Keystone Correction for Stereoscopic Cinematography

Feng Liu, Yuzhen Niu
Portland State University
{fliu,yuzhen}@cs.pdx.edu

Hailin Jin
Adobe Systems Inc.
hljin@adobe.com

Abstract

Keystone distortion is a long-standing problem in stereoscopic cinematography. Keystone distortion occurs when a stereoscopic camera toes in to achieve a desirable disparity distribution. One particular problem from keystone distortion is vertical disparity, which often compromises stereoscopic 3D viewing experience. Keystone distortion can be corrected by applying a proper homography; however, this damages the desirable disparity distribution. This paper presents an approach to keystone correction for stereoscopic cinematography that both corrects keystone distortion and preserves the original disparity distribution. Our method formulates keystone correction as a spatially-varying warping problem. Our method eliminates the vertical disparities and preserves the original horizontal disparities by encoding them as data terms in the warping problem. The energy terms are designed to be quadratic and thus the keystone correction problem can be quickly solved using a sparse linear solver. Our experiment shows that our method can effectively solve the keystone problem while preserving desirable horizontal disparities.

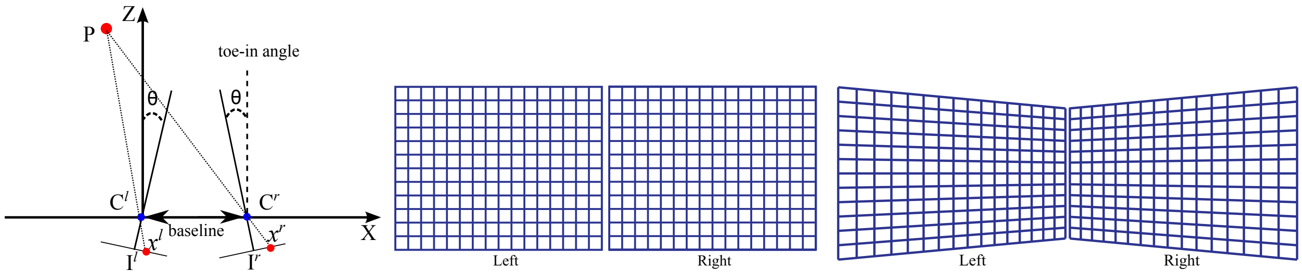
1. Introduction

Stereoscopic images and videos provide an immersive viewing experience by invoking stereo depth cues. They used to require special equipment to capture and display and therefore were mostly created by professionals. In recent years, the increasing availability of stereoscopic cameras and displays make it easy for consumers to capture stereoscopic content. There arises a problem that stereoscopic content, when captured poorly, may even pose a health risk on a viewer. Vision researchers and professional cinematographers have warned about the "3D fatigue" problem in viewing poorly captured stereoscopic images and videos.

A unique challenge in capturing and editing stereoscopic content is disparity adjustment between the left and right view. Professional film-makers often slightly adjust the camera baseline and toe-in the cameras to achieve an optimal disparity distribution. In this way, "vergence-accommodation conflicts" can be minimized and "window violations" are avoided. Typically, the disparity in the region of attention is small to relieve audiences' eyestrain. All these take sophisticated equipment, great effort, and experience. In contrast, the disparity distribution in an amateur video is not as optimal. Some consumer stereoscopic camcorders, even do not provide online manual disparity control during shooting a video.

There are two popular ways for disparity adjustment in the practice of stereoscopic cinematography, namely horizontal image translation (HIT) and camera toe-in [1]. HIT horizontally shifts the left and right images. It has the advantage that no undesirable vertical disparity will be introduced but is often insufficient to produce an optimal disparity distribution [10] and sometimes is limited by practical considerations [1]. In contrast, camera toe-in, sometimes combined with baseline adjustment, provides a more flexible approach to disparity adjustment. However, camera toe-in, as shown in Figure 1, causes keystone distortion, particularly vertical disparities, which often compromise the 3D viewing experience.

Keystone distortion can be corrected by applying a proper homography to each view of a stereoscopic image or video [14]. However, this will cancel the toe-in effect and damage the desirable disparity distribution. In this paper, we present a keystone correction method that both eliminates the undesirable vertical disparities and preserves the desirable (horizontal) disparities at the same time. We first estimate feature correspondences between the left and right view of a stereoscopic image/video. We then eliminate the vertical disparity between each pair of feature points by assigning the average vertical coordinate to both points. Meanwhile, we keep the horizontal disparities unchanged. As there is no global image warping that can simultaneously meet these two goals, we resort to a spatially-varying warping method [8, 10, 15]. Specifically, we divide each input image into a uniform grid mesh and formulate the image warping as a mesh warping problem. We encode these position constraints as data terms. The energy terms are care-



(a) Camera toe-in illustration (b) No keystone distortion ($\theta = 0$) (c) Keystone distortion ($\theta > 0$)

Figure 1. Stereoscopic camera and keystone distortion. When a 3D scene point **P** is projected onto the left and right camera, its left and right x-coordinate are different. Their difference $x^r - x^l$ is referred to as (horizontal) disparity. A stereoscopic camera system has two important parameters. The first is the baseline between the two camera optical centers and the second is the toe-in angle θ. Camera toe-in, sometimes combined with baseline adjustment, can be used to achieve a good disparity distribution. However, camera toe-in leads to keystone effect (c). The resulting vertical disparities are particularly problematic.

fully defined to be quadratic. The warping problem is thus quadratic and is solved using a standard linear solver.

The main contribution of this paper is a novel approach to keystone correction that both eliminates the vertical disparities and preserves the desirable horizontal disparities. In this paper, we show how this method can be used to correct keystone distortion for filmed stereoscopic images or to support disparity editing by changing the stereo camera settings.

The rest of this paper is organized as follows. We first briefly overview related work on keystone correction in Section 2. We then describe our method in Section 3. We discuss the evaluation of our method in Section 4 and conclude the paper in the last section.

2. Related Work

Keystone is a classic problem in computer vision and graphics. In the research on projectors, keystone is known as the distortion that occurs when an image is projected onto a surface that is not perfectly perpendicular to the optical axis of a projector. This keystone distortion can be corrected by finding and applying an optimal homography that accounts for the 3D rotation of the lens [2, 11–14, 19, 24]. Keystone correction has been a common feature available in most modern projectors.

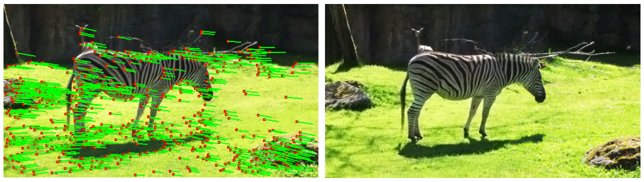
In stereoscopic cinematography, keystone occurs when a stereoscopic camera is not perfectly rectified. Specially, the optical axes of the left and right camera are not parallel to each other, as shown in Figure 1 (a). One particular problem with keystone distortion is vertical disparities, which often cause "3D fatigue" [17]. Keystone limits the capability of stereoscopic film-makers in adjusting the toe-in angle to achieve a desirable disparity distribution. Keystone has been addressed in some stereoscopic photo and video editing software. For example, *StereoPhoto Maker*[1] adopts the keystone correction methods designed for projectors. These methods, however, correct keystone distortion by canceling the toe-in effect totally, including undoing the desirable (horizontal) disparity distribution. *Nuke*[2], together with its plug-in *Ocula*[3], corrects vertical disparities by a range of 2D full-frame transformations and a vertical skew transformation (Since these algorithms are proprietary, it is unclear by reading the manuals whether the skew transformation is local or global). This paper describes the first approach to keystone correction that preserves the desirable disparity distribution and eliminates vertical disparities simultaneously.

Our method applies spatially-varying warping to keystone correction. Spatially-varying warping has been applied to a variety of computer graphics and vision applications, such as shape manipulation [8], multimedia retargeting [4, 26], video stabilization [15], and disparity editing [10]. We use this method to solve the keystone distortion problem, which has been a long-standing problem in stereoscopic photography and cinematography.

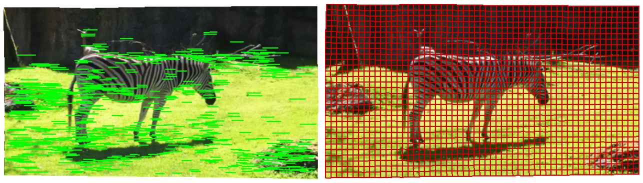
3. Keystone Correction

In a perfectly rectified stereoscopic camera, the optical axes of its left camera and right camera are parallel to each other and perpendicular to the baseline. In practice, stereoscopic cinematographers, sometimes with the help of a software system [7, 9, 27], toe-in the cameras to adjust the disparity distribution for a pleasant stereoscopic viewing experience. Camera toe-in, however, leads to keystone distortion. One particular problem with keystone distortion is vertical disparities. The challenge is how to eliminate the prob-

[1] http://stereo.jpn.org/eng/stphmkr/
[2] http://www.thefoundry.co.uk/products/nuke/
[3] http://www.thefoundry.co.uk/products/ocula/



(a) Input: left image with disparity and right image



(b) Keystone correction result: left with disparity and right with mesh

(c) Keystone correction result (cropped)

Figure 2. Keystone correction. The input stereoscopic image suffers from keystone distortion and has vertical disparities (a). Our method formulates keystone correction as a mesh warping problem and eliminates the vertical disparities and preserves the horizontal disparities, as shown in (b) and (c).

lematic vertical disparities and preserve the desirable horizontal disparities. As there is no global transformation that can achieve both goals, we use a spatially-varying warping method (c.f. [4, 8, 10, 15, 26]). Our approach to keystone correction consists of three steps.

1. Feature correspondence estimation. We estimate a set of sparse feature points between the left and right view of a stereoscopic image. Specifically, our method first estimates a set of matched SIFT feature points from the left image and right image [16]. For an image with size 800×450, we estimate about 500-1000 SIFT feature points. Then we use the epipolar geometrical constraint to rule out the outliers [5]. Figure 2 (a) shows an example of feature points and disparities.

2. Target feature position estimation. We compute for each feature point its output position. We keep the horizontal coordinate unchanged to preserve the desirable horizontal disparity distribution. We compute the average of the vertical coordinates of the pair of feature points and use it as the output vertical coordinate for both feature points. In this way, the vertical disparity is eliminated.

3. Image warping. We use the target feature positions to guide the warping of the left and right view of a stereoscopic image, as detailed below.

We apply a spatially-varying warping method to transform input images according to the output feature positions. Our method divides each input image into a $m \times n$ uniform grid mesh and reduces the image warping problem to a mesh warping problem. The unknowns are the coordinates of mesh vertices. Our method defines the mesh warping problem as a linear least squares problem that moves the feature points to the target positions and minimizes visual distortion. We now describe the energy terms.

Data term. Our method encourages each feature point $\mathbf{p}$ to be moved to its target position $\hat{\mathbf{p}}$. Because a feature point is not usually coincident with one of the mesh vertices, our method finds the mesh cell that $\mathbf{p}$ belongs to and represents $\mathbf{p}$ with a linear combination of the four vertices of the cell. The linear combination coefficients are computed using the inverse bilinear interpolation method [6]. Our method then defines the data term as follows.

$$E_d = \sum_{\mathbf{p}_i} (\sum w_j \hat{\mathbf{v}}_{i,j} - \hat{\mathbf{p}}_i)^2 \tag{1}$$

where $\hat{\mathbf{v}}_{i,j}$ are the vertices that enclose $\hat{\mathbf{p}}_i$, w_j is the bilinear coefficient, and $\hat{\mathbf{p}}_i$ is the target position of $\mathbf{p}_i$.

Smoothness term. To avoid geometric distortion, our method encourages to transform each mesh cell with a similarity transformation. A similarity transformation that maps (x, y) to (u, v) must satisfy the Cauchy-Riemann equations, namely $\frac{\partial u}{\partial x} = \frac{\partial v}{\partial y}$ and $\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} = 0$. Our method uses finite difference to compute the partial derivatives and applies this constraint to each vertex $\hat{\mathbf{v}}_{i,j} = (u_{i,j}, v_{i,j})$ in each cell and get the following smoothness energy term,

$$\begin{aligned} E_s = & \sum_{(u_{i,j}, v_{i,j})} (u_{i+1,j} - u_{i,j} - v_{i,j+1} + v_{i,j})^2 \\ & + (u_{i,j+1} - u_{i,j} + v_{i+1,j} - v_{i,j})^2 \end{aligned} \tag{2}$$

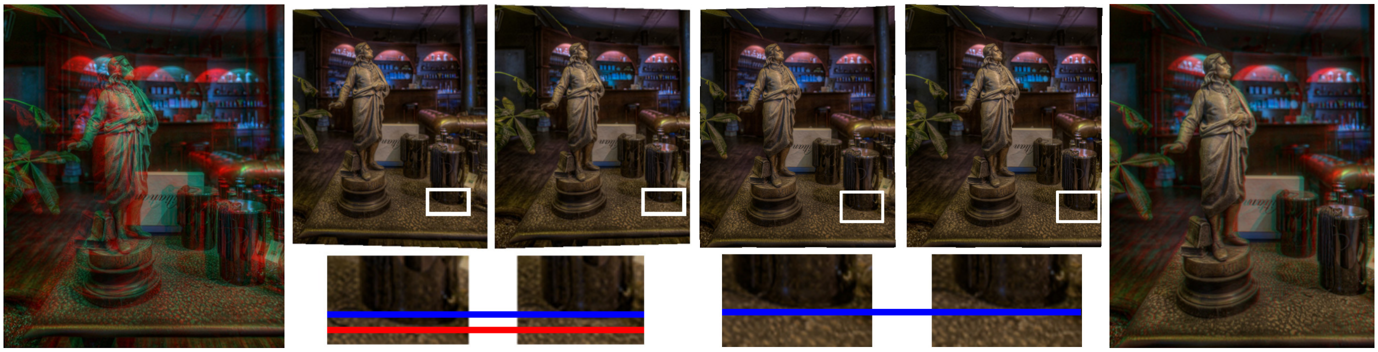
We combine the above data term and smoothness term and obtain the following linear least squares problem:

$$E = E_s + \lambda E_d, \tag{3}$$

where λ is weight with a default value 10. We solve this energy minimization problem using a sparse linear solver. Figure 2 (b) and (c) show a warping result, where the vertical disparities from the key-stone distortion are eliminated.

3.1. Application: Camera-centric Disparity Editing

Besides correcting keystone distortion in filmed stereoscopic content, our method can also support disparity editing by changing the stereo camera settings. Disparity editing has been a hot topic recently. Wang and Sawchuk presented a framework for disparity editing that either directly works on the dense disparity map or assumes known camera parameters and applies image-based rendering methods



(a) input (anaglyph) (b) toe-in angle change (left and right) (c) our result (left, right, and anaglyph)

Figure 3. Disparity editing. A user changes the toe-in angle to move the statue close to the screen. However, changing the toe-in angle introduces vertical disparities (b). Our method removes the vertical disparities by averaging the vertical positions between two images and applying a spatially-varying warping method to transform the input.

to novel view synthesis [25]. Lang *et al.* developed a series of disparity mapping tools to control the disparity distribution in a nonlinear and locally adaptive fashion [10,21]. They also directly edit the disparity map. Koppal *et al.* developed a viewer-centric editor for stereoscopic movies that provides tools for both shot planning and disparity editing, including changing the camera position and baseline [9]. Their method requires dense stereo matching and uses view interpolation to render the final result. Their method does not support the toe-in operator. We present a camera-centric disparity editing system as a useful alternative, which allows for the adjustment of both the baseline and toe-in angle and does not require dense stereo matching.

As disparity is only related to the relative pose between the left and right camera, we consider the world coordinate system whose origin is at the left camera center at each frame, as shown in Figure 1. The **Y** axis is coincident with the camera up vector and the **Z** axis is perpendicular to the plane formed by the Y axis and the baseline. Within this stereoscopic camera settings, our system to support disparity editing consists of three components.

1. We estimate the camera parameters of the right camera, namely the relative pose between the left and right camera and reconstruct a sparse set of 3D points at each video frame.

2. We obtain the new camera parameters according to the user-selected new baseline and toe-in angle and project the 3D points to the left and right image to compute the output feature positions.

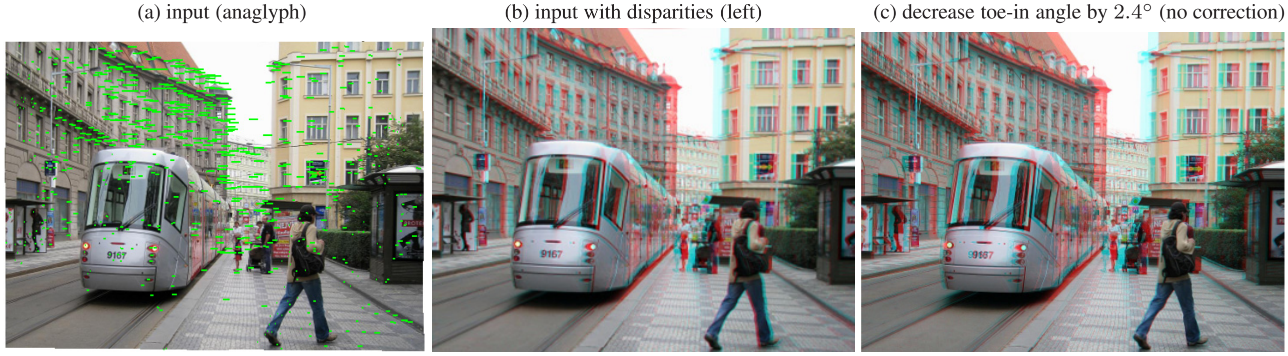
3. We finally use the warping method described above to warp each video frame. This method does not require the knowledge of camera motion over time.

Relative camera pose estimation is a well studied topic in computer vision. Our method uses the 6-point algorithm to estimate both the focal length and the relative pose between two cameras [23]. Specifically, our method first estimates a set of matched SIFT feature points from the left image and right image. Then we use the RANSAC method to repeatedly select 6 matching feature pairs, estimate the relative pose using the 6-point algorithm, and obtain a robust relative pose estimation [3]. We finally reconstruct the 3D coordinates of the feature points by triangulation [5]. One potential problem with this method is temporal incoherence since the relative pose is estimated at each time independently. We assume that the relative pose in the input video remains constant. This is typically valid for consumer-level stereoscopic camcorders. We then run the RANSAC algorithm on the feature pairs of the whole video and estimate a common relative pose for all the frames at once.

With the relative camera pose, a user adjusts the baseline and toe-in angle. Our method then evenly distributes the change to the left and right camera parameters. Our system provides visual feedback by visualizing the disparities. When the toe-in angle is changed, undesirable vertical disparities are introduced, as shown in Figure 3 (b). We solve this problem by setting the vertical coordinate of each feature point as the average of its vertical coordinate in the left and right image. We then apply the spatially-varying warping method described above to transform each input video that follows the new feature positions.

To minimize 3D fatigue, stereoscopic cinematographer often tracks an object of interest and gives it a small disparity. We support this with a key-frame based camera parameter adjustment method. Specifically, our system allows a user to adjust the baseline and toe-in angle at key frames and then automatically propagates them to neighboring frames using spline interpolation.

(a) input (anaglyph) (b) input with disparities (left) (c) decrease toe-in angle by $2.4°$ (no correction)



(d) decrease toe-in angle by $2.4°$ (left and anaglyph) (w/ correction) (e) decrease toe-in angle by $1.8°$ (w/ correction)

Figure 4. Decreasing the toe-in angle can move the train (d) or the person (e) near the screen. Toe-in angle change causes the keystone problem and introduces vertical disparities (c). Our method solves this problem (d, e).

4. Experiments

We show more examples in this section on keystone correction and disparity editing. All the stereoscopic images and videos are rendered in red-cyan anaglyph and better viewed electronically. Our method corrects keystone distortion in filmed stereoscopic images and videos automatically. Our system also allows a user to edit the disparity map by adjusting the stereoscopic camera settings. Specifically, the baseline can be used to adjust the disparity range and the toe-in angle can be used to change the location of an object with respect to the screen.

Given a stereoscopic photo shown in Figure 4 (a), a user can move the person or the train close to the screen by changing the toe-in angle. Toe-in angle change, however, causes the key-stone problem described in Section 3 and introduces vertical disparities, as shown in Figure 4 (c). Our results are free from the vertical disparities (d, e).

Our system supports disparity editing through adjusting the baseline and toe-in angle of the stereoscopic camera. Figure 5 (b) shows a result that assigns the biker near zero disparities to place him on the screen. It is created by changing the original toe-in angle by $2.1°$. This result suffers from the window violation as the car door is in front of the screen but it is cut by the screen edge [17]. The window violation can be avoided by reducing the baseline to 0.6 of the original one to position the car door close to the screen, as shown in Figure 5 (c).

Figure 6 shows several frames of a video where a boat with passengers moving towards the camera. Our system allows a user to adjust the camera parameters at key frames and then use spline interpolation to propagate the camera parameters. As a result, the boat and its passengers are consistently placed near the screen (with small disparities).

Our algorithm for keystone correction does not require any camera parameter estimation or 3D reconstruction. The application of disparity editing needs this information. Our algorithm for estimating relative poses occasionally gives poor results on still stereoscopic images. It is particularly true for scenes where feature points are far compared to the baseline between the two cameras. This is a well-known limitation in computer vision. The estimated focal length is often over-exaggerated. This, however, will only downgrade our algorithm to uniform horizontal shift. Moreover, when our algorithm runs on a stereoscopic video, it usually does not suffer from this problem, because we effectively combine feature correspondences throughout the entire video. For all the videos we have tested, there are always matching points that are sufficiently close to the camera to yield good relative pose estimates.

Our method processes each frame independently. This usually does not introduce temporal incoherence as the feature point set remains similar at neighboring frames. Occasionally, when there is drastic camera or object motion,

(a) input stereoscopic image (b) baseline: 1.0; toe-in angle change 2.1° (c) baseline: 0.6; toe-in angle change 0.7°

Figure 5. Camera-oriented disparity editing. Our system supports disparity editing through adjusting the baseline and toe-in angle of the stereoscopic camera. (b) shows a result that places the biker on the screen by assigning the corresponding region near zero disparities. It is created by a virtual camera that changes the original toe-in angle by 2.1° and keeps the original baseline. This result suffers from the window violation as the car door is in front of the screen but it is cut by the screen edge. (c) avoids the window violation by reducing the baseline to 0.6 of the original one to position the car door close to the screen.

our method sometimes introduces slightly temporal incoherence. We expect that this problem can be alleviated in future by weighting feature points that last long more than those last shortly. The final results have to be cropped as the warping results have non-rectangular boundaries. This is a challenging problem given the state of the art in in-painting research.

Our method currently relies on sparse feature points to remove vertical disparities and edit horizontal disparities. When there is no feature point in a region, our method cannot exactly achieve a desirable disparity map. This problem can be addressed by using more correspondences other than feature points, such as line correspondences [22] and dense disparity maps [18, 20].

The computational cost of our method contains two parts: relative pose estimation and warping. Relative pose estimation is slow as both the sparse feature matching step and the actual relative pose estimation step are slow. However, our method only needs to perform it once as a preprocessing step. (Note, keystone correction alone does not require relative pose estimation and only requires feature matching). Video warping time depends on the number of vertices in each frame. In our implementation, the warping step achieves 10 fps in transforming a 64×36 mesh.

5. Conclusions

In this paper, we present an approach to keystone correction for stereoscopic images and videos. Our approach eliminates problematic vertical disparities and preserves desirable horizontal disparities simultaneously. We also show how this approach enables a system that allows a user to simulate stereoscopic photography and cinematography on filmed stereoscopic images and videos. This system supports a user to adjust the baseline and toe-in angle to achieve an optimal disparity map. This system only requires to estimate the relative pose between the left and right camera. We show that this system provides a convenient way for users to edit disparity maps.

Acknowledgements

We would like to thank the reviewers for their constructive comments. We would also like to thank Flickr users, -ytf- and Dan(aka firrs), for letting us use their photos. This work was supported by the Portland State University Faculty Enhancement Grant.

References

[1] R. S. Allison. The camera convergence problem revisited. In *Stereoscopic Displays and Virtual Reality Systems XI*, pages 167–178, 2004. 1

[2] P. Beardsley, J. Van Baar, R. Raskar, and C. Forlines. Interaction using a handheld projector. *IEEE Computer Graphics and Applications*, 25(1):39–43, 2005. 2

[3] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24:381–395, 1981. 4

[4] R. Gal, O. Sorkine, and D. Cohen-Or. Feature-aware texturing. In *Proceedings of the 17th Eurographics Symposium on Rendering*, pages 297–303, 2006. 2, 3

[5] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000. 3, 4

Figure 6. Video example. Left: input. Right: our results. In this example, the boat and its passengers are consistently placed near the screen. This result is created by adjusting the camera parameters at key frames.

[6] P. S. Heckbert. Fundamentals of texture mapping and image warping. Technical report, UC Berkeley, 1989. 3

[7] S. Heinzle, P. Greisen, D. Gallup, C. Chen, D. Saner, A. Smolic, A. Burg, W. Matusik, and M. Gross. Computational stereo camera system with programmable control loop. *ACM Trans. Graph.*, 30:94:1–94:10, 2011. 2

[8] T. Igarashi, T. Moscovich, and J. F. Hughes. As-rigid-as-possible shape manipulation. *ACM Transactions on Graphics*, 24(3):1134–1141, 2005. 1, 2, 3

[9] S. Koppal, C. Zitnick, M. Cohen, S. B. Kang, B. Ressler, and A. Colburn. A viewer-centric editor for 3d movies. *IEEE Computer Graphics and Applications*, 31(1):20 –35, 2011. 2, 4

[10] M. Lang, A. Hornung, O. Wang, S. Poulakos, A. Smolic, and M. Gross. Nonlinear disparity mapping for stereoscopic 3d. *ACM Trans. on Graphics*, 29(4), 2010. 1, 2, 3, 4

[11] J. C. Lee, P. H. Dietz, D. Maynes-Aminzade, R. Raskar, and S. E. Hudson. Automatic projector calibration with embedded light sensors. In *Proceedings of the 17th annual ACM symposium on User interface software and technology*, pages 123–126, 2004. 2

[12] M. Leung, K. Lee, K. Wong, and M. Chang. A projector-based movable hand-held display system. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1109–1114, 2009. 2

[13] B. Li and I. Sezan. Automatic keystone correction for smart projectors with embedded camera. In *IEEE International Conference on Image Processing*, pages 2829–2832, 2004. 2

[14] Z. Li, K. Wong, Y. Gong, and M. Chang. An effective method for movable projector keystone correction. *IEEE Transactions on Multimedia*, 13(1):155–160, 2011. 1, 2

[15] F. Liu, M. Gleicher, H. Jin, and A. Agarwala. Content-preserving warps for 3D video stabilization. *ACM Trans. Graph.*, 28(3), 2009. 1, 2, 3

[16] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004. 3

[17] B. Mendiburu. *3D Movie Making: Stereoscopic Digital Cinema from Script to Screen*. Focal Press, 2009. 2, 5

[18] A. S. Ogale and Y. Aloimonos. Shape and the stereo correspondence problem. *Int. J. Comput. Vision*, 65(3):147–162, 2005. 6

[19] R. Raskar and P. Beardsley. A self-correcting projector. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 504–508. IEEE, 2001. 2

[20] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *Int. J. Comput. Vision*, 47(1-3):7–42, 2002. 6

[21] A. Smolic, S. Poulakos, S. Heinzle, P. Greisen, M. Lang, A. Hornung, M. Farre, N. Stefanoski, O. Wang, L. Schnyder, R. Monroy, and M. Gross. Disparity-aware stereo 3d production tools. In *European Conference on Visual Media Production*, 2011. 4

[22] M. E. Spetsakis and J. Aloimon. Closed form solution to the structure from motion problem from line correspondences. In *Proceedings of the sixth National conference on Artificial intelligence*, pages 738–743, 1987. 6

[23] H. Stewenius, D. Nister, F. Kahl, and F. Schaffalitzky. A minimal solution for relative pose with unknown focal length. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 789–794, 2005. 4

[24] R. Sukthankar, R. Stockton, and M. Mullin. Smarter presentations: Exploiting homography in camera-projector systems. In *Proceedings of the IEEE International Conference on Computer Vision*, volume 1, pages 247–253, 2001. 2

[25] C. Wang and A. A. Sawchuk. Disparity manipulation for stereo images and video. In *Proc. SPIE, Vol. 6803*, 2008. 4

[26] Y.-S. Wang, C.-L. Tai, O. Sorkine, and T.-Y. Lee. Optimized scale-and-stretch for image resizing. *ACM Trans. Graph.*, 27(5), 2008. 2, 3

[27] F. Zilly, M. Muller, P. Eisert, and P. Kauff. The stereoscopic analyzeran image-based assistance tool for stereo shooting and 3d production. In *IEEE International Conference on Image Processing*, pages 4029–4032, 2010. 2