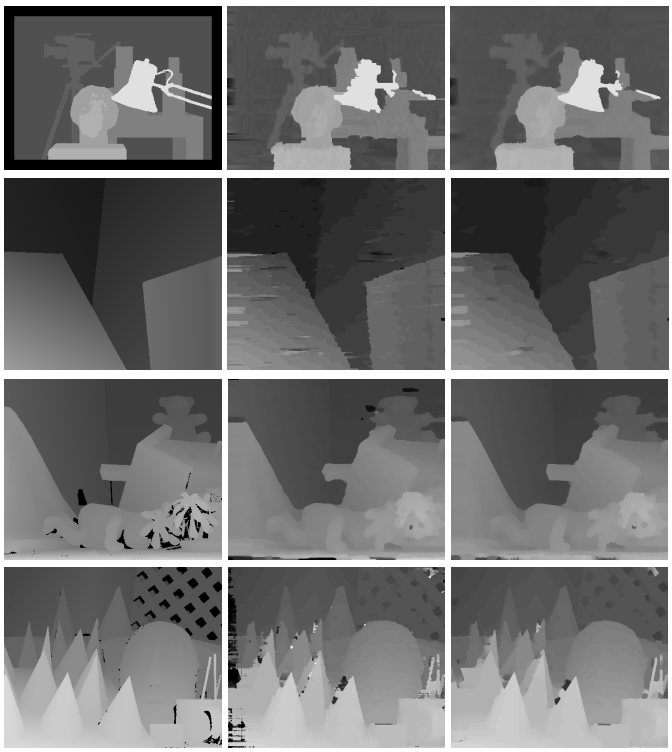



Fig. 5: Depth refinement on algorithms of [17]. From top to bottom: (a) Ground truth depth map of Tsukuba, Venus, Teddy, and Cones; (b) Results of SNCC, DPVI, RTCensus, and SGMDDW; (c) Our corresponding refinements.

## V. Conclusions

We have proposed a novel omnidirectional video completion framework for filling in large spatiotemporal holes with structured background. A preliminary depth map of the scene at a key frame is effectively recovered for one fixed virtual camera position via pixel motion analysis of frame sequences with known camera pose. The depth map is further refined accurately preserving the structure of the scene via a joint edge-aware smoothing filter guided with an enhanced intensity image. Filling in the holes of target frames is successfully achieved using a depth-based color propagation scheme. Qualitative and quantitative evaluations on synthesized and real data show that our framework accomplish superior performance for depth recovery and refinement as well as completed results with consistent visual coherence that significantly preserve the structure of the scenes inside the holes.

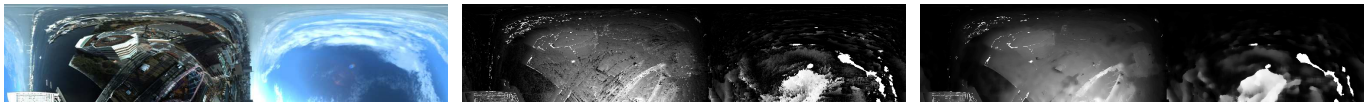
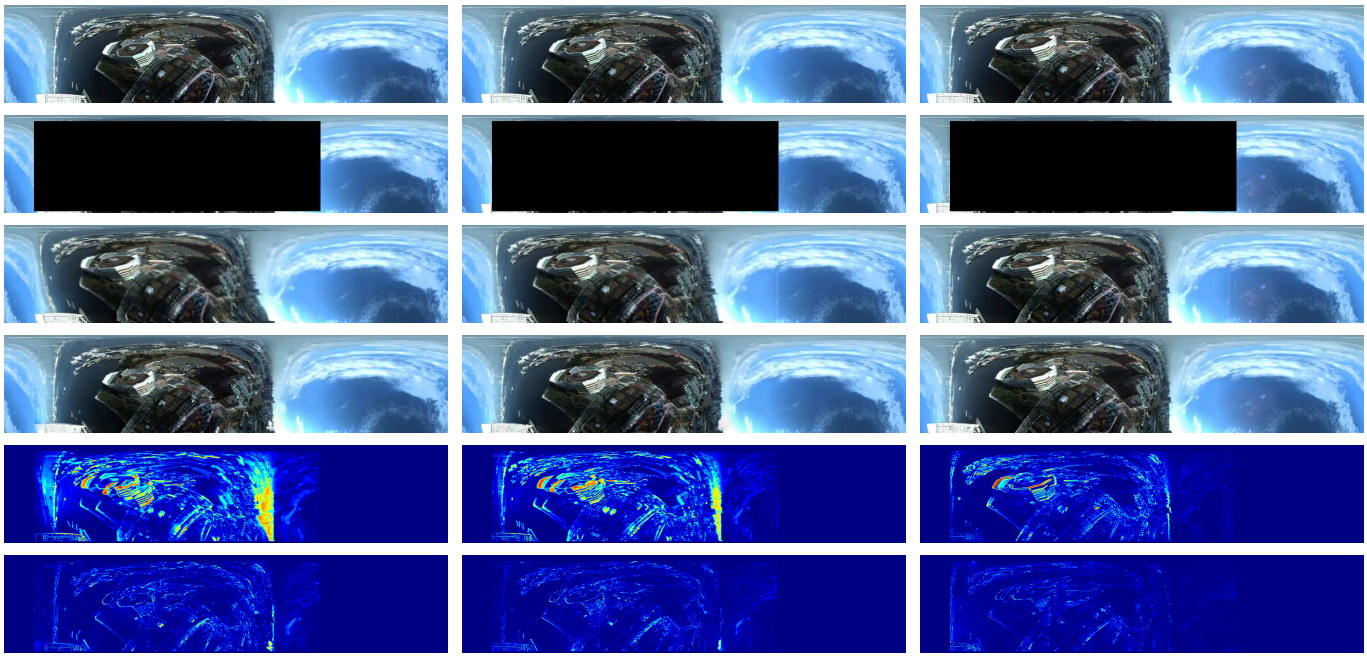


Fig. 6: Depth estimation in real scene. The images are the top half of a full equirectangular format. From left to right: RGB of key frame $k = 140$ and our preliminary and refined depth maps.



(a) (b) (c)

Fig. 7: Video completion results. Target frames for (a) $\ell = 0$, (b) $\ell = 60$, and (c) $\ell = 120$. From top to bottom: ground truth, synthetic holes, results of [12], our results, error of [12] with *jet* colormap (blue for $0\%$ and red for $100\%$), and our error.


## ACKNOWLEDGMENT

This work was, in part, supported by JSPS KAKENHI Grant Number 16H05864.


## REFERENCES


[1] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting," *Proc. ACM Siggraph*, pp. 414–424, 2000.

[2] A. Levin, A. Zomet, and Y. Weiss, "Learning how to inpaint from global image statistics," in *ICCV*, pp. 305–312, 2003.

[3] T. F. Chan and J. Shen, "Variational image inpainting," DTIC Document, Tech. Rep., 2005.

[4] X. Shao, Z. Liu, and H. Li, "An image inpainting approach based on the poisson equation," in *DIAL*, pp. 5–pp, 2006.

[5] J. A. Dobrosotskaya and A. L. Bertozzi, "A wavelet-laplace variational technique for image deconvolution and inpainting," in *TIP*, vol. 17, no. 5, pp. 657–663, 2008.

[6] K. A. Patwardhan, G. Sapiro, and M. Bertalmio, "Video inpainting of occluding and occluded objects," in *ICIP*, vol. 2, pp. II–69, 2005.

[7] S. Cheung, J. Zhao, and M. V. Venkatesh, "Efficient object-based video inpainting," in *ICIP*, pp. 705–708, 2006.

[8] I. Drori, D. Cohen-Or, and H. Yeshurun, "Fragment-based image completion," in *TOG*, vol. 22, no. 3, pp. 303–312, 2003.

[9] A. Criminisi, P. Pérez, and K. Toyama, "Region filling and object removal by exemplar-based image inpainting," in *TIP*, vol. 13, no. 9, pp. 1200–1212, 2004.

[10] Y.-T. Jia, S.-M. Hu, and R. R. Martin, "Video completion using tracking and fragment merging," in *The Visual Computer*, vol. 21, no. 8-10, pp. 601–610, 2005.

[11] Y. Wexler, E. Shechtman, and M. Irani, "Space-time completion of video," in *PAMI*, vol. 29, no. 3, pp. 463–476, 2007.

[12] M. Roxas, T. Shiratori, and K. Ikeuchi, "Video completion via spatio-temporally consistent motion inpainting," in *IMT*, vol. 9, no. 4, pp. 500–504, 2014.

[13] A. Flores and S. Belongie, "Removing pedestrians from google street view images," in *CVPRW*, pp. 53–58, 2010.

[14] N. Kawai, N. Inoue, T. Sato, F. Okura, Y. Nakashima, and N. Yokoya, "Background estimation for a single omnidirectional image sequence captured with a moving camera," in *IMT*, vol. 9, no. 3, pp. 361–365, 2014.

[15] C. Wu, "Visualsfm: A visual structure from motion system," 2011.

[16] M. Jancosek and T. Pajdla, "Multi-view reconstruction preserving weakly-supported surfaces," in *CVPR*, pp. 3121–3128, 2011.

[17] D. Scharstein and R. Szeliski, "Middlebury stereo evaluation," *Middlebury stereo evaluation-version 2*, 2008.

[18] Z. Farbman, R. Fattal, D. Lischinski, and R. Szeliski, "Edge-preserving decompositions for multi-scale tone and detail manipulation," in *TOG*, vol. 27, no. 3, p. 67, 2008.

[19] Q. Zhang, L. Xu, and J. Jia, "100+ times faster weighted median filter (wmf)," in *CVPR*, pp. 2830–2837, 2014.