

Multiple View Integration and Display Using Virtual Mirrors

Carmen E. Au and James J. Clark
 McGill University
 Centre for Intelligent Machines
 3480 University Street, Montreal, Canada
 {au, clark}@cim.mcgill.ca

Abstract

This paper describes a technique, called V-Mirroring, for integrating videos taken from different cameras with different viewpoints of the same scene. The term V-Mirroring stems from the use of virtual mirrors in order to composite videos together. These mirrors are placed in the scene, near to the locations of the cameras. Thereafter, for any given camera, its corresponding video is overlaid with the remaining videos in the locations of the virtual mirrors closest to their respective cameras. Thus, the objects in the scene that are imaged by more than one camera, can then be viewed from multiple viewpoints in a single video. Previous approaches for compositing images or videos, such as panorama mosaicing, require that the input videos' image planes lie on the same, or approximately the same 2D plane, thereby losing the 3D feeling of the environment. In this work, videos can be taken from very different viewpoints and still be combined into a single video containing the differing videos.

1. Introduction

In an age where cameras have largely become ubiquitous, algorithms for representing the plethora of captured images or videos in a meaningful way have become essential. Most major events are now documented by at least one camera. Multiple videos of a given rock concert, for example, are often uploaded to video sharing websites within mere minutes after the concert's end. The paparazzi and adoring fans follow celebrities around with cameras constantly documenting their every move with images and videos. In most retail stores or banks, several surveillance cameras are strategically placed to cast a watchful eye on would-be thieves. In all three examples, the resulting product is many disparate images and videos of the same scene, person or object. The benefits of having many images or videos of a common event captured from unique view-

points, however, are not reaped by the person viewing the images or videos, as they remain disjointed and can only be viewed as separate entities. Our goal is to combine similar videos, acquired from different cameras but of the same scene, into a single video.



Figure 1. Example paparazzi images with virtual mirroring: (a) image from principal camera, (b) image from secondary camera, and (c) image from principal camera with image from secondary camera overlaid onto virtual mirror.

By combining similar videos, viewers of the videos can

gain better understanding of how the videos relate to each other spatially. In other words, the viewer would be able to identify the position of the cameras in relation to one another, and thus, they will also be able to get a more three-dimensional representation of the scene, object or person. Thus, combining similar videos will enable a more immersive “being there” experience than otherwise experienced with multiple disparate videos.

One highly explored group of methods for combining similar videos or images is image stitching. The goal of image stitching is to combine many images into a composite image, such as a panorama. The images must be properly aligned and the seams between the images must not be visible, in other words, they must be well-blended. There are different approaches to solving the image stitching problem. Brown and Lowe categorize the methods into two main categories: direct and feature-based methods [16]. In direct methods [3, 12], the camera parameters are iteratively estimated by minimizing the difference in pixel intensity in the area of overlap. In feature-based methods [4, 10, 14, 20], a correspondence between points in the images is established using either low-level features, such as corners and edges, or high level features such as parts of objects. Most of these methods, however, are not robust to changes in illumination, zoom and rotation. In order to develop techniques that are in fact robust to the aforementioned changes, a type of features called invariant features was developed [2, 22]. In [16], Brown and Lowe describe their image stitching technique which use Lowe’s Scale Invariant Feature Transform (SIFT features) [17].

The quality of the resulting panorama is highly dependent on the type of images acquired for all of these techniques. Thus, image acquisition plays a pivotal role in image stitching. There are three main ways of acquiring images [7]:

1. **Camera Rotation:** The camera is rotated around an axis perpendicular to the optical axis and which passes through the optical center of the camera. Rotations in an axis other than the specified axis lower the quality of the panorama. Moreover, since the images are to be combined into a single two-dimensional (2D) image, but each image is taken on a different 2D plane, the images need to be projected onto a single surface, such as a cylinder or a sphere. Projection, however, degrades the quality of a panorama.
2. **Camera Translation:** The camera is mounted on a sliding plate and translated in a direction parallel to the image plane. One main disadvantage to this approach of image acquisition is that the camera translation must remain in the direction parallel to the image plane, otherwise the size of the objects may vary, which causes problems when stitching the images together. For ex-

ample, if the camera were to translate slightly closer to the objects of interest, those objects would appear larger in the subsequent pictures than in the previous ones and stitching these images together would be very difficult.

3. **Hand held Camera:** The camera is held by the user and is moved in a direction approximately parallel to the image plane. Naturally in this approach, the amount of undesirable translations and rotations is considerable, thus making the images acquired difficult to stitch together. The amount of overlap area between images must be increased in order to reduce the amount of inconsistencies.

Few techniques use the camera rotation method to acquire their data sets since there are limited applications that would benefit from this method of image acquisition. Thus, most image acquisition is done by translating the camera in a direction parallel to the image plane, with the camera either mounted on a sliding plate of sorts or held by hand. In fact all of the aforementioned image stitching techniques excluding one notable exception [2], require that the camera be moved along the same direction as the image plane. Since all the acquired images are either on, or approximately on the same plane, the resulting panoramic images created do not create the feeling of looking into a three-dimensional environment. By limiting the amount of viewpoint change, the images that can be acquired are limited and thus the number of possible applications for these techniques are also limited. Even in Baumberg’s approach [2], which allows for viewpoint changes and rotations, the maximum amount of rotation before the results begin to degrade is around 51 degrees. In our approach, we can use several cameras positioned along several viewpoints with an unlimited amount of relative rotation between the cameras.

Moreover, for these image stitching techniques, the amount of overlap of the photos significantly affects the quality of the resulting composite image. In fact, as Chen asserted [8], an overlap of approximately 50% is desirable in order to have the best results. If there is no overlap, then there is a risk of having holes in the lightfield. In other words, if the scene was not properly sampled, then there can be gaps in the resulting panorama.

Beyond image stitching, a few papers have described forays into dynamic panorama mosaicing, which is the creation of panoramic videos. Many of the techniques employ systems that either have specialized panoramic cameras such as catadioptric cameras, or use a mirroring system to generate the panoramas [5, 6, 13, 18, 23, 24, 25]. Agarwala *et al.* created Panoramic Video Textures (PVTs) with a single panning video camera [1]. Likewise, Rav-Acha *et al.* [21] created Dynamosaics using a single cam-

era. In both cases, since only a portion of the scene has been imaged at any given time, the resulting panorama does not have images for the entire panorama. Thus, the PVTs and Dynamosaics can only really work for repetitive motion, such as swaying trees or waterfalls. Neither technique can create dynamic panoramas for scenes with highly aperiodic motion, such as the inside of a bank or a rock concert.

With panoramas, whether static or dynamic, once the panorama is created, the viewpoint cannot be changed. There has been some work done on using multiple cameras with different viewpoints to render new views. Levoy and Hanrahan [15] created a method that could generate new views, without depth information or feature matching called Light Field Rendering. In their approach they define the input images as 2D slices of the lightfield. From the lightfield, they can create new views by extracting slices in the appropriate directions. Gortler *et al.* [11] also use the direction of the light to generate new views, but they define a 4D function called Lumigraph, that describes the flow of light at all positions in all directions. With the Lumigraph, they are able to generate new views of an object very quickly, without geometric information of the scene or object. More recently, Zitnick *et al.* [26] use depth maps and Bayesian matting to render new views. While their approach proved to be quite robust, their technique uses a 1D arc of eight cameras where the maximum disparity between neighbouring cameras is 100 pixels. On the other hand, although our technique does not render new views, it does create a composite view of the scene where one video is overlaid over the other by interpolation and it does not limit the amount of disparity between the cameras.

2. View Integration using Virtual Mirrors

The main problem encountered when attempting to combine similar videos from different cameras with differing viewpoints lies in how to represent a 3D world in a 2D plane. Our approach draws from the idea that when a person is looking into a mirror, the objects between that person and the mirror can be seen both directly and indirectly through the mirror. In that way, more than one side of the object can be seen at the same time. Thus, we developed a virtual mirror technique, V-Mirroring, whereby for each camera, a virtual mirror is placed in the scene close to it. The virtual mirror reflects the video captured by the camera. In other words, given the video from one camera, we label this camera the “principal camera” and its corresponding video the “principal video”. The cameras whose viewpoints overlap with the viewpoint of the principal camera are labelled “secondary cameras” and each secondary camera has its own corresponding virtual mirror placed in the scene. The videos from the secondary cameras are overlaid onto the principal video at the location of their cor-

responding virtual mirror. We call these generated videos “combined videos”. The V-Mirroring technique allows the objects within a scene captured by the similar videos to be observed from more than one viewpoint in just one combined video.

The idea of V-Mirroring is further illustrated in Figure 1. The image from the secondary camera (shown in Figure 1b) is overlaid onto the image from the principal camera (shown in Figure 1a) as a virtual mirror in order to produce Figure 1c. As can be seen in this illustration, two nearly opposing views of the woman are visible in one single 2D image.

The number of combined videos is equal to the number of similar videos. Each of the similar videos can serve as principal video with the remaining videos serving as mirror videos. The virtual mirrors can then act as thumbnails whereby clicking on one of them switches the combined video so that the video overlaid on the clicked virtual mirror becomes the principal video and all remaining videos are overlaid onto it. Further details to our approach will be presented in Section 3.

3. Implementation of V-Mirroring

As described in Section 2, given several similar videos acquired by different cameras with different but overlapping viewpoints of the same scene, we combine these similar videos into a single video using V-Mirroring. In order to determine the proper pixel mapping to create the virtual mirrors, the relative positions of the cameras are required. Determining these parameters is not a trivial problem, thus in our initial implementation, we use two calibrated cameras. The setup is described in Section 3.1.

3.1. Hardware

The similar videos used in our work were taken from McGill University’s *Shared Reality Environment* (SRE) [9], a laboratory space. The laboratory setup consists of several projection screens onto which we projected a landscape to serve as a generic background. Moreover, there are two calibrated 3Com U.S. Robotics BigPicture Cameras in the SRE. One camera lies just below the east screen, facing the west screen and the other is facing the north screen. We call these cameras “Camera East” (CE) and “Camera South” (CS) respectively. A diagram of the setup is shown in Figure 2.

CE and CS were calibrated and thus the relative rotation and translation between them could be easily found. With the rotation matrices, \mathbf{R} , and the translation vectors, \mathbf{T} , of each camera, mapping the pixels of one camera onto a virtual mirror placed in the video of another camera becomes possible.

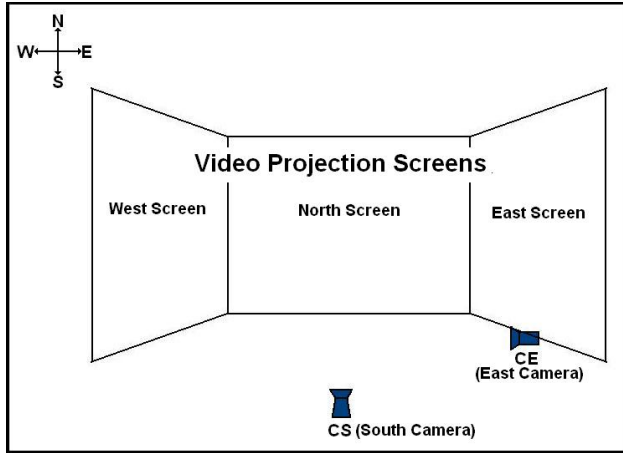


Figure 2. Shared Reality Environment setup: projection walls with calibrated cameras

3.2. Pixel Mapping

In this section, we describe how the videos of the secondary cameras are overlaid onto the principal video. Although there is no limit to the number of secondary cameras possible other than the physical limitation of the video resolution, for the purpose of clarity, our work uses only the two calibrated cameras, CE and CS. Since mapping each secondary camera to the principal video does not affect the mapping of the other secondary cameras, this method is easily extended.

Problem definition: For each point in the secondary camera image plane, find the corresponding point on the principal camera image plane.

We solve the corresponding points for each frame of the video. Each image point has 2 components, the x - (horizontal) and the y - (vertical) component and each one is solved for individually. We begin by solving for the corresponding x -components of the principal camera image plane to all the x -components of the secondary camera image plane. Thus, we project all the points onto the X - Z plane of the camera, as in Figure 3.

For each of the x -components of the secondary camera image plane we solve for the point of intersection between the ray that passes through the point itself, P_{SC} , and the optical center of the secondary camera, C_{SC} , and the virtual mirror plane. We call this point of intersection the “mirror point”, P_M . The virtual mirror plane is placed a small distance, M_X , away from the secondary camera, with the plane parallel to the secondary camera image plane. After we find the x -component of P_M , we can then find the corresponding x -component of the point on the principal cam-

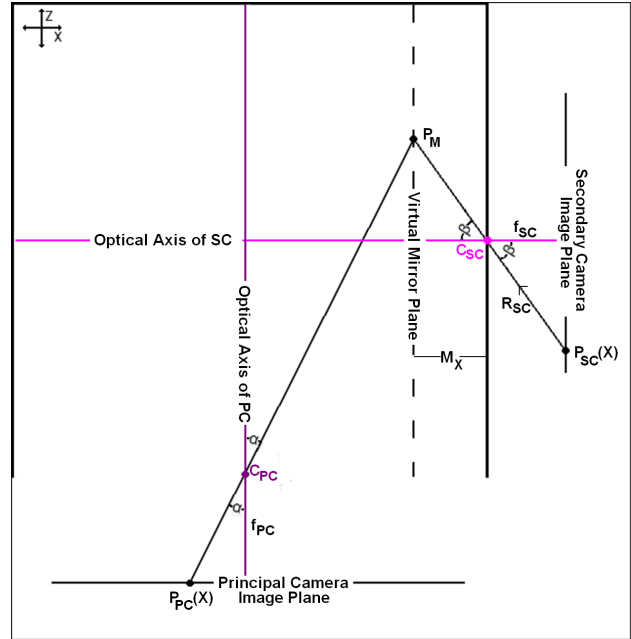


Figure 3. Diagram of Pixel Mapping. Points projected onto the X - Z plane, to determine the horizontal components of the homography. The point $P_{SC}(X)$ on the secondary camera (C_{SC}) image plane, is mapped to the point P_M on the virtual mirror plane, which is in turn mapped to the point $P_{PC}(X)$ on the principal camera (C_{PC}) image plane.

era’s image plane. The algorithm used to find these points is described in Figure 4.

Once all the horizontal components of the corresponding points on the principal camera’s image plane are found, the vertical components can be found by projecting all the points onto the Y - Z plane of the cameras. The algorithm remains as in Figure 4 by replacing the horizontal components with the vertical ones. Figure 5 shows the projection of the points onto the Y - Z plane. It is important to note that the Y - Z plane of each individual camera is not to be confused with the real-world coordinates. Thus, when the secondary camera is projected onto its Y - Z plane, the real-world projection is actually onto the X - Y plane.

With all the corresponding points, the pixels of the video of the principal camera at the virtual mirror can be replaced with the corresponding pixels from the secondary camera. This mapping, however, is a many-to-one function, meaning, more than one pixel from the secondary camera can correspond to a single pixel of the principal camera. Thus, the average of the pixels that map to the same location is taken and it is that value that is used for the virtual mirror.

Algorithm: Find $P_{PC}(x)$

1. **Assign $P_{SC}(x)$:** $P_{SC}(x)$ is a point on the secondary camera image plane in the horizontal direction. It takes on a value from 1 to w , where w is the width of the image.
2. **Find β :** β is the angle between R_{SC} and the secondary camera's optical axis, where R_{SC} is the ray passing through $P_{SC}(x)$ and C_{SC} , which is the position of the secondary camera.

$$\beta = \tan^{-1}\left(\frac{d}{f_{SC}}\right)$$

where d is the distance from P_{SC} and the center of the image, and f_{SC} is the focal length of the secondary camera.

3. **Find $P_M(x,z)$:** P_M is the point at which the ray R_{SC} passes through the virtual mirror plane. Given β , project the vector on the optical axis of the secondary camera onto the virtual mirror plane and multiply it by the magnitude of the distance between P_M and the optical axis along the virtual mirror plane.

$$adj = \sqrt{|v| \cdot |v|}$$

$$opp = adj * \tan(\beta)$$

$$P_M(x) = P_{OAVC}(x) - opp * V_M(x)$$

$$P_M(z) = P_{OAVC}(z) - opp * V_M(z)$$

where P_{OAVC} is the point where the virtual mirror's plane crosses the secondary camera's optical axis and V_M is the vector along the virtual mirror.

4. **Find α :** α is the angle between R_{PC} and the optical axis of the principal camera.

$$\alpha = \cos^{-1}\left(\frac{V_{OAPC} \cdot V_{PC}}{\|V_{OAPC}\| \|V_{PC}\|}\right)$$

where V_{OAPC} is a vector along the optical axis of C_{PC} and V_{PC} is a vector along the ray that passes through C_{PC} and P_m .

5. **Find $P_{PC}(x)$:** P_{PC} is the point on the principal camera's image plane that corresponds to P_{SC} .

$$P_{PC}(x) = mid + f_{PC} * \tan(\alpha)$$

where mid is the center of the principal camera image plane and f_{PC} is the focal length of the principal camera.

Figure 4. Algorithm to find x -component of pixel mapping

3.3. Virtual Mirrors as Thumbnails

Once the combined videos, the videos containing the virtual mirrors, are created, the virtual mirrors can serve as

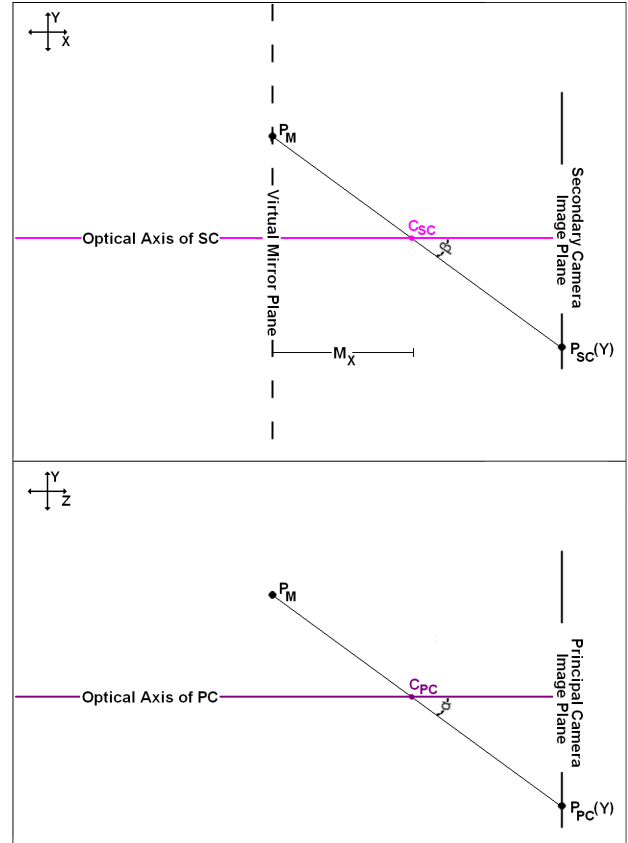


Figure 5. Diagram of Pixel Mapping. Points projected onto the Y-Z planes of the cameras, to determine the vertical component of the homography. The point $P_{SC}(Y)$ on the secondary camera (C_{SC}) image plane, is mapped to the point P_M on the virtual mirror plane, which is in turn mapped to the point $P_{PC}(Y)$ on the principal camera (C_{PC}) image plane.

thumbnails whereby clicking on them would switch to a new view with the clicked virtual mirror's camera serving as principal camera. In other words, if there are two cameras, there are two combined videos. In each of the combined videos, we have a view from the principal camera's field of view as well as the secondary camera displayed in the virtual mirror. By clicking on the virtual mirror, the view then switches, and its camera then becomes principal camera and what was the principal camera then becomes the secondary camera. In order to create our demonstration, a multimedia authoring program, Adobe Flash Professional was used to embed the video streams in a web page. The web page could then be viewed remotely by any authorized users, which is an ideal situation for video surveillance applications or for people wishing to share videos with friends.



Figure 6. Images from video with CE as the principal camera.



Figure 7. Images from video with CS as the principal camera.

4. Results

In Figure 6 and 7 images from the videos with CE and CS serving as the principal cameras are shown respectively. As can be seen in Figure 6, the view of the scene as captured by CE is displayed with a virtual mirror located in the approximate location of CS. In this case, the corresponding frame from the video captured by CS is inverted and displayed in the virtual mirror. Likewise, in Figure 7, CS is the principal camera, and the corresponding frame captured by CE is displayed in the virtual mirror.

4.1. Dealing with Occlusion

One of the main difficulties of the V-Mirroring approach is occlusion, whereby the foreground is between the virtual mirror and the principal camera's field of view. In the case of occlusion, the portions of the virtual mirror that are occluded should no longer be replaced by the corresponding pixels from the secondary camera. Thus, a technique for differentiating between foreground and background is required. In our approach, we found that a simple background subtraction method sufficed for the given videos. An image of the scene without any foreground from each of the cameras is used as the background model. Each frame is compared against its own respective background model using pixel-to-pixel subtraction. If, at the location of the virtual mirror, the frame and the background model match within a certain threshold of tolerance, then there is no occlusion. However, wherever the frame and background do not match within the location of the virtual mirror, then the pixels

are not replaced with the pixels from the secondary camera video. In Figure 8, although the foreground should be in front of the virtual mirror, without differentiating between foreground and background, the virtual mirror is replaced over the foreground rendering the person behind the mirror, and yet still "reflected" inside the virtual mirror. This occurrence, of course, is contrary to what would actually be seen had there been a real mirror located in the place of a virtual mirror. With the background subtraction method, however, as in Figure 9 the foreground is not replaced with the mirror, thus more closely reflecting what would occur in reality.

4.2. Dealing with Illumination Changes

Once the background subtraction method is used to fix the occlusion problem described in Section 4.1, then the changes in illumination becomes an issue that must be handled. The simplest approach, which is the approach we adopted, is to divide each frame by its own greatest intensity value and then multiply it by the greatest intensity value of the background model. In doing so, we manage to render the illumination of each frame closer to that of the background model.

5. Conclusions

In this paper, we have described a new method, called V-Mirroring, for integrating multiple videos of a scene, object or person in a single video. The main advantage to V-Mirroring, which differs from other techniques, is that the



Figure 8. The virtual mirror is replaced over the foreground causing the person to be behind the virtual mirror.



Figure 9. With background subtraction, the virtual mirror is not replaced in the pixels where the person occludes the mirror.

videos can be taken from different cameras and from different viewpoints. Because multiple views of a scene can be represented in a single video, the resulting combined video provides a feeling of looking into a 3D environment. Moreover, better understanding of the spatial relationships between views is achieved when the multiple videos are combined into a single video.

If we take the example of the video surveillance application, many surveillance systems consist of multiple cameras with one or more monitors displaying the videos captured by the cameras. In the case of one monitor, the displays cycle through the multiple videos, implying that at any given time, only one view can be surveyed by the guard. In the case of multiple monitors, although all the views can be displayed at once, the guard may not be able to turn his or her attention to all the monitors at once, thus situations that may warrant a response can easily go unnoticed. Moreover, with the multiple monitors, virtually no spatial relationships of one view to another is provided to the guard. With V-Mirroring, since multiple views are combined into a single video, the security guard can focus his or her attention on a single monitor, but still be able to see all the views at any given time. If more detail of a view is required, he or she can click on the corresponding virtual mirror to make that view the principal one.

One limitation of the V-Mirroring technique is that the cameras must be calibrated in order to map the virtual video onto the virtual mirror. While surveillance systems can make use of the technique, the goal is to use V-Mirroring for other applications where hand held cameras are used, such as rock concerts, paparazzi filled events, or even weddings. It is then necessary to find a technique to establish the

relative positions of the cameras such as Nistér's five-point algorithm [19].

Overall, however, the V-Mirroring technique is a simple but effective method of combining multiple similar videos into a single video.

6. Acknowledgments

We would like to thank Professor Jeremy Cooperstock of McGill University as well as his students, Wei Sun and Samuel Audet for providing the calibrated videos described in this paper. This paper was made possible in part by a generous scholarship from Precarn Incorporated.

References

- [1] A. Agarwala, K. C. Zheng, C. Pal, M. Agrawala, M. Cohen, B. Curless, D. Salesin, and R. Szeliski. Panoramic video textures. *ACM Transactions on Graphics*, 24(3):821–827, July 2005.
- [2] A. Baumberg. Reliable feature matching across widely separated views. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 774–781, 2000.
- [3] J. Beis and D. Lowe. Shape indexing using approximate nearest-neighbor search in highdimensional spaces. In *Conference Computer Vision Pattern Recognition*, pages 1000–1006, 1997.
- [4] D. Capel and A. Zisserman. Automated mosaicing with super-resolution zoom. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1998.
- [5] J. S. Chahl and M. V. Srinivasan. Reflective surfaces for panoramic imaging. *Applied Optics*, 36:8275–8285, Nov. 1997.

- [6] J. S. Chahl and M. V. Srinivasan. A complete panoramic vision system, incorporating imaging, ranging, and three dimensional navigation. In *OMNIVIS '00: Proceedings of the IEEE Workshop on Omnidirectional Vision*, page 104, Washington, DC, USA, 2000. IEEE Computer Society.
- [7] C.-Y. Chen and R. Klette. Image stitching - comparisons and new techniques. In *CAIP '99: Proceedings of the 8th International Conference on Computer Analysis of Images and Patterns*, pages 615–622, London, UK, 1999. Springer-Verlag.
- [8] S. E. Chen. QuickTime VR — an image-based approach to virtual environment navigation. *Computer Graphics*, 29(Annual Conference Series):29–38, 1995.
- [9] J. Cooperstock. Shared reality environment. <http://www.cim.mcgill.ca/sre/>.
- [10] P. Debevec, Y. Yu, and G. Boshokov. Efficient view-dependent image-based rendering with projective texture-mapping. Technical Report CSD-98-1003, 20, 1998.
- [11] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen. The lumigraph. In *SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 43–54, New York, NY, USA, 1996. ACM.
- [12] M. Irani and P. Anandan. About direct methods. In *Proceedings of the International Workshop on Vision Algorithms*, pages 267–277, London, UK, 2000. Springer-Verlag.
- [13] H. Ishiguro, K. C. Ng, R. Capella, and M. M. Trivedi. Omnidirectional image-based modeling: three approaches to approximated plenoptic representations. *Machine Vision Applications*, 14(2):94–102, 2003.
- [14] Y. Kanazawa and K. Kanatani. Image mosaicing by stratified matching. In *Workshop on Statistical Methods in Video Processing*, 2002.
- [15] M. Levoy and P. Hanrahan. Light field rendering. In *SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 31–42, New York, NY, USA, 1996. ACM.
- [16] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 20:91–110, 2003.
- [17] D. G. Lowe. Object recognition from local scale-invariant features. In *International Conference on Computer Vision*, pages 1150–1157, 1999.
- [18] S. K. Nayar. Catadioptric omnidirectional camera. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, page 482, Washington, DC, USA, 1997. IEEE Computer Society.
- [19] D. Nistér. An efficient solution to the five-point relative pose problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(6):756–777, 2004.
- [20] V. Rankov, R. J. Locke, R. J. Edens, P. R. Barber, and B. Vojnovic. An algorithm for image stitching and blending. In *Proceedings of SPIE*, vol. 5701, pages 190–199, 2005.
- [21] A. Rav-Acha, Y. Pritch, D. Lischinski, and S. Peleg. Dynamosaics: Video mosaics with non-chronological time. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 58–65, Washington, DC, USA, 2005. IEEE Computer Society.
- [22] C. Schmid and R. Mohr. Local grayvalue invariants for image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(5):530–535, 1997.
- [23] T. Svoboda and T. Pajdla. Epipolar geometry for central catadioptric cameras. *International Journal on Computer Vision*, 49(1):23–37, 2002.
- [24] K. Tan, H. Hua, and N. Ahuja. Multiview panoramic cameras using mirror pyramids. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 26(7):941–946, 2004.
- [25] S. Zimmermann and D. P. Kuban. A video pan/tilt magnify/rotate system with no moving parts for motion simulation and telepresence. In *Proceedings of the 3rd annual virtual reality conference and exhibition on VR becomes a business*, pages 181–189, Westport, CT, USA, 1993. Meckler Corporation.
- [26] C. L. Zitnick, S. B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski. High-quality video view interpolation using a layered representation. *ACM Trans. Graph.*, 23(3):600–608, 2004.