Generation of a Super-resolved Stereo Video Using Two Synchronized Videos with Different Magnifications

Yusuke Hayashi[1], Norihiko Kawai[1], Tomokazu Sato[1],
Miyuki Okumoto[2], and Naokazu Yokoya[1]

[1] Graduate School of Information Science, Nara Institute of Science and Technology
8916-5 Takayama, Ikoma, Nara 630-0192, Japan
{hayashi.yusuke.hq7,norihi-k,tomoka-s,yokoya}@is.naist.jp
http://yokoya.naist.jp

[2] Tokuyama College of Technology, Gakuendai, Shunan, Yamaguchi 745-8585, Japan
okumoto@tokuyama.ac.jp

Abstract. In this paper, we address the problem of changing the optical zoom magnification of stereo video that uses a stereo camera system with 4K or 8K digital cameras. We proposes a solution for generating a zoomed stereo video from a pair of zoomed and non-zoomed videos. To achieve this, part of the non-zoomed video image is isolated and super-resolved, so that the resolution of the image becomes the same as that of the optically-zoomed image. The non-zoomed video is super-resolved by energy minimization using the optically-zoomed image as an example. The effectiveness of this method is validated through experiments.

Keywords: Stereo video, Super-resolution, Energy minimization

1 Introduction

The increasing popularity of 3D TV is leading to an increase in 3D video generation. Two methods are typically used to generate stereo videos of real scenes: one uses two identical video cameras that are arranged in parallel, and the other converts 2D video (captured using a single video camera) to 3D [1]. In this paper, we address the problem of changing the optical zoom magnification of stereo video. When using two cameras, the stereo camera system that synchronizes the optical zoom magnifications of the two cameras is mechanically complex and costly. This is especially true for a 3D digital cinema camera system using 4K (4096×2160) or 8K (8192×4320) cameras. However, 2D/3D conversion often gives unnatural stereoscopic images [2]. This paper proposes a solution by generating a zoomed stereo video from a pair of optically-zoomed and non-zoomed videos in the stereo camera system.

Most conventional methods for super-resolution can be broadly classified into two categories. One uses multiple low-resolution images and the other uses high-resolution images as examples. In the former method, input images are super-resolved by aligning multiple low-resolution images with sub-pixel accuracy [3, 4,

5, 6]. This method requires the capture of many images of the same scene without moving objects. Therefore, this approach is not suitable for our solution. In the latter example-based method, correspondences between a target low-resolution image and example high-resolution images are determined, which are used to super-resolve the target image [7, 8]. In this category, several attempts have been made at efficiently obtaining good results. Hashimoto et al. [9] proposed a method of efficiently searching for correspondences using a binary tree dictionary. Baker et al. [10] restricted the category of examples in the database by considering that the quality of the resultant image largely depends on the examples.

In this paper, we use the example-based super-resolution approach. We propose a new system to generate a zoomed stereo video from two synchronized cameras with different magnifications, which has not previously been attempted. In our method, any camera unit can be used for generating stereo video images unlike the mechanical approach where a new system is required every time a new camera unit is released. In the proposed method, we isolate part of a non-zoomed video image that corresponds to the other optically-zoomed image. This is then super-resolved using the example optically-zoomed image, so that it has the same zoomed resolution. This is effective because there is a high correlation between the target low-resolution image and the high-resolution image captured by the two cameras arranged in parallel.

2 Generation of Super-resolved Stereo Video

Figure 1 illustrates the flow of the proposed method. In this study, we shoot a target scene using two cameras that are set so that their optical axes are parallel, and horizontal lines are parallel. To simplify our explanation, let the optically zoomed left-eye image be L, and the non-zoomed right-eye image be R. First, the section corresponding to L is cut out from R, and the cutout right-eye image is called R_d (Process I). Next, R_d is enlarged to the size of L using initial values for the generated image, R_s (Process II). Then, an energy function is defined based on the pattern similarity between L and R_s, and the difference in intensity between R_d and R_s. We then minimize the energy to super-resolve the image, by repeating two processes: search L for a similar texture (Process III-1) and update all the pixel values in R_s (Process III-2). In the following sections, we describe the cutout and energy minimization methods.

2.1 Cutout of Non-zoomed Image

To isolate R_d so that it corresponds to the shooting range of L, we estimate the transform matrix $\mathbf{M}$ that projects the four corners of L onto four points in R for every frame, as shown in Fig. 2. The transform matrix, $\mathbf{M}$, defined as

$$\mathbf{M} = \begin{pmatrix} s & 0 & t_x \\ 0 & s & t_y \\ 0 & 0 & 1 \end{pmatrix}. \tag{1}$$

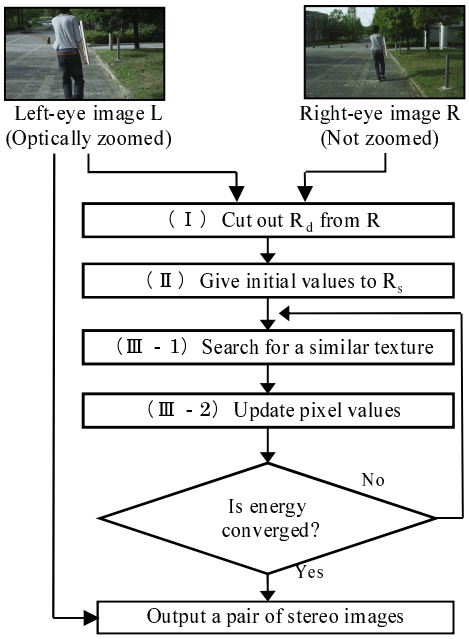


Fig. 1: Flow diagram of proposed method.

The translation parameters, t_x and t_y, are determined in advance by calibrating the camera. The scaling parameter, s (where the zoom magnification is $1/s$), is determined so that a similarity measure between R_d and L is maximized.

The similarity measure is based on the normalized cross-correlation between two graphs generated using the average intensities of the scanlines in images L and R. To successfully generate stereo images that can be fused by human eyes, it is important to align the horizontal lines. Thus, to determine the magnification parameter s, it is more effective to use the average intensity for each horizontal line than the vertical line. As shown in Fig. 3, graph $h_L(y)$ of L is generated so that the vertical axis is the y coordinate of L, and the horizontal axis is the average pixel value on one horizontal line. For the graph $h_R(y)$, R_d is first cut out from R using the tentative matrix $\mathbf{M}$ and then enlarged to the size of L. The tentative graph $h_R(y)$ is then generated from the enlarged R_d in the same way as for L. The graphs for the R, G and B components are generated, and the sum of the normalized cross-correlations for R, G and B are used as a similarity measure.

After determining $\mathbf{M}$, we need to compensate for the color tone of R_d. The R, G and B values of the image R_d are linearly transformed so that graphs $h_R(y)$ for RGB fit the graphs $h_L(y)$ for RGB in a least-squares manner.

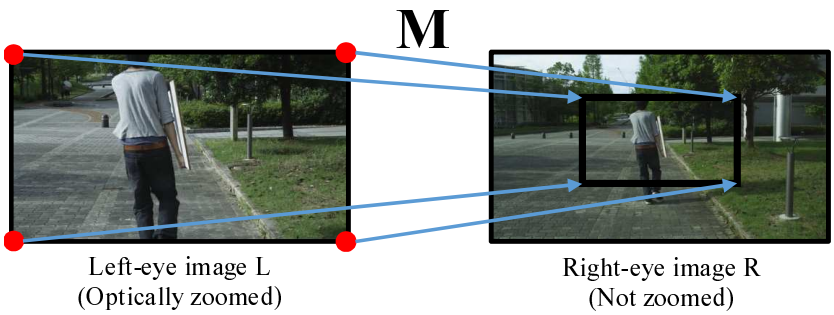


Fig. 2: Projection of four corners using transform matrix $\mathbf{M}$.

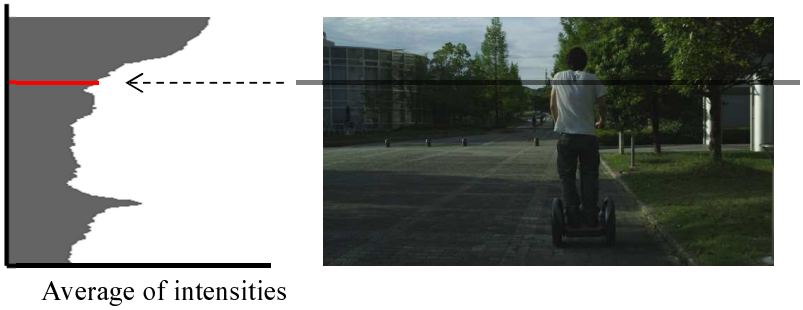


Fig. 3: Example of graph for average intensities of scan lines.

2.2 Definition of Energy Function

The energy function, E, is defined using two different kinds of energy terms. E_{ssd} represents the pattern similarity between R_s and L, and E_{dif} represents the intensity difference between R_s and R_d.

$$E = \sum_{\mathbf{x}_i \in R_s} \{\lambda E_{ssd}(\mathbf{x}_i, \mathbf{x}_j) + (1-\lambda) E_{dif}(\mathbf{x}_i, g(\mathbf{x}_i))\}, \tag{2}$$

$$E_{ssd}(\mathbf{x}_i, \mathbf{x}_j) = \sum_{\mathbf{p} \in W} \{R_s(\mathbf{x}_i + \mathbf{p}) - L(\mathbf{x}_j + \mathbf{p})\}^2, \tag{3}$$

$$E_{dif}(\mathbf{x}_i, g(\mathbf{x}_i)) = \{R_s(\mathbf{x}_i) - R_d(g(\mathbf{x}_i))\}^2. \tag{4}$$

Here, W is a square window, and λ is a weight for balancing the two terms. $\mathbf{x}_i$ denotes a pixel in R_s, and $\mathbf{x}_j$ is a pixel from L. $R_s(\mathbf{x}_i)$, $R_d(\mathbf{x}_i)$ and $L(\mathbf{x}_i)$ represent the intensities of the pixel $\mathbf{x}_i$ in the images R_s, R_d and L. $g(\mathbf{x}_i)$ denotes a pixel position in R_d that corresponds to the pixel $\mathbf{x}_i$ in R_s. The relationship is

$$g(\mathbf{x}_i) = \mathbf{M}'\mathbf{x}_i, \tag{5}$$

where the matrix $\mathbf{M}'$ is the same as matrix $\mathbf{M}$ except that the translation parameters are 0. E_{ssd} represents the effect of increasing the resolution of the generated image, and E_{dif} represents the preservation of the texture of the original right-eye image.

2.3 Iterative Energy Minimization

The energy function, E, is minimized using the framework of a greedy algorithm. In the proposed method, the energy function is minimized by iterating the following two processes: search for similar patterns in L (Ⅲ-1) and update pixel values in R_s (Ⅲ-2).

Process (Ⅲ-1). The whole pixel values in R_s are fixed, and the position $\mathbf{x}_k$ of the most similar texture pattern to $\mathbf{x}_i$ is updated so as to satisfy the following equation:

$$\mathbf{x}_k = f(\mathbf{x}_i) = \underset{\mathbf{x}_j \in \phi(\mathbf{x}_i)}{\operatorname{argmin}} E_{ssd}(\mathbf{x}_i, \mathbf{x}_j), \tag{6}$$

where $\phi(\mathbf{x}_i)$ is the search region of L that corresponds to the pixel $\mathbf{x}_i$ in the generated image. Here, $\phi(\mathbf{x}_i)$ includes a set of pixels on the epipolar line and several pixels above and below the epipolar line.

Process (Ⅲ-2). The pixel values $R_s(\mathbf{x}_i)$ in the generated image are updated in parallel so as to minimize the energy function E defined in Eq.(2) while keeping all the similar pairs fixed. The energy function E is resolved into the element energy $E(\mathbf{x}_i)$ for each pixel $\mathbf{x}_i$ in R_s.

$$E(\mathbf{x}_i) = \lambda \sum_{\mathbf{p} \in W} \{R_s(\mathbf{x}_i) - L(f(\mathbf{x}_i + \mathbf{p}) - \mathbf{p})\}^2 + (1-\lambda)\{R_s(\mathbf{x}_i) - R_d(g(\mathbf{x}_i))\}^2. \tag{7}$$

Each element energy includes only one parameter and the total energy, E, consists of the sum of all element energies. Therefore, E can be minimized by minimizing each element energy. $R_s(\mathbf{x}_i)$ that minimizes $E(\mathbf{x}_i)$ can be calculated by differentiating $E(\mathbf{x}_i)$ with respect to $R_s(\mathbf{x}_i)$, and is

$$R_s(\mathbf{x}_i) = \frac{\lambda \sum_{\mathbf{p} \in W} L(f(\mathbf{x}_i + \mathbf{p}) - \mathbf{p}) + (1-\lambda) R_d(g(\mathbf{x}_i))}{\lambda N_W + (1-\lambda)}, \tag{8}$$

were N_W denotes the number of pixels in the window.

Additionally, in order to reduce the computational cost and avoid local minima, we use a coarse-to-fine approach for energy minimization. We first generate an image pyramid. In the coarsest level, the above processes (Ⅲ-1) and (Ⅲ-2) are iterated until the energy converges. The pixel correspondences between images L and R_s in the last iteration are committed memory. In subsequent levels, the generated texture in the previous level is used for the initial pixel values.

Table 1: Estimated zoom magnifications

Ground truth	1.0	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9	2.0
Estimated result	1.001	1.106	1.204	1.300	1.396	1.497	1.613	1.721	1.831	1.908	1.996

The processes (Ⅲ-1) and (Ⅲ-2) are iterated until convergence where the search area for each pixel in R_s in the process (Ⅲ-1) is limited to a range around the memorized corresponding position in the previous level. In addition, in the finest level, we repeat the energy minimization while reducing the size of the window. This enables more detailed textures to be reproduced.

3 Experiments

In order to demonstrate the effectiveness of the proposed method, we have performed experiments using a stereo image dataset [11, 12] and real stereo video captured by two parallel digital video cameras (Red Digital Cinema Camera Company: Red One).

We first confirmed the validity of the estimation of the zoom scale by process (I). Then, we calculated the super-resolved results, and compared them to the results of conventional methods.

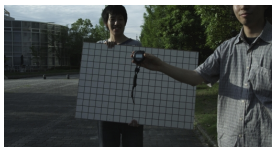
3.1 Confirmation of Validity of the Zoom Scale Estimation

In this section, we confirm the validity of the scale estimation method described in Section 2.1. In this experiment, we used a stereo image (right and left images with 4096×2160 pixels) of a real stereo video in which the zoom magnifications of the two cameras were the same. To simulate a stereo pair with different zoom magnifications, the right-eye image was resized as a non-zoomed image and the left-eye image was cut out and resized as an optically-zoomed image so that the resolutions of the two images became the same. We resized the right-eye image to 2048×1080 pixels as a non-zoomed image. We cut out the left-eye image to simulate a zoom, changing the magnification from 1.0 to 2.0 with a 0.1 skip and resizing the respective cutout images to 2048×1080 pixels. Figure 4 shows the examples of input stereo pairs with different zoom magnifications.

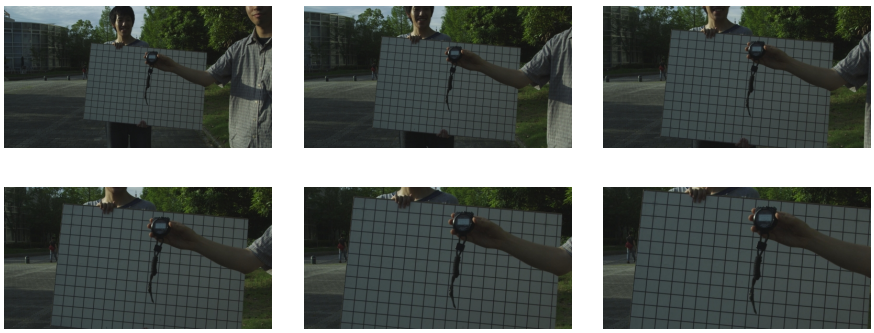
Table 1 shows the estimated zoom magnifications using the proposed method. The estimated magnifications were almost the same as the ground truth in many cases. However, when the zoom magnification was 1.8, the error ratio was 1.7%. Other cases had worse results. This is attributed to the fact that the graph $h_L(y)$ is relatively flat when the magnification is 1.8, because the board with a regular pattern occupies a large area of the image.

3.2 Experiments Using a Stereo Image Dataset

In this experiment, we used two images (Aloe with 640×554 pixels, and Tsukuba with 384×288 pixels) selected from the stereo image dataset [11, 12]. We reduced



(a) Simulated a non-zoomed right-eye image.



(b) Simulated optically-zoomed left-eye images with different magnifications (From left to right, top to bottom: 1.0, 1.2, 1.4, 1.6, 1.8, 2.0).

Fig. 4: Examples of input stereo pairs with different zoom magnifications.

the size of each right image to simulate a pair of input images with different zoom magnifications. In this experiment, we did not use an image pyramid for the coarse-to-fine approach because the input image size was small enough. In the original scale, we first set the window size W to be 7×7 pixels, and reduced it to 5×5 and 3×3 pixels as the energy converges.

First, we verified the results using different λ values (from 0.0 to 1.0), where λ balances the two energy terms. For this simulation, we resized each right image to 1/2 of the original size and super-resolved the images using the proposed method with a known scaling parameter. Figure 5 shows the PSNR values of the generated images for different λ values. From this result, we can confirm that the PSNR becomes higher as λ approaches 0.6.

Next, we compared the results of three methods: the proposed method with $\lambda = 0.6$, an example-based super-resolution method [9], and bi-cubic interpolation. We resized each right image to 1/2 and 1/4 of the original size. Figure 6 shows the results that correspond to the two resized rates using the three methods. By comparing these images, we can confirm that the proposed method successfully generated the high-frequency component. When the resized rate was 1/4, the difference was especially noticeable. Figure 7 shows the PSNR values of the results of each method. The PSNR values of the proposed method are the highest for both images, when compared with the other methods. However, as shown in Fig. 8, the image generated using our method with $\lambda = 1.0$ includes

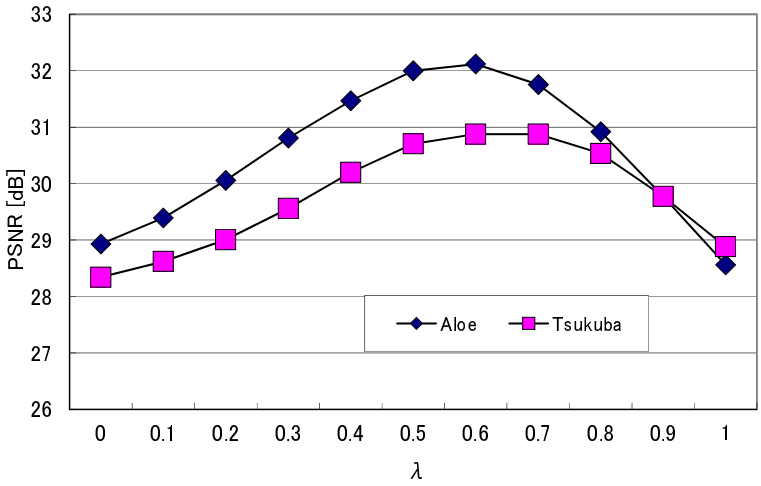


Fig. 5: PSNR with different λ.

some incorrect texture patterns. This is because similar patterns do not exist in the optically zoomed high-resolution image due to occlusions.

3.3 Experiments Using Real Stereo Video

In this experiment, we captured stereo video with 4096×2160 pixels using two digital video cameras. We changed the magnification of the left camera and kept the right magnification fixed. In the proposed method, $\lambda = 0.6$, as was suggested by our preliminary experiments (see Fig. 5). For the coarse-to-fine approach, we resized the input image to 1/8, 1/4 and 1/2 of the original size. We set the window size, W, to be 13×13 pixels. At the finest level, the window size was reduced to 9×9 and 5×5 pixels as the energy converged. In this experiment, approximately 5 minutes were required to generate each pair of stereo images using a PC (Intel Core i7 3.40GHz of CPU and 8.00 GB of memory).

Figure 9 shows the input images of the target frame and a pair of stereo images generated using the proposed method. The generated stereo images can be viewed stereoscopically. We have also confirmed that our method compensated for the color tone of the generated right-eye image. Figure 10 shows the magnified images generated by the proposed method and by using bi-cubic interpolation. Using Fig. 10, we can confirm that the texture generated by the proposed method is clearer than that by bi-cubic interpolation. The experimental results demonstrate the effectiveness of this method for improving the quality of the real stereo video.

(a) Input images.

(b) Proposed method (Resized rate: 1/2).

(c) Proposed method (Resized rate: 1/4).

(d) Example-based SR [9](Resized rate: 1/2).

(e) Example-based SR [9](Resized rate: 1/4).

(f) Bi-cubic interpolation (Resized rate: 1/2).

(g) Bi-cubic interpolation (Resized rate: 1/4).

Fig. 6: Super-resolved results by three methods.

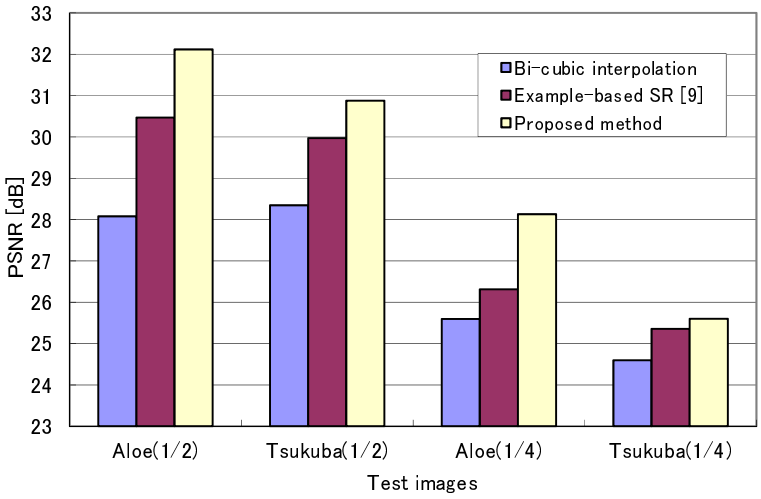


Fig. 7: PSNR of all test images.

Fig. 8: Example of unnatural generated texture.

4 Conclusion

We have proposed a new system for generating high-resolution stereo video from two synchronized videos with different magnifications. In the proposed method, a non-zoomed video is super-resolved by energy minimization, using the optically-zoomed image as an example. In experiments using a stereo dataset and real video, we have demonstrated the effectiveness of the proposed method by comparing our results to conventional methods. In the future, we will focus on improving the quality of the generated image by considering occlusions.

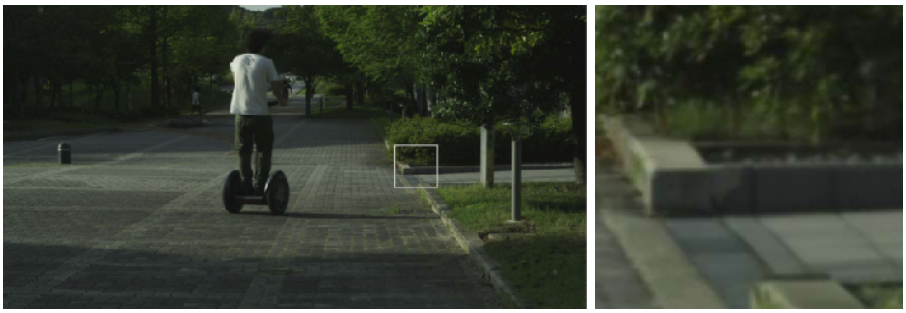
Acknowledgments. This research was partially supported by Grant-in-Aid for Scientific Research (A), No. 23240024 and Challenging Exploratory Research, No. 25540086.

(a) Input left-eye image L.

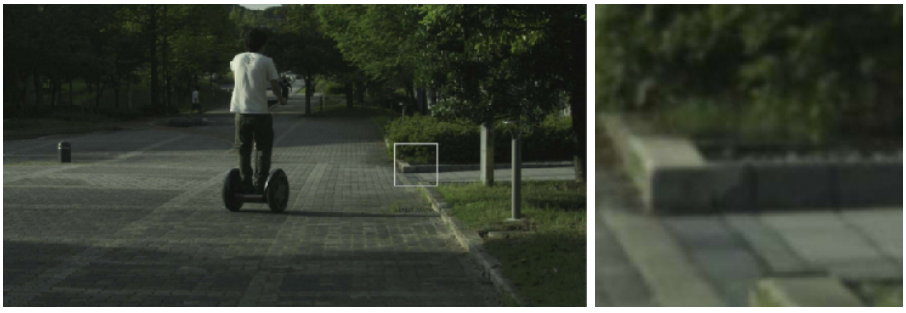
(b) Input right-eye image R.

(c) Generated stereo image.

Fig. 9: Input images and generated result.



(a) Proposed method.



(b) Bi-cubic interpolation.

Fig. 10: Comparison of results.

References

1. Mendiburu, B.: 3D Movie Making - Stereoscopic Digital Cinema from Script to Screen. Focal Press (2009)
2. Mita, T.: 2D to 3D image upconversion technology. Journal of Institute of Image Information and Television Engineers, vol. 67, no. 2, pp. 116–121 (2013)(in Japanese)
3. Park, S.C., Park, M.K., Kang, M.G.: Super-resolution image reconstruction: A technical overview. IEEE Signal Processing Magazine , vol. 20, no. 3, pp. 21–36 (2003)
4. Iketani, A., Sato, T., Ikeda, S., Kanbara, M., Nakajima, A., Yokoya, N.: Super-resolved video mosaicing for documents by extrinsic camera parameter estimation. Proc. Int. Conf. on Computer Vision and Graphics, pp. 327–336 (2004)
5. Irani, M., Peleg, S.: Improving resolution by image registration. Graphical Models and Image Processing, vol. 53, no. 3, pp. 231–239 (1991)
6. Farsiu, S., Robinson, M., Elad, M., Milanfar, P.:Fast and robust multiframe super resolution. IEEE Trans. Image Processing, vol. 13, no. 10, pp. 1327–1344 (2004)
7. Freeman, W.T., Jones, T.R., Pasztor, E.C.:Example-based super-resolution. IEEE Computer Graphics and Applications, vol. 22, pp. 56–65 (2002)
8. Begin, I., Ferrie, F.P.: Blind super-resolution using a learning-based approach. Proc. Int. Conf. on Pattern Recognition, vol. 2, pp. 85–89 (2004)
9. Hashimoto, A., Nakaya, T., Kuroki, N., Hirose, T., Numa, M.: Binary tree dictionary for learning-based super-resolution. IEICE Trans. D, vol. J96-D, no. 2, pp. 357–361 (2013) (in Japanese)
10. Baker, S., Kanade, T.: Hallucinating faces. Proc. IEEE Int. Conf. on Automatic Face and Gesture Recognition, pp. 83–88 (2000)
11. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. Int. Journal of Computer Vision, vol. 47, no. 1–3, pp. 7–42 (2002)
12. Hirschmuller, H., Scharstein, D.:Evaluation of cost functions for stereo matching. Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pp. 1–8 (2007)