16th International Conference on Machine Vision Applications (MVA)
National Olympics Memorial Youth Center, Tokyo, Japan, May 27-31, 2019.

Bullet-time image generation without 3-D

Yasuyuki SUGAYA
Toyohashi University of Technology
Toyohashi, Aichi 441-8580 Japan
sugaya@iim.cs.tut.ac.jp

Keita OHSEKI
Toyohashi University of Technology
Toyohashi, Aichi 441-8580 Japan
ohseki@iim.cs.tut.ac.jp

Abstract

We propose a new method for generating bullet-time images without any 3-D information. In a general way, we estimate 3-D camera pose and a 3-D position of a target point, then compute a homography that moves the target point to the image center. In this work, we re-parameterize the homography without 3-D information of the camera and the target point. We also estimate the homography by using an image matching technique. We compare the proposed method with an existing method that uses 3-D information and show the effectiveness of our method.

1 Introduction

A bullet-time is a movie effect that a camera moves around an object in the time-stopping world. Recently, the bullet-time is well used for sports scenes in TVs. In order to realize bullet-time effects, we need to locate multiple cameras so that all their optical axes pass through the same focusing point. However, it is very difficult to satisfy this condition strictly. Therefore, image translation methods are proposed[7][1][3].

These methods compute a homography that translates an input image so that a focusing point moves to the image center by using the camera internal and external parameters. If we do not know camera parameters, we need to estimate them from feature point correspondences between images. For camera parameter estimation and 3-D reconstruction from images, recently, structure from motion techniques are actively studied[4][5]. However, estimating camera parameters from images robustly is very difficult.

In this paper, we propose a new method for generating bullet-time images without any 3-D information. We assume that we know the camera internal parameter, for example the focal length, a priori. We introduce an image rotation angle, a scale parameter, and an image coordinates of the focusing point as parameters. Then, we re-parameterize the homography by them and estimate it by using an image matching technique. By using this method, we have possibility to robustly generate bullet-time images of the target scenes for which we fail 3-D reconstruction and camera pose estimation.

2 Key idea

2.1 Parameters for homography

For generating bullet-time images, Tomiyama el al.[7] and Sakamoto el al.[3] compute the following parameters by estimating camera parameters:

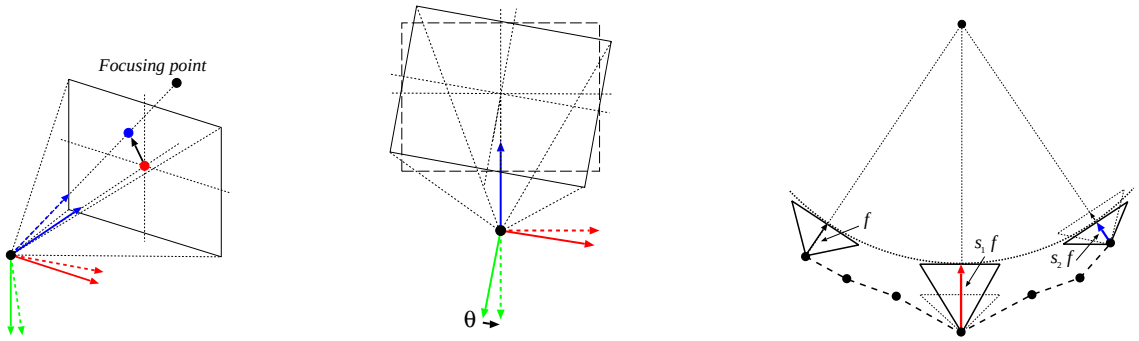
1. 3-D coordinates of a focusing point,
2. Camera rotation around the optical axis,
3. Scale parameter of a camera focal length.

We need to know the 3-D coordinates of the focusing point to virtually rotate the camera so that the focusing point moves to the principal point. If we know the relative motion of the two cameras we can compute it by triangulation. We use the scale parameter for aligning the scene scale and compute it based on the distance between the camera and the focusing point. The rotation parameter is used so that the vertical axis of the camera is perpendicular to the ground plane of the scene.

2.2 Parameter estimation without 3-D

We virtually estimate the above parameters without using 3-D camera information. We summarize the idea for estimating them as follows:

1. Position of the focusing point(Fig. 1(a))
   The 3-D position of the focusing point is used for computing the direction from a camera to it. Therefore, if we detect the projected image coordinates of it, we can compute the direction from the camera to the focusing point. In this paper, we select one image among input images and manually specify the image coordinates of a focusing point on it. Then, for the remaining images, we estimate the focusing point by using our proposed method.
2. Camera rotation around the optical axis(Fig. 1(b))
3. Scaling parameter(Fig. 1(c))
   The effect of these two parameters appears on the image transformed by a homography. Therefore, we select a template image among the input images and estimate these parameters by comparing it with the image transformed by them.



(a) (b) (c)

Figure 1. Parameters for homography. (a) Direction to the focusing point. (b) Rotation around the optical axis. (c) Scaling for aligning the image scales.

3 Homography re-parameterization

3.1 Homography for bullet-time

Let f be the focal length of a camera and (c_x, c_y) the image coordinates of the principal point. Then, the internal parameter matrix $\boldsymbol{K}$ is written by

$$\boldsymbol{K} = \begin{pmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{pmatrix}. \tag{1}$$

Using a rotation matrix $\boldsymbol{R}$ that rotates the camera so as to locate a focusing point at the principal point, the homography $\boldsymbol{H}$ is obtained by

$$\boldsymbol{H} = \boldsymbol{K}\boldsymbol{R}\boldsymbol{K}^{-1}. \tag{2}$$

If we denote an (ij) element of the homography $\boldsymbol{H}$ as h_{ij}, the relationship between (x, y) and (x', y') is written by

$$x' = \frac{h_{11}x + h_{12}y + h_{13}}{h_{31}x + h_{32}y + h_{33}}, y' = \frac{h_{21}x + h_{22}y + h_{23}}{h_{31}x + h_{32}y + h_{33}}, \tag{3}$$

where the point (x, y) is the image coordinates of the input image, and the point (x', y') is its corresponding image coordinates transformed by $\boldsymbol{H}$[2][6]. If we use the homogeneous coordinates for the image points $\boldsymbol{x} = (x, y, 1)^\top$ and $\boldsymbol{x'} = (x', y', 1)^\top$, Eq. (3) can be written by

$$\boldsymbol{x}' = Z[\boldsymbol{H}\boldsymbol{x}], \tag{4}$$

where $Z[\cdot]$ denotes a normalization operator ($Z[\boldsymbol{a}] \equiv \boldsymbol{a}/a_3$, a_3 is the third element of the vector $\boldsymbol{a}$.).

3.2 Rotation matrix

Let (g_x, g_y) be an image coordinates of a 3-D focusing point $\boldsymbol{G}$. Then, the unit vector $\boldsymbol{e}_z$ that passes through the camera center and the point $\boldsymbol{G}$ is given by

$$\boldsymbol{e}_z = N[\begin{pmatrix} g_x - c_x \\ g_y - c_y \\ f \end{pmatrix}], \tag{5}$$

where $N[\cdot]$ denotes a normalization operator ($N[\boldsymbol{a}] \equiv \boldsymbol{a}/\|\boldsymbol{a}\|$).

Now we consider a camera rotation around the optical axis. We define a camera coordinate system so that we locate its origin at the camera center and set the Z axis along the optical axis and the Y axis to the vertical direction as shown in Fig. 1.

If we rotate the camera coordinate system around the Z axis by θ, the rotated Y axis becomes to $(-\sin\theta, \cos\theta, 0)$, and let $\boldsymbol{e}_x$ be the vector product of vectors $(-\sin\theta, \cos\theta, 0)$ and $\boldsymbol{e}_z$. Moreover, let $\boldsymbol{e}_y$ be the vector product of vectors $\boldsymbol{e}_z$ and $\boldsymbol{e}_x$. We define the matrix $\boldsymbol{R}$ so that whose row vectors are $\boldsymbol{e}_x$, $\boldsymbol{e}_y$, and $\boldsymbol{e}_z$ in the form

$$\boldsymbol{R} = \boldsymbol{R}(g_x, g_y, \theta) = (\boldsymbol{e}_x, \boldsymbol{e}_y, \boldsymbol{e}_z)^\top. \tag{6}$$

The matrix $\boldsymbol{R}$ is a rotation matrix that rotates the camera so as to locate a focusing point at the principal point.

3.3 Image scaling

In order to align the scale of the scene in the bullet-time images, we introduce a scale parameter s for the focal length of the camera(Fig. 1(c)). If we replace the focal length f to the scaled focal length sf, the internal parameter matrix $\boldsymbol{K}(s)$ is expressed by

$$\boldsymbol{K}(s) = \begin{pmatrix} sf & 0 & c_x \\ 0 & sf & c_y \\ 0 & 0 & 1 \end{pmatrix}, \tag{7}$$

and the homography $\boldsymbol{H}$ becomes to

$$\boldsymbol{H} = \boldsymbol{K}(s)\boldsymbol{R}(g_x, g_y, \theta)\boldsymbol{K}^{-1}. \tag{8}$$

4 Parameter estimation by image matching

In this paper, we assume that we already know the camera internal parameter. Then, we estimate the four parameters g_x, g_y, θ, and s by using image matching technique.

4.1 Image matching

Let T be a template image and I'_κ the κ-th image transformed by the homography $\boldsymbol{H}$ defined in Eq. (8). We estimate four parameters g_x, g_y, θ, and s by minimizing the following function $J(g_x, g_y, \theta, s)$:

$$J(g_x, g_y, \theta, s) = \frac{1}{2N} \sum \Big(I'_\kappa(x', y') - T(x, y) \Big)^2, \quad (9)$$

where, $T(x, y)$ and $I_\kappa(x', y')$ mean the pixel values of the points (x, y) and (x', y') in the images T and I'_κ, respectively. N means the number of the pixels commonly included in the both image regions.

Since input images are taken from different positions and disparity appears between the two images, Eq. (9) does not become to 0 theoretically even for optimal parameters. However, if the template image and a target image are taken at near positions, Eq. (9) has small value for optimal parameters. Therefore, we regard the parameters that minimize Eq. (9) as the optimal parameters.

We select a template image among an already transformed image sequence. If we have no transformed image, we select one image among the input image sequence and transform it by manually specifying those four parameters.

Let p_i, $i = 1, ..., 4$ be g_x, g_y, θ, and s, respectively. Then, the derivative of Eq. (9) by p_i is obtained by

$$\frac{\partial J}{\partial p_i} = \frac{1}{N} \sum \Big(I'_\kappa(x', y') - T(x, y) \Big) (I'_{x'} \frac{\partial x'}{\partial p_i} + I'_{y'} \frac{\partial y'}{\partial p_i}), \quad (10)$$

where, $I'_{x'}$ and $I'_{y'}$ are pixel values of the smooth differential images for x' and y' directions. By using the Gauss-Newton approximation, the second derivative of Eq. (9) is obtained by

$$\frac{\partial^2 J}{\partial p_i \partial p_j} = \frac{1}{N} \sum (I'_{x'} \frac{\partial x'}{\partial p_i} + I'_{y'} \frac{\partial y'}{\partial p_i})(I'_{x'} \frac{\partial x'}{\partial p_j} + I'_{y'} \frac{\partial y'}{\partial p_j}). \quad (11)$$

We can show differentials $\partial x'/\partial g_k$, $\partial y'/\partial g_k$, $k = x, y$ as follows:

$$\frac{\partial x'}{\partial g_k} = \frac{h_{11}^{(g_k)} x + h_{12}^{(g_k)} y + h_{13}^{(g_k)}}{h_{31} x + h_{32} y + h_{33}} - \frac{(h_{11} x + h_{12} y + h_{13})(h_{31}^{(g_k)} x + h_{32}^{(g_k)} y + h_{33}^{(g_k)})}{(h_{31} x + h_{32} y + h_{33})^2}, \quad (12)$$

$$\frac{\partial y'}{\partial g_k} = \frac{h_{21}^{(g_k)} x + h_{22}^{(g_k)} y + h_{23}^{(g_k)}}{h_{31} x + h_{32} y + h_{33}} - \frac{(h_{21} x + h_{22} y + h_{23})(h_{31}^{(g_k)} x + h_{32}^{(g_k)} y + h_{33}^{(g_k)})}{(h_{31} x + h_{32} y + h_{33})^2}, \quad (13)$$

where, we define $h_{ij}^{(g_k)}$, $i, j = 1, ..., 3$, $k = x, y$ as the differentials of h_{ij} by g_k. Here by limitation of the pages, we omit the details. We also have the differentials $\partial x'/\partial\theta$, $\partial y'/\partial\theta$, $\partial x'/\partial s$, and $\partial y'/s$ as follows:

$$\frac{\partial x'}{\partial \theta} = sf \frac{R_{11}^{(\theta)}(x - c_x) + R_{12}^{(\theta)}(y - c_y) + R_{13}^{(\theta)} f}{R_{31}(x - c_x) + R_{32}(y - c_y) + R_{33} f}, \quad (14)$$

$$\frac{\partial y'}{\partial \theta} = sf \frac{R_{21}^{(\theta)}(x - c_x) + R_{22}^{(\theta)}(y - c_y) + R_{23}^{(\theta)} f}{R_{31}(x - c_x) + R_{32}(y - c_y) + R_{33} f}, \quad (15)$$

$$\frac{\partial x'}{\partial s} = f \frac{R_{11}(x - c_x) + R_{12}(y - c_y) + R_{13} f}{R_{31}(x - c_x) + R_{32}(y - c_y) + R_{33} f}, \quad (16)$$

$$\frac{\partial y'}{\partial s} = f \frac{R_{21}(x - c_x) + R_{22}(y - c_y) + R_{23} f}{R_{31}(x - c_x) + R_{32}(y - c_y) + R_{33} f}, \quad (17)$$

where, R_{ij} is an (ij) element of the rotation matrix $\boldsymbol{R}$ defined by Eq. (6) and $R_{ij}^{(\theta)}$ is its differential by θ.

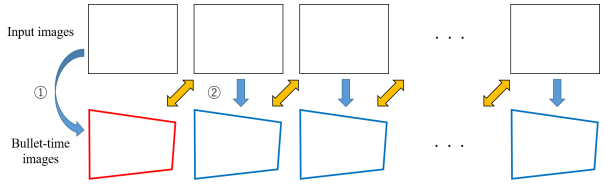


Figure 2. Flow of our bullet-time image generation

4.2 Levenberg-Marquardt method

By using the Levenberg-Marquardt method, we estimate the parameters that minimize the Eq. (9). We summarize the algorithm as follows:

1. Initialize (g_x, g_y), θ, and s. Set $C = 0.0001$.
2. Compute $\boldsymbol{H}$ and the initial value of Eq. (9) as J_0.
3. Compute the first derivatives $\partial J/\partial g_x$, $\partial J/\partial g_y$, $\partial J/\partial\theta$, and $\partial J/\partial s$, and the second derivatives $\partial^2 J/\partial g_x^2$, $\partial^2 J/\partial g_x \partial g_y$, $\partial^2 J/\partial g_x \partial\theta$, $\partial^2 J/\partial g_x \partial s$, $\partial^2 J/\partial g_y^2$, $\partial^2 J/\partial g_y \partial\theta$, $\partial^2 J/\partial g_y \partial s$, $\partial^2 J/\partial\theta^2$, $\partial^2 J/\partial\theta \partial s$, and $\partial^2 J/\partial s^2$.

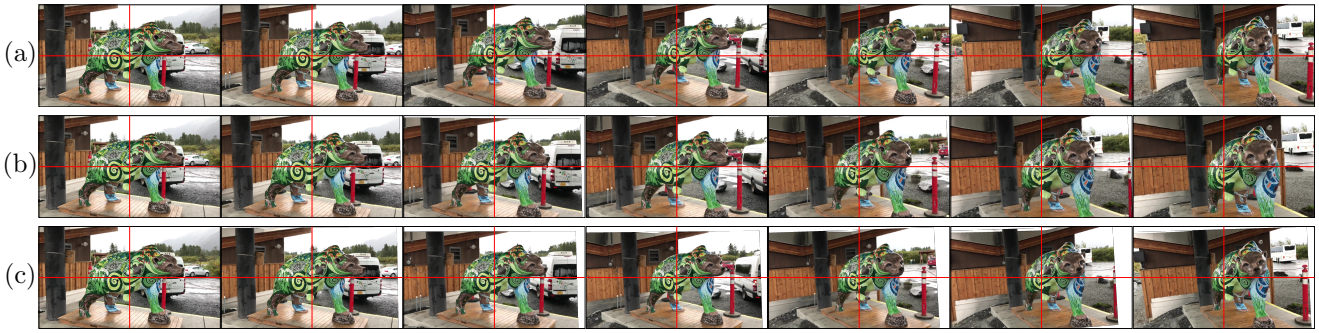


Figure 3. Generated bullet-time images: colorful-bear. (a) input images, (b) proposed method, (c) existing method with 3-D information

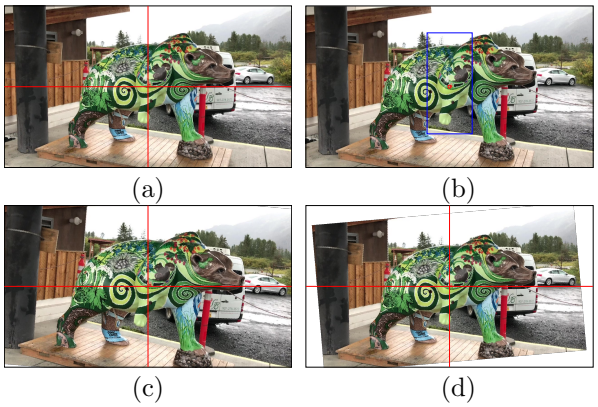


Figure 4. Parameter estimation test. (a) Original image of the first frame. Red lines show the center line of the image. (b) Transformed template image. Red circle shows the manually specified focusing point. The parameters θ and s are set to 0° and 1, respectively. Blue rectangular region shows the target region for image matching. (c) Target image that is transformed from (a). (d) Generated image by our method.

4. Solve the following equation.

$$\begin{pmatrix} (1+C)\partial^2 J/\partial g_x^2 & \partial^2 J/\partial g_x \partial g_y & \partial^2 J/\partial g_x \partial \theta & \partial^2 J/\partial g_x \partial s \\ \partial^2 J/\partial g_x \partial g_y & (1+C)\partial^2 J/\partial g_y^2 & \partial^2 J/\partial g_y \partial \theta & \partial^2 J/\partial g_y \partial s \\ \partial^2 J/\partial g_x \partial \theta & \partial^2 J/\partial g_y \partial \theta & (1+C)\partial^2 J/\partial \theta^2 & \partial^2 J/\partial \theta \partial s \\ \partial^2 J/\partial g_x \partial s & \partial^2 J/\partial g_y \partial s & \partial^2 J/\partial \theta \partial s & (1+C)\partial^2 J/\partial s^2 \end{pmatrix} \begin{pmatrix} \Delta g_x \\ \Delta g_y \\ \Delta \theta \\ \Delta s \end{pmatrix} = - \begin{pmatrix} \partial J/\partial g_x \\ \partial J/\partial g_y \\ \partial J/\partial \theta \\ \partial J/\partial s \end{pmatrix} \tag{18}$$

5. Compute $J = J(g_x+\Delta g_x, g_y+\Delta g_y, \theta+\Delta\theta, s+\Delta s)$.

6. If $|J - J_0| < \delta$ or $J < J_0$, go to Step 7. Else, let $C \leftarrow 10C$ and go back to Step 4.

7. Modify the parameters as $g_x \leftarrow g_x + \Delta g_x$, $g_y \leftarrow g_y + \Delta g_y$, $\theta \leftarrow \theta + \Delta\theta$, and $s \leftarrow s + \Delta s$,

8. If $|\Delta g_x| < \epsilon_{g_x}$ and $|\Delta g_y| < \epsilon_{g_y}$ and $|\Delta\theta| < \epsilon_\theta$ and $|\Delta s| < \epsilon_s$, return g_x, g_y, θ and s, and stop the procedure. Else, let $J_0 \leftarrow J$ and $C \leftarrow C/10$ and go back to Step 3.

4.3 Flow of our method

We summarize the flow of our bullet-time image generation(Fig. 2).

1. Initial template image generation:

   We first select one image among input image sequences and convert it by the homography $\boldsymbol{H}$. For computing the homography $\boldsymbol{H}$, we manually specify the image coordinates of a focusing point (g_x, g_y) and set the values of θ and s be 0° and 1, respectively (Fig. 2 ①). We use it as the template image for the first parameter estimation.

2. Parameter estimation and bullet-time image generation:

   We estimate a homography parameters by comparing the template image with the neighboring input image and generate bullet-time image by the estimated parameters(Fig. 2 ②). For the initial values of the proposed method, we use the estimated values in the previous estimation step.

   We set the generated bullet-time image as the new template image and repeat Step 2 for all the input images.

5 Experiments

We did two experiments for confirm the efficiency of our proposed method.

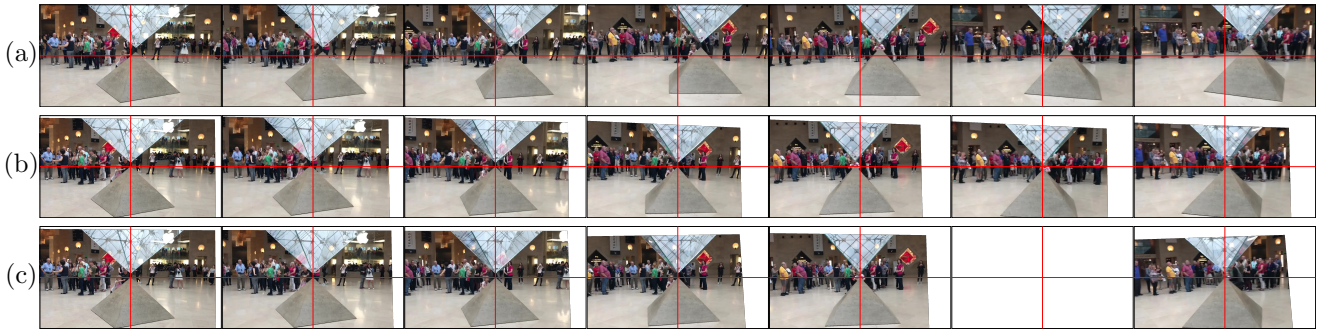


Figure 5. Generated bullet-time images: louvre. (a) input images, (b) proposed method, (c) existing method with 3-D information

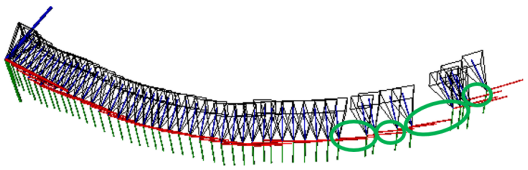


Figure 6. Camera positions reconstructed by VisualSFM. The red, green, and blue lines indicate the X, Y, and Z axes of the camera coordinate system for each camera. The green circles show the camera positions that can not compute their poses by VisualSFM.

5.1 Parameter estimation

First, we tested parameter estimation of our method. We used the first frame of Fig. 3(a) for this experiment. Figure 4(a) is the original image. We manually specified a focusing point, which was shown as the red circle in Fig. 4(b). Figure 4(b) is the transformed image so that the focusing point moves to the image center. Here, we set $\theta = 0$ and $s = 1.0$. We used the blue rectangular region for image matching. We also manually applied the image scaling and rotation for the image shown in Fig. 4(a). We used the scaling parameter $s = 1.1$ and the rotation angle $\theta = -5°$. The transfomed image is shown in Fig. 4(c). By using Fig. 4(b) as the template image and Fig. 4(c) as the target image, We applied our proposed method and estimated the homography parameters.

The estimated values are $s = 0.90879$ and $\theta = 4.9721°$. Since $1.1 \times 0.90879 = 0.99967$ and $5.0 - 4.9721 = 0.0279$, we confirmed that the estimated values are accurate. We also checked the estimated focusing point visually, then we confirmed that the focusing point was located at the image center. We show the generated bullet-time image in Fig. 4(d).

5.2 Bullet-time image generation

5.2.1 Data1: colorful-bear

We generated a bullet-time images for the input 100 frame sequence (Fig. 3(a)). We selected the first frame as the initial template image and used the same setting shown in the first experiment(Fig. 4(a)). Fig. 3(b) shows the generated bullet-time images by our method. For the initial values of θ and s, we used the estimated values for the previous image. For the focusing point, we applied a feature point matching between the template image and the target image and used the result as the initial value of our method. We also used the generated bullet-time image as the template image of the next frame. From Fig. 3(b), we can see that the focusing point was not located at the image center after the 65-th frame. Because, in this experiment, we estimated homography parameters by using the neighboring frame as the template, miss estimation of the focusing point propagated frame to frame.

For comparison, we reconstructed the camera motions and the 3-D position of the focusing point by Visual SFM[8] and generated the bullet-time images by the method of Sakamoto[3]. Figure 3(c) shows the resulting bullet-time images. As you can see that the generated images look natural and the focusing point is fixed at the image center. By comparing the result of two methods, the scaling parameters were estimated larger than that of the existing method. These error also propagated frame to frame.

5.2.2 Data2: louvre

We also tested our method for other input sequences. Figure 5(a) is seven decimated images from 60 frame sequence. In this scene, the view-point changes smoothly, however, two persons move behind the target object.

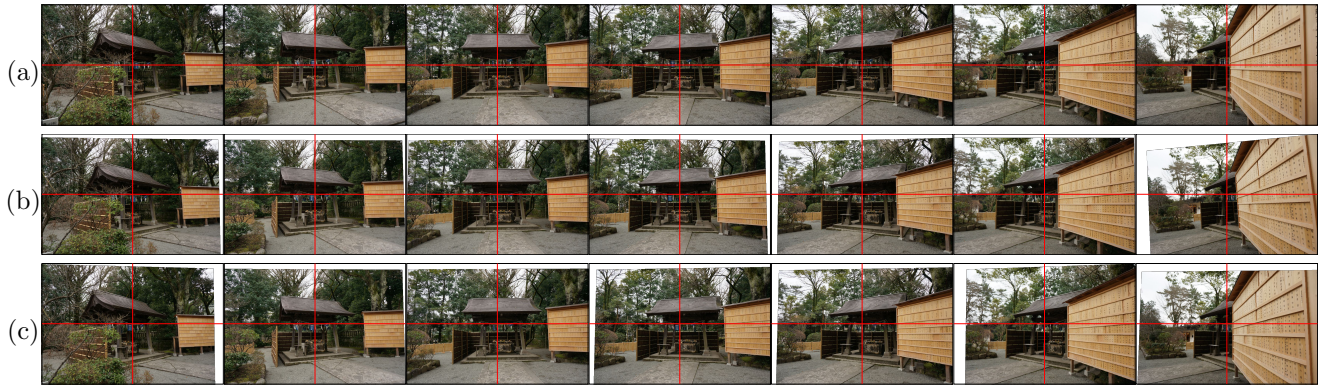


Figure 7. Generated bullet-time images: Temizuya. (a) input images, (b) proposed method that estimates all four parameters, (c) proposed method that estimates only scaling and angle parameters.

We selected the first frame as the initial template image. Figure 5(b) shows the generated bullet-time images by our method. As you can see, the focusing point is almost fixed at the image center and the generated bullet-time images look like natural.

Figure 5(c) shows the bullet-time image generated by the method of Sakamoto. For this scene, we could not compute eleven camera poses by VisualSFM. We shows the reconstructed camera poses in Fig. 6 and marked the missing camera positions by the green circles. In Fig. 5(c), we could not generate the bullet-time image of the 50-th frame because of the lack of its camera pose information.

From this result, we can show that the proposed method can generate bullet-time images of the scenes for which we fail camera pose estimation. This is an advantage of our proposed method for the existing method.

5.2.3 Data3: Temizuya

Figure 7(a) is seven decimated images from 22 frame sequence. In this scene, the view-point changes significantly frame to frame. So, miss-estimation of the focusing point occurs in some images (Fig. 7(b)). Therefore, we manually specified the focusing point on each frame and estimated only the scale and the rotation parameters. The resulting bullet-time images are shown in Fig. 7(c). From this result, we could generate visually natural bullet-time images.

However, in order to improve the performance of our method, we need to stably estimate focusing point. This is one of the future works.

6 Conclusions

We proposed a new method for generating bullet-time images without any 3-D information. We re-parameterized the homography without 3-D information of the camera and the target point and estimated the homography parameters by using an image matching technique. From the experiments, we confirmed the efficiency of our method. For generating a bullet-time sequence, however, we should solve the problem that how do we select a proper template image and initial parameters.

Acknowledgements: This work was supported in part by JSPS Grant-in-Aid for Scientific Research(C) (18K11353).

References

[1] N. Akechi, I. Kitahara, R. Sakamoto, and Y. Ohta, Multi-resolution bullet-time effect, SIGGRAPH-AISA, 2014.

[2] R. Hartley and A. Zisserman, Multiple view geometry in computer vision, Cambridge Univ. Press, 2000.

[3] R. Sakamoto and D. Chen, Optimized Homography for Natural and High-Quality Bullet-Time Camera Work, IPSJ SIG Technical Reports Interactive, pp. 65-70, 2014.

[4] N. Snavely, S. M. Seitz, and R. Szeliski, Photo Tourism: Exploring Photo Collections in 3D, ACM Transactions on Graphics, Vol. 25, pp. 835–846, 2006.

[5] N. Snavely, S. M. Seitz, and R. Szeliski, Modeling the World from Internet Photo Collections, International Journal of Computer Vision, Vol. 80, No. 2, pp. 189–210, 2008.

[6] R. Szeliski, Computer vision: Algorithms and applications, Stringier 2011.

[7] K. Tomiyama, and Y. Iwadate, Development of Multi-View HDTV Image Generation System, The journal of the Institute of Image Information and Television Engineers, Vol. 64, No. 4, pp. 622–628, 2010.

[8] C. Wu, VisualSFM: A Visual Structure from Motion System, http://ccwu.me/vsfm/