Stereo Image Retargeting with Shift-Map*  

Ryo NAKASHIMA†(a), Student Member, Kei UTSUGI†, Keita TAKAHASHI†, and Takeshi NAEMURA†, Members

SUMMARY We propose a new stereo image retargeting method based on the framework of shift-map image editing. Retargeting is the process of changing the image size according to the target display while preserving as much of the richness of the image as possible, and is often applied to monocular images and videos. Retargeting stereo images poses a new challenge because pixel correspondences between the stereo pair should be preserved to keep the scene’s structure. The main contribution of this paper is integrating a stereo correspondence constraint into the retargeting process. Among several retargeting methods, we adopt shift-map image editing because this framework can be extended naturally to stereo images, as we show in this paper. We confirmed the effectiveness of our method through experiments.

key words: stereo image editing, image retargeting, shift-map image editing

1. Introduction

3D image technology plays an important role in virtual reality and augmented reality systems [1],[2]. Stereoscopic vision, as the simplest form of 3D visions, is widely used because it can provide a higher sense of reality than 2D images by just presenting a pair of stereo images corresponding to left and right eyesights. We focus on the problem of stereo image retargeting, which is essential in postproduction of stereo images.

Retargeting is the process of fitting the image size to various display devices with different resolutions. Recent retargeting methods aim to do more than simply scaling or cropping images. These methods add/remove non-salient regions to/from the target image while preserving as much of the salient regions as possible [3]–[8]. However, most studies have focused on monocular images and videos. Only a few studies have been conducted on stereo images. Retargeting stereo images imposes a new challenge because pixel correspondences between the stereo pair should be preserved to maintain consistency. The key idea of this paper is to integrate a stereo correspondence constraint into the retargeting process to keep the underlying scene structure unchanged.

Among several retargeting methods, we adopt shift-map image editing [7]. This method can be used to perform various editing operations, such as retargeting, inpainting, reshuffling (object rearrangement), and image composition, in a unified manner, with a shift-map, which represents the correspondence between the input and output pixels. The resulting image is obtained by global optimization of an energy function which is defined over the shift-map and encodes pixel saliencies, user-defined constraints, image smoothness, and so on. This framework can be extended naturally to stereo images with our stereo correspondence constraint.

The most similar work to ours is that of Utsugi et al. [9]. They extended seam-carving [3] to stereo images by adding/removing corresponding seams on both images simultaneously. Their method often falls into local optima because seam-carving is based on a greedy search. Our method is more stable due to the nature of global optimization of the shift-map editing framework.

The rest of this paper is constructed as follows. In Sect. 2, we describe the algorithm of shift-map image editing [7] as a preparation. In Sect. 3, we propose our stereo image retargeting method by extending the shift-map framework to stereo images with the stereo correspondence constraint. Experimental validations are presented in Sect. 4. Section 5 summarizes this paper.

2. Shift-Map Image Editing

A shift-map $M$ represents the correspondence between an input image $I'$ and an output image $I$. When the shift of a pixel $p$ is $M(p)$, the output pixel $I(p)$ is derived from $I'(p')$ where $p' = p + M(p)$. An illustration is shown in Fig. 1(a).

The optimal shift-map minimizes the following energy $E$.

$$E(M) = \alpha \sum_{p \in I} E_{data}(M(p)) + \sum_{(p,q) \in N} E_{smooth}(M(p),M(q)).$$  (1)

The data term $E_{data}$ encodes user-defined constraints and pixel saliencies. The smoothness term $E_{smooth}$ penalizes artifacts in the output image and is defined for pairs of neighboring pixels $N$.

The design of the data term $E_{data}$ is key to implementing various image editing tasks. $E_{data}$ takes complicated forms for image reshuffling applications, but becomes relatively simple for image retargeting. The minimum constraint for $E_{data}$ is that the rightmost and leftmost columns of the output image must be derived from the rightmost and leftmost columns of the input image, respectively.

$$E_{data}(M(p)) = \begin{cases} 0 & M(p) = (0,0) \\ \infty & \text{otherwise} \end{cases}$$  (2)
and for the rightmost column:

\[
E_{\text{data}}(M(p)) = \begin{cases} 
0 & M(p) = (I_w' - I_w, 0) \\
\infty & \text{otherwise}
\end{cases}, \tag{3}
\]

where \(I_w\) and \(I_w'\) are the widths of \(I\) and \(I'\), respectively. Optionally, pixel saliencies \(S\) can be specified to the remaining pixels.

\[
E_{\text{data}}(M(p)) = -S(p + M(p)). \tag{4}
\]

To enforce that a pixel in the input image \(I'(p')\) should disappear in the output image, the algorithm sets a larger penalty for selecting \(I'(p')\) by setting a small value to \(S(p')\). Meanwhile, to keep \(I'(p')\), it sets a large value to \(S(p')\).

The smoothness term \(E_{\text{smooth}}\) penalizes discontinuities in the output image. \(E_{\text{smooth}}\) is defined over pairs of neighboring pixels \((p, q) \in N\) as:

\[
E_{\text{smooth}}(M(p), M(q)) = V_1(p, q) + V_2(q, p) + \beta(V_2(p, q) + V_2(q, p)), \tag{5}
\]

\[
V_1(p, q) = (I'(p + M(p)) - I'(p + M(q)))^2, \tag{6}
\]

\[
V_2(p, q) = (\nabla I'(p + M(p)) - \nabla I'(p + M(q)))^2, \tag{7}
\]

where \(\nabla\) denotes the gradient. This equation means that a larger penalty is imposed for larger changes in the color channel with discontinuities of the shift-map \((M(p) \neq M(q))\). Note that image variations without shift-map discontinuities are not penalized.

Objective function \(E\) is minimized using multi-label graph cuts\([10]–[12]\). A hierarchical optimization is used to reduce computational cost. First, all possible shifts are examined in the lowest resolution. In a higher resolution, the shift-map obtained from the lower resolution is used as the initial guess. Limiting the range of possible shifts \(M(p)\) also decreases the computational cost. For retargeting applications, \([0, \ldots, I_w' - I_w]\) shifts are allowed in the horizontal direction, but vertical shifts are limited within a few pixels.

3. Proposed Method

We propose a stereo image retargeting method based on the framework of shift-map image editing. We integrate a stereo correspondence constraint into the framework to preserve pixel correspondences between a pair of stereo images.

3.1 Shift-Maps for Stereo Images

The proposed method optimizes shift-maps for the left and right images simultaneously to maintain stereo correspondences. Let \(M_L\) and \(M_R\) be shift-maps for the left and right images, respectively, as shown in Fig. 1 (b). The optimal shift-maps minimize the following energy.

\[
E(M_L, M_R) = E_{\text{intra}}(M_L) + E_{\text{intra}}(M_R) + E_{\text{inter}}(M_L, M_R), \tag{8}
\]

where \(E_{\text{intra}}\) is the intra-image energy of a shift-map and \(E_{\text{inter}}\) is the inter-image energy for measuring consistency between stereo images. We use the energies \(E_{\text{data}}\) and \(E_{\text{smooth}}\) in Eq. (1) as \(E_{\text{intra}}\). \(E_{\text{inter}}\) is a new term that represents the stereo correspondence constraint we propose, which we describe next.

3.2 Stereo Correspondence Constraint

\(I_L\) and \(I_R\) denote the output left and right images, and \(I'_L\) and \(I'_R\) the input left and right images. Let \(p_L \in I_L\) and \(p_R \in I_R\) be the pixels shifted from \(p'_L \in I'_L\) and \(p'_R \in I'_R\). Based on the definition of shift-maps \(M_L\) and \(M_R\), we obtain

\[
p'_L = p_L + M_L(p_L), \tag{9}
\]

\[
p'_R = p_R + M_R(p_R). \tag{10}
\]

These relations are illustrated in Fig. 1 (b).

We assume the input stereo images are rectified in advance, and that pixelwise disparities between them are available. Let \(D_{LK}(p'_L)\) be the left-to-right horizontal disparity of
the pixel \(p'_L\). The right-to-left disparity \(D'_{RL}(p'_R)\) is defined similarly. We describe \(p'_L \sim p'_R\) if pixel \(p'_L \in I'_L\) and \(p'_R \in I'_R\) represent the same point of the scene.

\[
p'_L \sim p'_R \iff \begin{pmatrix} p'_L.x - D'_{LR}(p'_L) \\ p'_L.y \end{pmatrix} = \begin{pmatrix} p'_R.x \\ p'_R.y \end{pmatrix}, \quad (11)
\]

where \(*.x, *.y\) denote the x- and y-coordinates of the pixel *, respectively. This state is referred to as the stereo correspondence between \(p'_L\) and \(p'_R\). Without occlusions,

\[
p'_L \sim p'_R \implies D'_{LR}(p'_L) = D'_{RL}(p'_R) \quad (12)
\]

is satisfied by definition.

According to the definition of the shift-maps, it is naturally required that the output disparities, \(D_{LR}(p_L)\) and \(D_{RL}(p_R)\), are derived from those of the input images using shift-maps \(M_L(p_L)\) and \(M_R(p_R)\):

\[
D_{LR}(p_L) = D'_{LR}(p_L + M_L(p_L)) = D'_{LR}(p'_L), \quad (13)
\]

\[
D_{RL}(p_R) = D'_{RL}(p_R + M_R(p_R)) = D'_{RL}(p'_R). \quad (14)
\]

Using Eqs. (13) and (14), we define the stereo correspondence between the output pixels \(p_L \in I'_L\) and \(p_R \in I'_R\) as:

\[
p_L \sim p_R \iff \begin{pmatrix} p_L.x - D_{LR}(p_L) \\ p_L.y \end{pmatrix} = \begin{pmatrix} p_R.x \\ p_R.y \end{pmatrix} \wedge D_{LR}(p_L) = D_{RL}(p_R). \quad (15)
\]

Note that \(D_{LR}(p_L) = D_{RL}(p_R)\) is not always satisfied because \(p_L\) and \(p_R\) are given by independent shift-maps as shown in Eqs. (9) and (10).

Equations (11) and (15) give the stereo correspondence constraint, which can be used to define the inter-image energy \(E_{\text{inter}}\) in Eq. (8).

\[
E_{\text{inter}}(M_L, M_R) = \left( \sum_{p_L \sim p_R} [p'_L \sim p'_R] + \sum_{p_L \not\sim p_R} [p'_L \sim p'_R] \right) K, \quad (16)
\]

where \([\cdot]\) is a function that returns 1 if the condition in the bracket is satisfied, or 0 otherwise. This equation counts the number of pixel pairs with stereo inconsistency before and after retargeting. Stereo inconsistency arises if the correspondence in the input images is broken in the output images or vice versa. \(K\) is a large positive value to penalize stereo inconsistencies. Minimizing this term enforces stereo consistency between the left and right output images.

4. Experiment

We implemented our method with \(\alpha = 1\) in Eq. (1), \(\beta = 2\) in Eq. (5), and \(K = 1000\) in Eq. (16). The pixel saliencies \(S(p'_L)\) and \(S(p'_R)\) in Eq. (4) were set to zero. A 3-level hierarchical representation was used for optimization. The vertical range of the shifts was set to \(+/-4\) pixels.

We tested our method with the “Tsukuba” sequence from the Middlebury stereo dataset [13], which contains five horizontal viewpoints. We used the leftmost and central images as the left and right input images, which are shown in Fig. 2(a). The image sizes were 358 \(\times\) 252 pixels. The ground-truth disparity contained in the dataset was used to define the stereo correspondences in the input images.

We compared several methods for resizing the image pair to a 75% width without changing the height. We applied the stereo matching method [14], which is available in OpenCV 2.1, to the resulting image pair to obtain a disparity map. This disparity map is expected to be similar to the one obtained from the input image pair because the underlying scene structure should be unchanged before and after
Figures 2 (b)–(e) show the results of our experiment. (b) is the result of linear scaling, where the entire scene was uniformly shrunk in the horizontal direction. The disparity values were also shrunk to 75% of the input disparities. (c) is the case where the left and right images were independently resized by shift-map editing without a stereo correspondence constraint. Each of the resulting images looks natural without visible distortions, but the scene structure was entirely destroyed, as can be seen from the disparity map between the resulting images. (d) is the result of our method, where the resulting images were natural, similar to those in (c). Furthermore, the resulting disparity map indicates that the entire scene structure and the disparity values remained unchanged after resizing. This result proves the effectiveness of our stereo correspondence constraint. (e) shows the result of stereo seam carving [9], in which stereo correspondences are also cared in a different manner. However, due to the nature of seam carving, visible distortions (i.e. some straight lines are curved) were found in the resulting images.

5. Conclusion

We proposed a new stereo image retargeting method based on the framework of shift-map image editing. We integrated a stereo correspondence constraint into the shift-map framework. Our experimental results show that this constraint is effective in preserving the underlying scene structure and producing natural-looking results.

Our future work is to extend other image editing operations to stereo images. Various image editing operations such as reshuffling, inpainting, and image composition can be handled with the shift-map editing framework, and our stereo correspondence constraint would also be applicable to these operations. We also plan to extend our method to handle user-specified constraints such as depth control of each object.

Acknowledgments

This study was conducted under an extramural research contract of the National Institute of Information and Communications Technology (NICT).

References

[1] A. Enomoto and H. Saito, “Diminished reality using multiple handheld cameras,” ACCV Workshop on Multi-dimensional and Multi-view Image Processing, pp.130–135, 2007.
[2] C. Manders, F. Farbiz, T.K. Yin, Y. Miaolong, B. Chong, and C.G. Guan, “Interacting with 3D objects in a virtual environment using an intuitive gesture system,” 7th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and Its Applications in Industry, pp.1–5, 2008.
[3] S. Avidan and A. Shamir, “Seam carving for content-aware image resizing,” ACM Trans. Graphics, vol.26, no.3, 2007.
[4] M. Rubinstein, A. Shamir, and S. Avidan, “Improved seam carving for video retargeting,” ACM Trans. Graphics, vol.27, no.3, pp.1–9, 2008.
[5] T.S. Cho, M. Butman, S. Avidan, and W.T. Freeman, “The patch transform and its applications to image editing,” IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp.1–8, 2008.
[6] C. Barnes, E. Shechtman, A. Finkelstein, and D.B. Goldman, “PatchMatch: A randomized correspondence algorithm for structural image editing,” ACM Trans. Graphics, vol.28, no.3, Aug. 2009.
[7] Y. Pritch, E. Kav-Venaki, and S. Peleg, “Shift-map image editing,” International Conference on Computer Vision, pp.151–158, Sept. 2009.
[8] Y. Hu and D. Rajan, “Hybrid shift map for video retargeting,” IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2010.
[9] K. Utsugi, T. Shibahara, T. Koike, K. Takahashi, and T. Naemura, “Seam carving for stereo images,” 3DTV Conference, 2010.
[10] Y. Boykov, O. Veksler, and R. Zabih, “Fast approximate energy minimization via graph cuts,” IEEE Trans. Pattern Anal. Mach. Intell., vol.23, no.11, pp.1222–1239, 2001.
[11] V. Kolmogorov and R. Zabih, “What energy functions can be minimized via graph cuts?,” IEEE Trans. Pattern Anal. Mach. Intell., vol.26, no.2, pp.147–159, 2004.
[12] Y. Boykov and V. Kolmogorov, “An experimental comparison of min-cut/max-flow algorithms for energy minimization in computer vision,” IEEE Trans. Pattern Anal. Mach. Intell., vol.26, no.9, pp.1124–1137, 2004.
[13] D. Scharstein and R. Szeliski, “A taxonomy and evaluation of dense two-frame stereo correspondence algorithms,” Int. J. Comput. Vis., vol.47, no.1-3, pp.7–42, 2002.
[14] V. Kolmogorov and R. Zabih, “Computing visual correspondence with occlusions using graph cuts,” International Conference on Computer Vision, pp.508–515, 2001.