

Stereoscopic cameras for the real-time acquisition of panoramic 3D images and videos

Luis E. Gurrieri and Eric Dubois

School of Electrical Engineering and Computer Science
University of Ottawa
Ottawa, ON, Canada.

ABSTRACT

There are different panoramic techniques to produce outstanding stereoscopic panoramas of static scenes. However, a camera configuration capable to capture omnidirectional stereoscopic snapshots and videos of dynamic scenes is still a subject of research. In this paper, two multiple-camera configurations capable to produce high-quality stereoscopic panoramas in real-time are presented. Unlike existing methods, the proposed multiple-camera systems acquire all the information necessary to render stereoscopic panoramas at once. The first configuration exploits micro-stereopsis arising from a narrow baseline to produce omni-stereoscopic images. The second panoramic camera uses an extended baseline to produce poly-centric panoramas and to extract additional depth information, e.g., disparity and occlusion maps, which are used to synthesize stereoscopic views in arbitrary viewing directions. The results of emulating both cameras and the pros and cons of each set-up are presented in this paper.

Keywords: stereoscopic panoramas, panoramic cameras, omnistereo, stereoscopic cameras

1. INTRODUCTION

The advent of commercial panoramic cameras has enabled a variety of image-based applications in recent years. However, there is an aspect not sufficiently explored in the field of image-based virtual environments: the practical acquisition and rendering of stereoscopic imagery of dynamic scenes in arbitrary gazing directions. There are numerous acquisition strategies capable to produce high quality stereoscopic panoramas. Unfortunately, most of these proposals are constrained to static scenes, whereas a practical omnistereoscopic camera should be able to acquire dynamic scenes. Some existing methods can acquire two panoramic views of the scene with vertical parallax, but this strategy is suboptimal when rendering stereoscopic views suitable to stimulate human binocular stereopsis.

The problem can be defined as how to acquire the necessary information of the scene, from a single viewing point, to be able to synthesize stereoscopic views suitable for stimulating human binocular stereopsis, and to do so in any gazing direction around the capture point. In order to satisfy the constraints of the problem, it is necessary to propose a panoramic camera capable to acquire of all the scene's necessary visual information to reconstruct stereoscopic views in arbitrary directions at once. One of the immediate considerations is how to keep a consistent illusion of depth in every direction, and how to do this using the sampled visual information of the scene.

In this paper, we propose using a multiple-camera approach to acquire a number of partially overlapped views of the scene. One possible approach is to mosaic these stereoscopic images to produce novel views in arbitrary gazing directions. An alternative technique involves the acquisition of a panoramic view of the scene and its depth map to synthesize novel stereoscopic views in any direction. We introduce in this paper the panoramic cameras suitable for both rendering strategies and the results of emulating these two schemes.

Further author information: (Send correspondence to Luis E. Gurrieri)

Luis E. Gurrieri: E-mail: Luis.Gurrieri@uOttawa.ca, Website: <http://www.LuisGurrieri.net/>

Eric Dubois: E-mail: edubois@uottawa.ca, Website: <http://www.eecs.uottawa.ca/~edubois/>

Luis E. Gurrieri and Eric Dubois "Stereoscopic cameras for the real-time acquisition of panoramic 3D images and videos", Proc. SPIE 8648, Stereoscopic Displays and Applications XXIV, 86481W (March 12, 2013); doi:10.1117/12.2002129
Copyright 2013 Society of Photo-Optical Instrumentation Engineers. One print or electronic copy may be made for personal use only. Systematic electronic or print reproduction and distribution, duplication of any material in this paper for a fee or for commercial purposes, or modification of the content of the paper are prohibited.
<http://dx.doi.org/10.1117/12.2002129>

1.1 The need for an alternative omnistereoscopic camera

Real-world scenes are intrinsically dynamic and the majority of the panoramic cameras and rendering techniques proposed in the last decade are inadequate to capture omnistereoscopic snapshots of such events. Recent multiple-camera configurations promise solutions for this problem, but their limitations and benefits to solve the problem of real-time omnistereoscopic acquisition needs to be studied. Although the improvements by adding stereoscopic vision are evident, a panoramic camera capable to capture omni-stereoscopic snapshots is commercially unavailable to the authors' best knowledge.

The commercial development of single-snapshot panoramic cameras has already facilitated the popularization of image-based virtual tours and virtual navigation applications, i.e. Google StreetView, MapJack and EveryScape to mention some of them. These virtual tours need the acquisition of large number of panoramic snapshots of the scene to facilitate the free navigation in image-based simulations of a remote world locations.¹ A camera mounted on a mobile platform, e.g. cameras mounted in cars, bicycles, or robots, are generally used to capture a large number of panoramic snapshots. Enhanced stereoscopic virtual tours can be made in similar manner, if a practical method to acquire large numbers of omni-stereoscopic samples were available.

The addition of stereoscopic visualization will improve the realism of image-based simulations, e.g. image-based backgrounds for stereoscopic video production, training simulations using stereoscopic snapshots of remote locations, educational applications, stereoscopic telepresence, etc. The benefits of omni-stereoscopic media can be extended to other applications such as free-viewpoint TV,² which can use an omni-stereoscopic camera to provide interactive 3DTV to the viewer, with the additional capability to navigate the scene at will.

The *stereoscopic budget* of a particular scene is a term coined in the 3D film production which involves adjusting the baseline of the stereoscopic camera rig to satisfy different depth effects. For instance, on indoor scenarios the stereoscopic baseline b may be smaller than the interocular distance in adults to provide a more interesting depth effect (*hypo-stereo*), while in outdoor scenes, b may be chosen larger than the average interocular distance (*hyper-stereo*). It would be desirable improvement to be able to adjust the stereoscopic budget of an omnistereoscopic video in postproduction.

1.2 Proposed solutions in the literature

Over the last decade, problems such as robot navigation and omnidirectional 3D scene recovery originated strategies for omnistereoscopic depth map acquisition.^{3,4} Although attractive for their simplicity, these methods are in general inadequate to simulate binocular human stereopsis since they can only in estimate vertical disparities.^{5,6} Occlusion handling and other problems arise when trying to adapt these techniques to create omnistereoscopic views of the scene.

Other omnistereoscopic cameras based on catadioptric systems, e.g. lenses and mirrors, are suitable for the task. However, these complex configurations, which are in general inflexible to change the stereoscopic budget, can not produce stereoscopic imagery suitable for human viewing. In some of them, a limited stereoscopic baseline adjustment is possible, but the whole arrangement must be subject to re-calibration and the scene depth budget is limited by the camera configuration used during the acquisition.⁷

Other proposals are suitable to emulate horizontal binocular stereopsis in every direction, but they are based on lengthy sequential acquisition methods.⁸⁻¹¹ These methods were conceived for omnistereoscopic sampling of static environments. Although the parallelization of the acquisition process has been proposed,¹² the number of cameras necessary render the idea impractical.

In more recent times, the interest in new immersive and interactive media has motivated several proposals suitable to capture real-time omnistereoscopic imagery. Some recent proposals based on multi-camera arrangements, were conceived specifically to satisfy the dynamic scene sampling constraint. These configurations mostly come from patent applications^{13,14} and there not enough theoretical studies behind these designs, nor there are good experimental rendering results to support them.

An omnistereoscopic camera based on multiple cameras introduces new unsolved challenges in terms of rendering. For instance, the parallax between projection centers limits the camera's use to scenarios where the foreground in the scene is far from the camera. Paradoxically, the human binocular perception works by exploiting the horizontal disparity between elements in the foreground of the scene as seen by each eye, being secondary for the backgrounds scene. Therefore, the farther the elements in the scene are, the weaker is our binocular perception of depth. In background scenes, other depth perception mechanisms are predominant, i.e. perspective, blurring, shading, and even colour contrast. Hence, the

use of certain multiple-camera configurations for omnistereoscopic photography is marginal. More studies are required to understand the limitations and advantages of different multi-cameras configurations.

2. THE OMNISTEREOSCOPIC VIEWING FUNCTION

The visual information to be acquired by an omnidirectional sensor can be modeled by the *viewing function*, which is an array of real valued functions $C = \{c_1, c_2, c_3\}$. Each function c_i models a color component in a determined color space, e.g., typically a trichromatic color space, as real-valued function. More specifically, the viewing function C describes what a sensor located in the position in space $\mathbf{r} \in \mathbb{R}^3$ would measure for a gazing direction $(\theta, \phi) \in S^2$, where S^2 is the 2-sphere. This function depends also of the time t , therefore, the sensor should integrate each color component c_i over Δt , which is the sensor's exposition time. $C : S^2 \times \mathbb{R}^3 \times \mathbb{R} \rightarrow \mathbb{R}^3$ as a vector quantity

$$C(\mathbf{u}) = (c_1(\mathbf{u}), c_2(\mathbf{u}), c_3(\mathbf{u}))^T, \quad (1)$$

where the input is the array of parameters $\mathbf{u} = (\theta, \phi, \mathbf{r}, t)^T$.

An example of the acquisition process is illustrated in Fig. 1. In this case, a camera (sensor) located at \mathbf{r}_o is sampling the visual field over a limited field-of-view (FOV) by integrating the tri-chromatic values c_i over an exposition time Δt . The sampling is done in the gazing direction (θ_o, ϕ_o) and over the FOV: $(\Delta\theta, \Delta\phi)$. The camera samples C to produce an bi-dimensional image $I : Z^2 \rightarrow \mathbb{R}^3$, which can be resampled into a planar format producing I' .

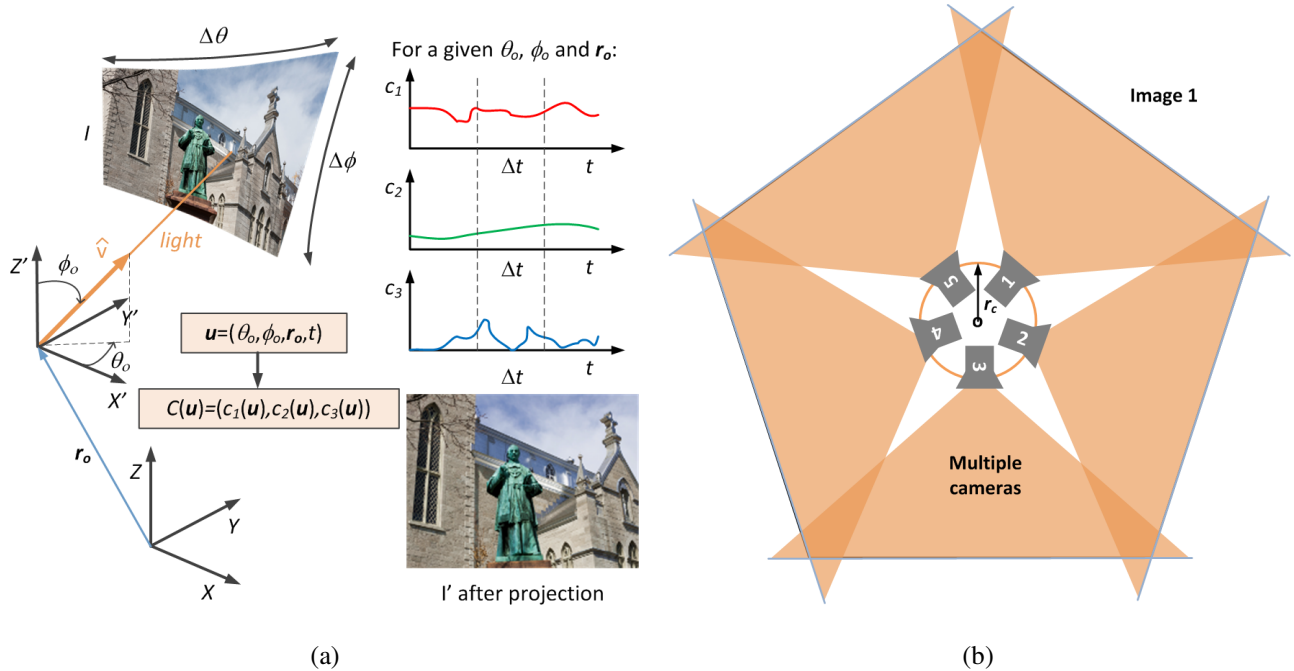


Figure 1. Sampling of the viewing function C to produce a monoscopic view of the scene: (a) concept of sampling C from the viewpoint \mathbf{r}_o , in the gazing direction (θ_o, ϕ_o) and for a field-of-view $(\Delta\theta, \Delta\phi)$, and (b) multi-camera configuration suitable to acquire the necessary information to approximate C .

The viewing function C represents the omnidirectional, but monoscopic, visual information to be acquired by a panoramic camera. The different panoramic cameras available today sample C to produce approximate panoramic renditions with different type of geometric distortions. For instance, a technique based on rotating a camera around its nodal point acquire partially overlapped images sequentially. This technique produces a single-viewpoint panorama that is a close approximation of sampling C . A suitable real-time approach may involve using multiple-sensors, which due to physical constraints, samples the visual scene from multiple viewpoints.

A multiple-camera configuration that exemplifies this limitation is shown in Fig. 1-(b) using five wide-angle sensors which projection centers are located at \mathbf{r}_c from the center O . One drawback of this method is the introduction of ghosting

due to mosaicking images acquired from different projection points. In this example, a radial distance $r_c \neq 0$ implies multiple projection points with non-zero parallax between them.

An omnistereoscopic version of the viewing function can be defined using two viewing functions with horizontal parallax b . Each of these viewpoints can be modeled by a viewing functions C_L and C_R , modelling the left and right binocular viewpoints, respectively. This zero parallax distance d is used to model the toe-in of the left and right sensor's optical axes, so they intersect at a distance d from \mathbf{r} , e.g., similarly to the vergence of both eyes towards a region of interest in the scene. For any gazing direction (θ, ϕ) and vergence distance d , measured from the reference point \mathbf{r} , the gazing directions for left (C_L) and right (C_R) viewing function are determined. The visual scene can be stereoscopically scanned by moving these binocular viewpoints around *circle of vision* of diameter b around the reference point \mathbf{r} . The circle of vision is defined on a plane parallel to floor level XY -plane in the world reference of coordinates XYZ . All these constraints help to model an omnistereoscopic viewing function as

$$C_S(\mathbf{u}_s, t) = (C_L(\mathbf{u}_l, t), C_R(\mathbf{u}_r, t))^T, \quad (2)$$

where t is the time variable, $\mathbf{u}_s = (\theta, \phi, \mathbf{r}, b, d)$ and the input parameters of the binocular viewing functions C_L and C_R , \mathbf{u}_l and \mathbf{u}_r can be calculated from \mathbf{u}_s using the constraints of the omnistereoscopic viewing model.

2.1 Approximating the omnistereoscopic viewing function

An omnistereoscopic camera, suitable to sample the scene from given observation point \mathbf{r} in real-time, should sample $C_S = \{C_L, C_R\}$ in any gazing direction (θ, ϕ) , given stereoscopic baseline b and a zero parallax distance d , and over a time lapse Δt . For instance, the visual sampling of the scene in the gazing direction (θ_0, ϕ_0) can be carried on by sampling a partial field-of-view $(\Delta\theta, \Delta\phi)$ FOV of the scene from two distinct viewpoints as modelled by C_S , integrating the acquired signal over Δt . The end result of the acquisition pipeline in this case will be a stereoscopic panorama $I_S = \{I_L, I_R\}$, where I_L and I_R are the left and right eye panoramic views, respectively. The acquisition concept based on the omnistereoscopic viewing model is illustrated in Fig. 2-(a).

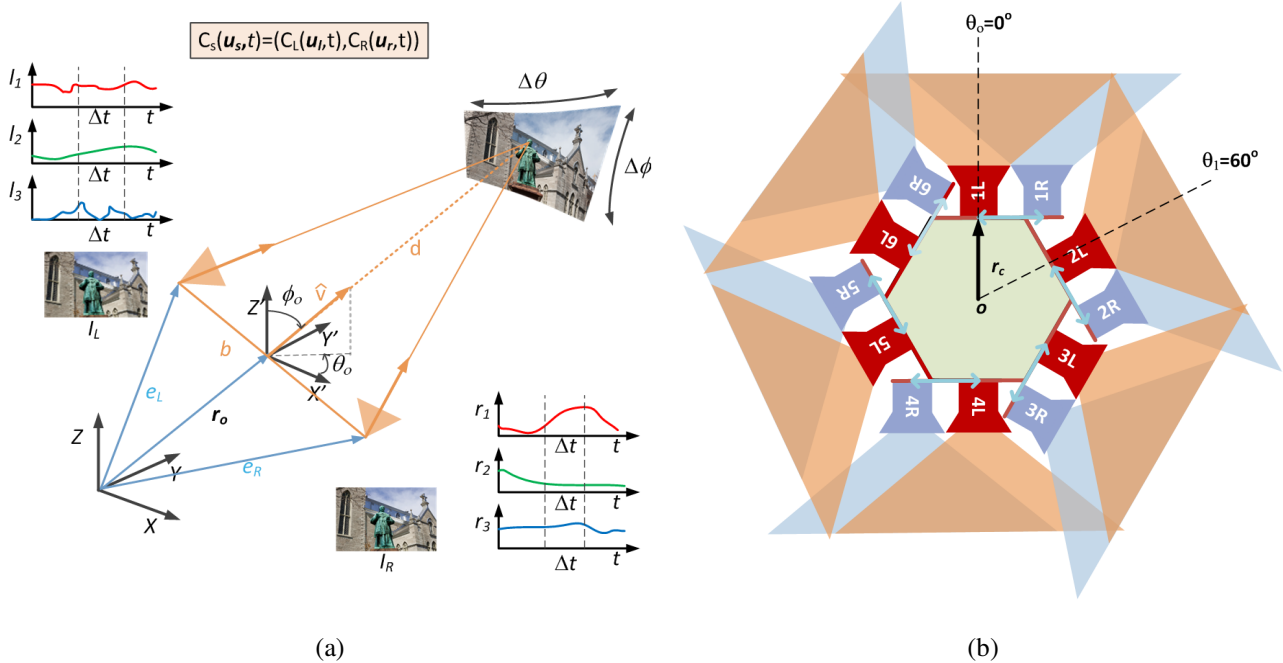


Figure 2. Acquisition of a stereoscopic (partial FOV) of the scene: (a) the binocular viewing functions C_L and C_R are sampled by a sensor device to provide a satisfactory binocular experience to the viewer, and (b) the multi-camera configuration proposed to stereoscopically sample the scene.

2.2 The proposed omnistereoscopic camera

In a static scene, a suitable omnistereoscopic acquisition strategy could be rotating a pair of cameras with horizontal parallax b around the middle point of between nodal points, taking partially overlapped images. This acquisition strategy corresponds to the acquisition model illustrated in Fig. 3-(a). This approach, which is a direct method to sample the value of C_S using a reduced number of stereoscopic samples, can be implemented with multiple cameras, enabling the omnistereoscopic sampling of C_S for dynamic scenarios. The design of such omnistereoscopic camera has to consider the number of cameras necessary given the azimuthal FOV of each camera lens. This consideration has to be taking into account to prevent mutual occlusion between cameras. Hence, narrow FOV lenses are needed, increasing the number of necessary cameras, e.g., some proposal suggested using several tents.¹³ The use of narrow FOV lenses with large baselines affects the minimum distance to objects in the scene for correct stereoscopic rendering.

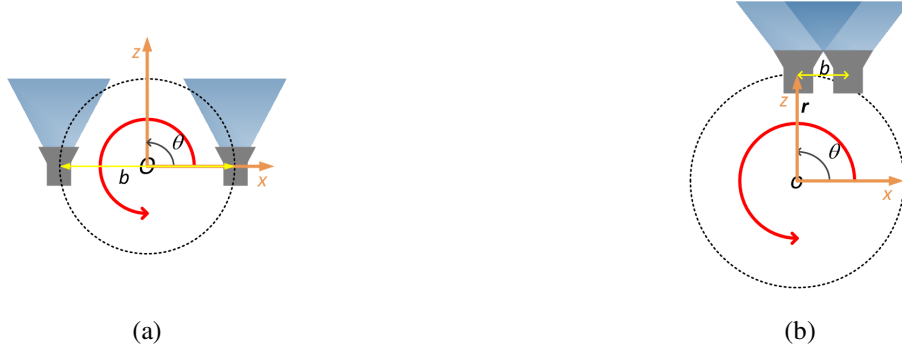


Figure 3. Stereoscopic viewing models used in the omnistereoscopic rendering: (a) an acquisition model for rotating a stereoscopic camera rig around the middle of its cameras' nodal points or some multi-camera arrangements, and (b) the acquisition model that fits the proposed multi-camera configuration.

In order to reduce the number of necessary cameras for a complete omnistereoscopic sampling, a radial configuration of multiple cameras is proposed. The proposed omnistereoscopic multi-camera is illustrated in Fig. 2-(b), which is based on the acquisition model depicted in Fig. 3-(b). This configuration enables using a reduced number (six) of wide-angle lenses to cover the whole scene. An advantage of this configuration is that, by using wide-angle lenses, it enables the stereoscopic capture of scene elements located closer to the camera than other multi-camera configurations.¹³ This camera is a modification of the configuration presented in Fig. 1-(b) to include a second horizontal parallax view in every direction. Unlike other multi-camera configurations based on the acquisition model depicted in Fig. 3-(a),¹⁵ the proposed camera facilitates the rendering a geometrically correct monoscopic panorama of the scene plus additional information to render omnistereoscopic views.

A monoscopic panorama of the scene with respect to the center of the arrangement O can be created by mosaicking the images acquired by the subset of cameras A . The radial configuration of the subset A makes easy the fast rendering of a polycentric panorama, which is a close approximation of a singular viewpoint panorama with respect to the geometric center of the arrangement. In the context of the omnistereoscopic viewing model, this is equivalent to sampling the viewing function C_L to estimate the left-eye viewpoint. The images captured by the subset of cameras B are used with the images acquired by the subset A to approximate the omnistereoscopic function C_S in any gazing direction (θ, ϕ) .

The proposed camera consists of six stereoscopic camera rigs, which simultaneously acquire six partially overlapped stereoscopic snapshots, by sampling the scene for $\theta_n = n \times 60^\circ$, for $n = \{0, \dots, 5\}$. Two set of images are acquired: one acquired by the subset of cameras $A = \{1L, 2L, 3L, 4L, 5L, 6L\}$, which exhibits a *star-like* configuration, and a second set of images acquired by the subset of cameras $B = \{1R, 2R, 3R, 4R, 5R, 6R\}$, which are the stereoscopic counterpart of the cameras in the subset A . To facilitate the analysis, the images acquired by each camera iL of the subset A are labeled I_{iL} , and the images acquired by each cameras iR of the subset B are labeled I_{iR} , in both cases, for $i = \{1, \dots, 6\}$. Using this notation, each stereoscopic image is labeled (I_{iL}, I_{iR}) , for $i = \{1, \dots, 6\}$, corresponding to a gazing direction θ_n , where $n = i - 1$.

A minimum azimuthal FOV $\Delta_\theta = 90^\circ$ is required for each identical camera lens. Under this constraint, the usable azimuthal FOV per image is $\Delta_\theta = 60^\circ$, measured from the center of each image. There are also two $\Delta_\theta = 15^\circ$ width

overlapping areas at least, each one measured from both image borders, towards each image’s center. These overlapped areas will be used in the aligning correction and blending of multiple images.

The radial distance r_c originates ghosting around boundaries between stitched images. This ghosting will be more noticeable for scene elements closer to the camera. However, this limitation can be minimized by choosing a short r_c , which will also affect the effective baseline b between stereoscopic cameras. Notice that in this camera configuration, the radial distance r_c is equivalent in length to the baseline b .

The zero parallax distance d introduced in the omnistereoscopic viewing model, is represented in this camera by the distance where each camera’s optical axis in the stereoscopic pair intersects. To avoid excessive parallax while mosaicking images, stereoscopic cameras with parallel axes are used. In other words, the zero parallax distance d is located at infinity.

There are two approaches for rendering using the information acquired by this multi-camera configuration. The first consist in using a short radial distance r_c , which will also determine a narrow stereoscopic baseline b . This configuration exploits the *microstereopsis* arising in narrow baselines. The second approach is based on a wider stereoscopic baseline to get better depth estimation. In this camera, a wider baseline leads to a larger r_c , which impacts the effective distance to the scene to avoid stitching artifacts. The second approach is based on using 2D-to-3D techniques to synthesize a horizontal parallax view based on the images captured by the subset A plus additional depth information. This approach helps to adjust the stereoscopic budget dynamically, besides enabling additional improvements in stereoscopic visual comfort. In the next section, both rendering approaches are discussed.

3. METHOD

The simplest approach to render omnistereoscopic views is using the images acquired by the subset of cameras A and B to create partially overlapped stereoscopic views in any gazing direction. One approach is mosaicking iL ($i = \{1, \dots, 6\}$) to create a left-eye panorama I_L , and doing the same with iR ($i = \{1, \dots, 6\}$) to render a right-eye panorama I_R . The geometry of the proposed camera configuration makes easier to render cylindrical projections of the scene, hence (I_L, I_R) are cylindrical panoramas. Alternative, partial stereoscopic views of the scene (I'_L, I'_R) in desired gazing directions (Fig. 2-(a)) can be rendered dynamically using this approach. Intermediate views between acquired images, e.g., for $\theta_n < \theta < \theta_{n+1}$ (for $n = \{0, \dots, 4\}$), can be extrapolated by mosaicking images corresponding to θ_n and θ_{n+1} for left and right-eye views. Although the intermediate stereoscopic views created with this method are not the same as those obtained by sampling C_S , the rendered stereoscopic images can provide a satisfactory binocular experience to the human viewer. The low complexity of this approach makes it suitable for real-time rendering of omnistereoscopic video. However, the mosaicking method introduces distortions by in combining images with different projective centers. To reduce distortion and to minimize the stitching problems, a short r_c is recommended. But, a small r_c implies a narrow stereoscopic baseline, since in this case $r_c \simeq b$. The depth perceived in these stereoscopic images is due to microstereopsis, which produces a gentle stereoscopic experience to the human viewer. This approach is explained in Section 3.2.

The second method is based on extracting depth information from each stereoscopic pair of images, (I_{iL}, I_{iR}) (for $i = \{1, \dots, 6\}$), to synthesize the right-eye view for any gazing direction. As in the previous method, the left-eye view is extracted from the monoscopic panorama I_L created with the images acquired by the subset of cameras A . This approach demands more computational processing to estimate the dense disparity-occlusion maps from each stereoscopic pair of images, combine this estimation in a panoramic depth-occlusion map, and finally warp the left-eye view according to the desired depth effect to create synthetic right-eye views. Despite the extra complexity, this rendering approach is flexible to adapt the depth budget in postprocessing, eliminating the need to reconfigure the multi-camera baseline for each scene. In addition, this approach facilitates the simulation of binocular visual mechanisms such as vergence and accommodation at rendering time. Offsetting these advantages, larger r_c leads to visible ghosting in I_L the mosaicked view and increased minimum distance to scene elements, at least in the left-eye panoramic rendering. To reduce ghosting, images can be warped based on minimizing the distance between matching features in the overlapping areas.^{10,16} This intentional image distortion has to be taken into account in the stereoscopic pair of images. This method is detailed in Section 3.3.

3.1 Rendering the left-eye view

Both rendering methods have one stage in common: the rendering of a monoscopic panoramic view, in this case sampling the C_L viewing function to render a *left-eye panorama* of the scene I_L , using the subset of images I_{iL} , for $i = \{1, \dots, 6\}$, acquired by the subset of cameras A . The rendering using partially overlapped snapshots of the scene obtained with multiple

cameras, or a rotating single camera, is well known.^{16,17} In this section, we propose a computationally efficient method to render views of intermediate gazing directions between captured views based on mosaicking the closest neighbour views. This method uses *microstereopsis* from narrow baselines b , due to short radial distance r_c . In the case of larger r_c , we propose corrective techniques based on warping images in order to reduce visible stitches while mosaicking. Notice that it is not necessary to render the whole panoramic view I_L at once since only partial FOV are required. However, for the sake of clarity, the rendering of a complete cylindrical panorama I_L is described. The rendering can be done off-line and the extraction of partial FOV images from it can be done in real-time according to the user's viewing direction.

The rendering process of the left-eye panorama I_L is illustrated in Fig. 4. First, the geometric distortions on each captures images I_{iL} need to be corrected generating a planar image I_{iL}^p . Then I_{iL}^p can be projected onto a cylinder, which originates I_{iL}^c . A fine aligning of each image with its immediate neighbours has to be considered to reduce stitching errors, which produces the aligned images I_{iL}^a . The next step is correcting the global color and luminance to improve the blending generating I_{iL}^{cc} . To reduce visible stitching, the optimal cut between neighbor images is calculated. Finally, the six color corrected images I_{iL}^{cc} corresponding to left-eye view are mosaicked using the calculated optimal cut to create a cylindrical panorama I_L . Details of the processing blocks are given in Sections 3.1.1-3.1.4.

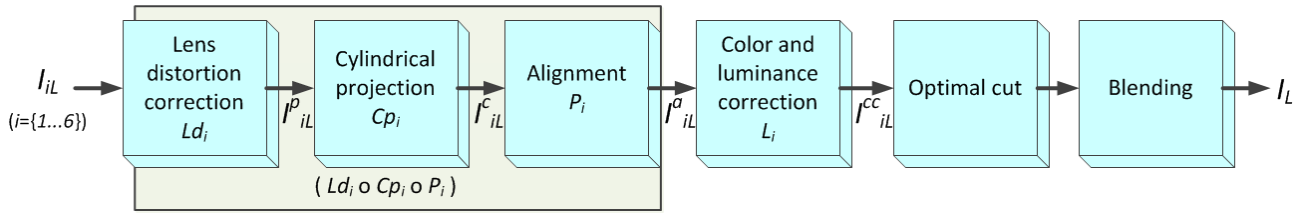


Figure 4. Rendering pipeline for the left-eye panorama I_L .

3.1.1 Lens distortion correction

The first step is to correct the lenses' distortion. Wide-angle lenses, i.e., fish-eye lenses, present spherical or aspherical distortion, therefore a resampling into planar images is necessary before mosaicking. The assumed distortion model will impact the correction results. Each of the twelve cameras involved need to be calibrated to estimate each image center and the parameters to correct the distortion. The calibration can be done using a checkerboard pattern from which a set of feature points can be extracted. An iterative method can be used to approximate the feature points to the desired undistorted set of features for a given model.¹⁸ A correction transformation Ld_i (for $i = \{1, \dots, 6\}$) is applied to each image I_{iL} to produce a planar image I_{iL}^p .

For the proposed multi-camera, the six pairs of images need to be projected first into a planar surface, then into a cylindrical surface, before proceeding to reconstruct the panoramic view of the scene. An example of the result of this processing is illustrated in Fig. 5.

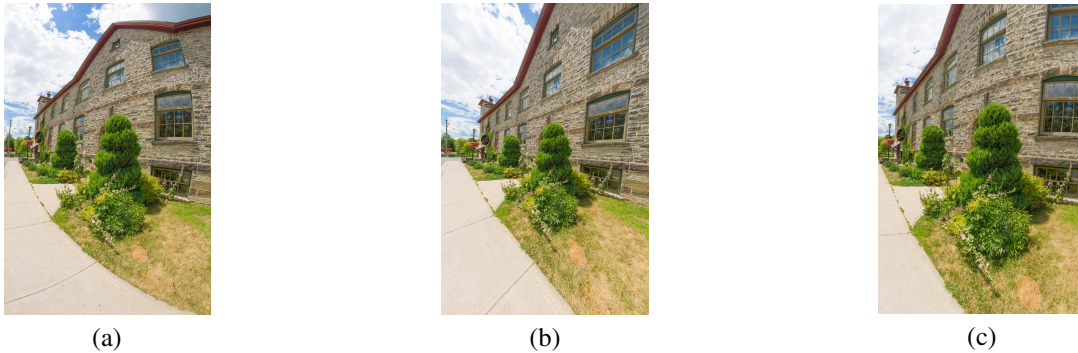


Figure 5. Aspherical lens distortion correction: (a) sampled image I_{1L} , (b) planar projection $I_{1L}^p = Ld_1(I_{1L})$, and (c) cylindrical projection $I_{1L}^c = Cp_1(I_{1L}^p)$.

3.1.2 Mosaic aligning

The next step is to correct the aligning between images. When the radial distance r_c is small (microstereopsis case), it is possible to use a simple translation transformation to align between neighboring images I_{iL} and $I_{(i+1)L}$.¹⁶ In most scenarios, using small r_c and a calibrated set of cameras, this step is not necessary or it can be avoided without penalty of ghosting. However, in case it is necessary given the proximity of scene elements to the camera or because a large r_c is used, a set of affine transformations P_i can be calculated to reduce the ghosting.

One possible method is extracting a set of feature points from the overlapped areas of neighbour images. A recursive method can be used to obtain the transformation P_i to be applied to images I_{iL}^c in order to minimize the mean-squared distance between feature points over the overlapped areas.¹⁶ An example of such an alignment procedure is shown in Fig. 6.

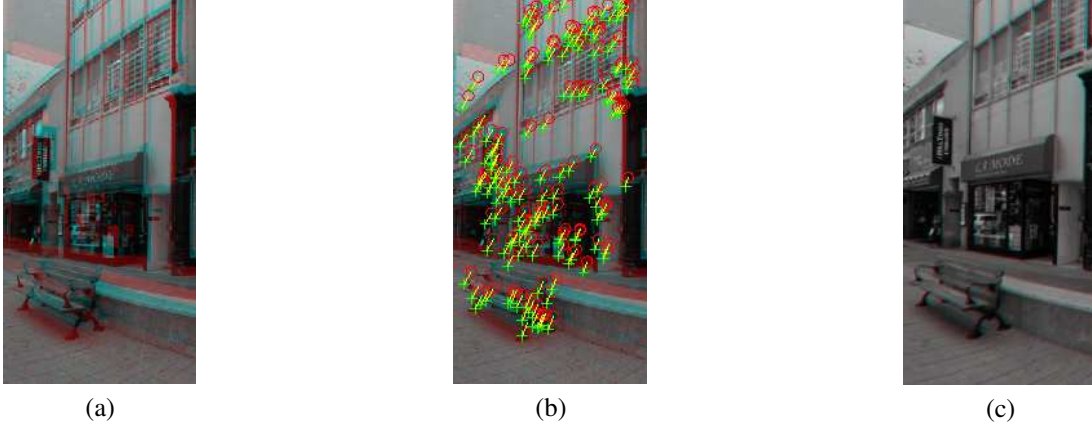


Figure 6. Mosaics alignment: (a) red-cyan anaglyph of the overlapping area between images I_{1L}^c and I_{2L}^c , (b) feature points calculated in the overlapping areas, and (c) red-cyan anaglyph of the overlapping area after alignment.

The set of optimally aligned images is $I_{iL}^a = P_i(I_{iL}^c)$, for $i = \{1, \dots, 6\}$. The price for reducing the stitching ghosting is to introduce geometric distortions. Each transformation P_i has to be applied to both images in the stereoscopic pair to prevent distorting the estimated disparity map or, in the case of mosaicking the right-eye view, to prevent introducing undesired vertical disparities in the rendered stereoscopic views. In order to prevent image degradation by blurring, a single combined transformation should be applied to the source images to account for the lens correction, planar projection and alignment, that is $I_{iL}^a = Wl_i(I_{iL})$, where $Wl_i = (Ld_i \circ Cp_i \circ P_i)$.

3.1.3 Color blending

The set of images acquired by the subsets of cameras A and B may have been originated with different photographic parameters. In a calibrated set of cameras, the white balance is set for all cameras before start the acquisition. However, each camera adjusts automatically the exposition time and aperture to optimize each shot, hence the final images' color and luminance need to be corrected before blending.

There have been different proposals for color blending, however, recent techniques proposed for low computational power, i.e, mobile phones cameras, can be very efficient in equalizing luminance and color between a reduced set of images. Xiong and Pulli¹⁹ proposed a simple algorithm for doing this based on calculating compensation coefficients over the overlapped areas between images plus a global compensation to reduce accumulative errors. This algorithm is easily parallelizable for real-time applications. It is usable in this case as long the color and luminance differences between adjacent images are not too large. An example of this color correction approach is shown in Fig. 7.

The global color-luminance correction is defined as a transformation L_i to be applied to each previously aligned image I_{iL}^a , such as a new set of color compensated images is defined: $I_{iL}^{cc} = L_i(I_{iL}^a)$.



Figure 7. Effect of the global color correction on the panorama I_L .

3.1.4 Mosaic stitching

Mosaicking to produce ghost-free images after alignment and color-luminance compensation can be done with different methods. The simplest approach is a linear blending where each color component is multiplied by a coefficient that weights its image contribution to the blend depending on the distance to the blending threshold. In other words, a blending of images I_1 and I_2 , each of them of width w , can be done using a function $a(x) = x/(w - 1)$ (for $x = \{0, \dots, w - 1\}$), which is then used to weight pixel values line wise: $I(x, y) = (1 - a(x)) \cdot I_1(x, y) + a(x) \cdot I_2(x)$. The linear blending is the fastest approach, but it may result in ghosting especially when parallax between cameras is important, e.g., too large r_c .

An improvement over the linear blending is using a multi-band decomposition. Typically, a Gaussian pyramidal decomposition is applied to the target areas to blend. The decomposition produces N increasingly blurred images, which are blended linearly. The reverse decomposition combines the new N blended images to produce the final blended image. This approach improves the blending adding more processing, but it does not improve the ghosting when r_c is too large.

The approach used to reduce the visible artifacts due to excessive parallax between neighbour cameras is based on obtaining the optimal cut over the region to blend. Linear blending is applied line by line based on the location of the optimal cut. The linear programming algorithm proposed by Ha et al.²⁰ is based on obtaining the locations of minimum intensity gradient between two images to blend. A scoring system enables to find the optimal cut line by line. This algorithm is attractive for its simplicity and can be easily parallelized for a real-time application. An example of the image blending is presented in Fig. 8. The example illustrates the effect of pyramidal blending and the optimal-cut method on two images obtained from the proposed camera using a large r_c .



Figure 8. Image blending for the proposed multi-camera ($r_c = 9$ cm) where the rectangular area illustrates the difference between using: (a) multi-band (pyramidal) blending and (b) linear blending based on the optimal-cut.

3.2 Method 1: rendering based on mosaicking stereoscopic snapshots

The multi-camera configuration for the omnistereoscopic mosaicking method has a narrow baseline b between neighbour cameras. This is done to produce *hypo-stereoscopic* rendition of the scene. The microstereopsis effect is sufficient to stimulate human binocular perceptual mechanisms of depth. In addition, a narrow r_c helps to reduce the likelihood of

cameras' parallax errors after mosaicking left and right views. An extra advantage is enabling the stereoscopic registration of scene elements located close to the camera. The stereoscopic baseline is typically chosen around the average interocular distance of the target viewer: between 5 cm and 6.5 cm. In the case of *hypo-stereoscopic* acquisition, b between 2 cm and 4 cm is adequate. Hence, the radial distance of the proposed camera should be restricted to: $2 \text{ cm} \leq r_c \leq 4 \text{ cm}$.

In the mosaicking approach, the right-eye panorama I_R is produced by mosaicking the images acquired by the subset of cameras B for a narrow baseline b . The left-eye view panorama I_L can be rendered with the procedure described in Section 3.1. The mosaicking procedure for the right-eye panorama I_R is similar to the method proposed to render the left-eye panorama, but it involves two extra steps. First, the registration of each stereoscopic pair of images is done over the planar images (I_{iL}^p, I_{iR}^p) after the lens distortion correction and before the cylindrical projection. The second extra consideration is the color and luminance equalization of each stereoscopic pair. This processing block can be inserted after the cylindrical projection block. The complete rendering pipeline for I_R is shown in Fig. 9 and the stereoscopic registration and histogram correction steps are explained in Section 3.2.1 and 3.2.2, respectively.

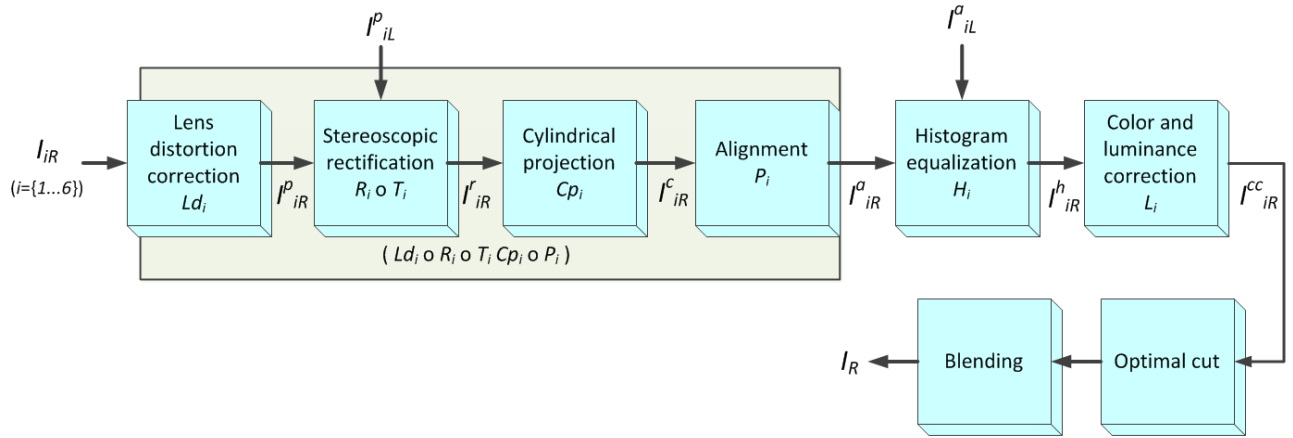


Figure 9. Right-eye panorama rendering pipeline.

Given the geometric disposition of the subset of cameras B , the artifacts in the final mosaicked panorama I_R will be more noticeable than in the case of I_L . However, since r_c is small, the stitching errors are minimal and affect only one of the stereoscopic views. The latter makes these errors more difficult to be perceived by the untrained observer when viewing a stereoscopic rendition of the scene.

3.2.1 Stereoscopic registration and depth consistency

Part of the calibration of each individual stereoscopic camera rig involves the stereoscopic registration. This can be done using a checkerboard calibration pattern located at a controlled distance from the camera pair. The calibration helps to find a projective transformation R_i that register I_{iR}^p on I_{iL}^p . This can be done using a similar procedure to the aligning method (Section 3.1.2), finding a controlled set of corresponding feature points and defining the projective transformation that minimize an error function, i.e., the mean distance between corresponding features. The second step of this calibration is to define a translation transformation T_i , which when applied to the registered I_{iR}^p will give the stereoscopic pair the desired horizontal disparity at the center of the image. This can be done accurately knowing the distance from the stereoscopic pair to the calibration pattern, which should be perpendicular to each camera optical axis. The result is a registered image $I_{iR}^r = (R_i \circ T_i)(I_{iR}^p)$.

Any transformation to apply on the set of images acquired by A will affect the stereoscopic registration. Hence, after applying the aligning transformation P_i to the image I_{iL}^c to produce the image I_{iL}^a , the same transformation should be applied to I_{iR}^r generating $I_{iR}^a = P_i(I_{iR}^r)$. As in the case of rendering I_L , to reduce blurring in the end result, a combined transformation (Fig. 9) should be applied to each right-eye image such as: $I_{iR}^a = Wr_i(I_{iR}^r)$, where $Wr_i = (Ld_i \circ R_i \circ T_i \circ Cp_i \circ P_i)$.

3.2.2 Color and illumination equalization

Another important step before mosaicking is to equalize the color and luminance of each stereoscopic image to prevent a flickering effect while watching the stereoscopic rendition of the scene. If the exposition and aperture of the cameras in each stereoscopic rig is not very different a simple histogram equalization H_i is sufficient. The equalization is applied after the right-eye image aligning: $I_{iR}^h = H_i(I_{iR}^a)$. Notice from Fig. 9 that after the color equalization another global color and luminance is applied (see Section 3.1.3). The latter is done to improve the blending of images acquired by the subset of cameras B and it does not affect the color equalization.

3.2.3 Zero parallax setting

Once the cylindrical stereoscopic panorama (I_L, I_R) has been rendered using the proposed mosaicking method, the zero parallax point (zero disparity location in the omnistereoscopic image) can be set by applying a pure translation to one of the cylindrical panoramas. This has the effect of toeing the virtual stereoscopic cameras as proposed in the omnistereoscopic viewing model. However, this method helps to simulate eyes vergence and caution must be taken. An excessive horizontal translation may originate areas in the scene with horizontal disparities beyond the threshold of stereoscopic visual comfort.

3.2.4 Visualization in the mosaicking scenario

The whole processing pipeline to mosaic an omnistereoscopic pair of images (I_L, I_R) in cylindrical format can be done off-line. If this is the case, a rectangular stereoscopic image, which approximates the sampling of C_s in the chosen gazing direction (θ, ϕ) can be rendered and streamed to the user on-demand. Alternatively, the whole omnistereoscopic view can be used in an immersive environment such as a CAVE, after retargeting the source panoramas to the display dimensions. A different alternative involves storing the twelve images and perform the rendering of partial views according to the user gazing direction. The latter partial rendering is done in real-time using the calibration transformation stored along with the captured images. Alternatively, the real-time option can be used to generate omnistereoscopic video, where the stereoscopic frames are dynamically generated using the twelve stereoscopic video sources. This can be done by mosaicking neighbour stereoscopic views frame by frame as a function of the virtual stereoscopic camera panning direction.

3.3 Method 2: rendering based on synthetic stereo

The proposed rendering method to synthesize omnistereoscopic imagery is based on estimating a dense horizontal disparity and occlusion maps using the information acquired by the proposed camera. The left-eye panoramic view is created by mosaicking images acquired by the subset of cameras A as explained in Section 3.1, while the right-eye views are rendered using a 2D-to-3D conversion. For instance, given a left-eye rectangular image, the right-eye image is rendered by warping the left-eye image according to the dense horizontal disparity map. The occluded areas after the pixel displacement can be restored using the captured occlusion information by texture inpainting or other hole-filling techniques.^{21,22}

Unlike the mosaicking method, which relies on microstereopsis arising from narrow baselines between cameras, this case requires a wide baseline to improve the resolution of the depth estimation. However, the proposed camera using a large baseline, i.e., $b > 9$ cm, will limit the minimum distance of objects to the camera for an artifact-free stitching. In other words, a larger b implies a larger r_c , which leads to more parallax between neighbour cameras and makes the rendering of I_L more difficult. However, there are methods to deal with the problems of excessive parallax if geometric distortion in the end-result is more acceptable than ghosting due to stitching (Section 3.1.2). In addition, any warping of the left-eye images has to be considering when mapping a left-eye pixel with its estimated horizontal disparity.

3.3.1 Optical flow

After the stereoscopic registration of each planar image (Section 3.2.1), the dense optical flow is estimated on each rectified image pair (I_{iL}^r, I_{iR}^r) . The estimation of the dense optical flow using stereoscopic images is an open area of research. In this paper, we used the first-order primal-dual algorithm for convex optimization proposed by Chambolle et al.²³ to estimate the dense optical flow. We propose for this application first to estimate a depth based segmentation of each image I_{iL}^r using the dense optical flow calculated over (I_{iL}^r, I_{iR}^r) , and then register each pixel on I_{iL}^r with its displacement magnitude to obtain a dense disparity map D_i for each stereoscopic image. The occlusion maps O_i are estimated by applying the estimated displacement map to I_{iL}^r and mapping, subtracting the novel image from I_{iR}^r and mapping the areas of high contrast as the occluded regions. Finally, panoramic versions of the horizontal disparity D_p and occlusion O_p can be created by fusing the data obtained from the partial estimations D_i and O_i .

3.3.2 Right-eye view synthesis

The produced disparity D_i and occlusion O_i maps for each image (I_{iL}^r, I_{iR}^r) can be used to map each pixel of the left-eye panorama I_L onto the corresponding disparity and occlusion values. The result of combining D_i and O_i (for $i = \{1, \dots, 6\}$) are panoramic disparity and occlusion maps D_p and O_p , respectively, that can be used along I_L to synthesize stereoscopic views in arbitrary gazing directions. For instance, a partial FOV view I'_L can be extracted from the left-eye panorama I_L for a gazing direction (θ, ϕ) , and then used along the horizontal disparity and occlusion maps corresponding to that gazing direction to synthesize the right-eye partial FOV I'_R . The rendering pipeline to synthesize the right-eye view is illustrated in Fig. 10.

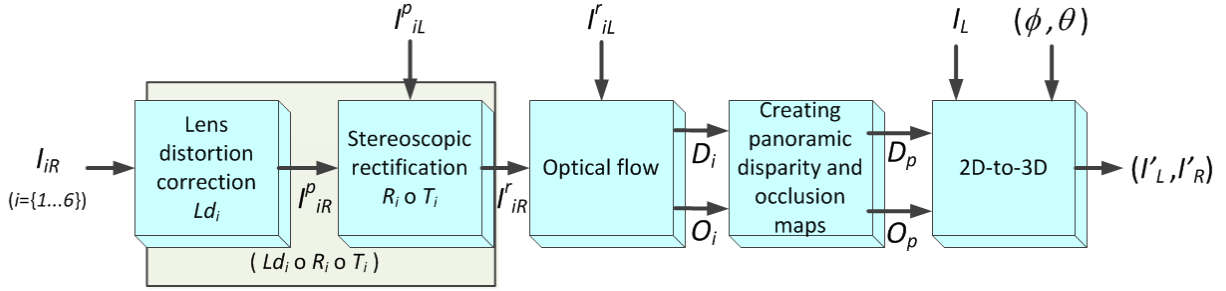


Figure 10. Synthetic omnistereoscopic rendering pipeline.

3.3.3 Visualization in the 2D-to-3D scenario

The left-eye mosaicked panorama I_L along the panoramic horizontal disparity D_p and occlusion maps O_p can be created off-line with the information acquired by the camera. In a case of a video, this can be done off-line frame by frame and stored in a server for future access.

One possible scenario is synthesizing right-eye views given a selected left-eye views extracted from I_L . In this case, the information in D_p can be adapted to create comfortable stereoscopic views of the scene considering the maximum and minimum parallax for a given display situation.²⁴ Another scenario is synthesizing the whole I_R off-line and stream the omnistereoscopic video or still to be projected in an immersive environment for a shared experience.

4. RESULTS

In order to prove the concept of the proposed multi-camera system, we built a test bench consisting of a single camera and a stereoscopic camera rig, both mounted on a panoramic head as shown in Fig. 11. The arrangement allows us to precisely position the camera in the location of each of the cameras in the subsets A and B shown in Fig. 2-(b). The experimental set-up helped us to emulate the proposed camera before actually building it by sequentially acquiring single or stereoscopic snapshots of the scene, in accordance with the spatial distribution of cameras the proposed multi-camera would have. Notice that the emulation respected the proposed camera configuration, which avoids occlusions between cameras, e.g., no camera in the array occludes the FOV of another camera.



Figure 11. Test set-up used to emulate the proposed camera: (a) sequential acquisition using a single camera, and (b) rotating stereoscopic rig.

The camera used is a DSLR Canon Rebel Xti 400D, which acquires 10.1 megapixel (3888×2592 pixels) resolution images. A set of identical wide-angle (fish-eye) lenses Bower (a.k.a Samyang) SLY-358C with a focal length of 8 mm were used. An aperture of $f11$ was used to guarantee a large depth-of-field. Using this aperture in this camera, exposition times in the range of 50 ms to 100 ms were enough to get a clear picture avoiding motion blur in most of the chosen scenes. This lens enabled an effective 90° FOV in azimuth and 150° FOV in elevation after lens correction, mounting the camera in portrait mode.

The camera was mounted on a panoramic head Manfrotto 303SPH, which was installed on a solid tripod. The arrangement was rotated 360° in azimuth on a horizontal plane, parallel to the ground plane. The rotating plane was leveled before starting of each capture. An example of the single camera set-up is illustrated in Fig. 11-(a). This set-up was used to emulate omnistereoscopic acquisition in static environments. In order to capture dynamic scenes, the single camera was replaced by a stereoscopic camera to emulate the simultaneous capture of left and right eye images as shown in Fig. 11-(b). The stereoscopic camera rig was built using two identical cameras and lenses.

4.1 Method 1

Once the camera-lens nodal point was calibrated, the camera was moved off-center $r_c = 3$ cm. In the case of a single camera, a sequence of six snapshots were taken emulating the acquisition with the subset of cameras A . This was done by rotating the panoramic head $\Delta_\phi = 60^\circ$ clockwise after each snapshot. Once a complete sequence was completed, the camera was laterally displaced b cm to the right of its original position to emulate the subset of cameras B . Again, the panoramic head was rotated Δ_ϕ after each capture. A total of twelve pictures, (I_{iL}, I_{iR}) (for $i = \{1, \dots, 6\}$), per viewing location were acquired.

To emulate the acquisition of dynamic scenes, the stereoscopic camera rig was rotated $\Delta_\phi = 60^\circ$ after each stereoscopic snapshot. As a result, six partially overlapped stereoscopic images were collected per location. Unlike the single camera set-up, this test set-up was able to capture dynamic scenes, at least on each image. Problems occurred in fast changing scenes where the scene elements appeared registered in more than one image or in between images. Beside this sequential limitation, this set-up proves the concept of stereoscopic acquisition of dynamic environments.

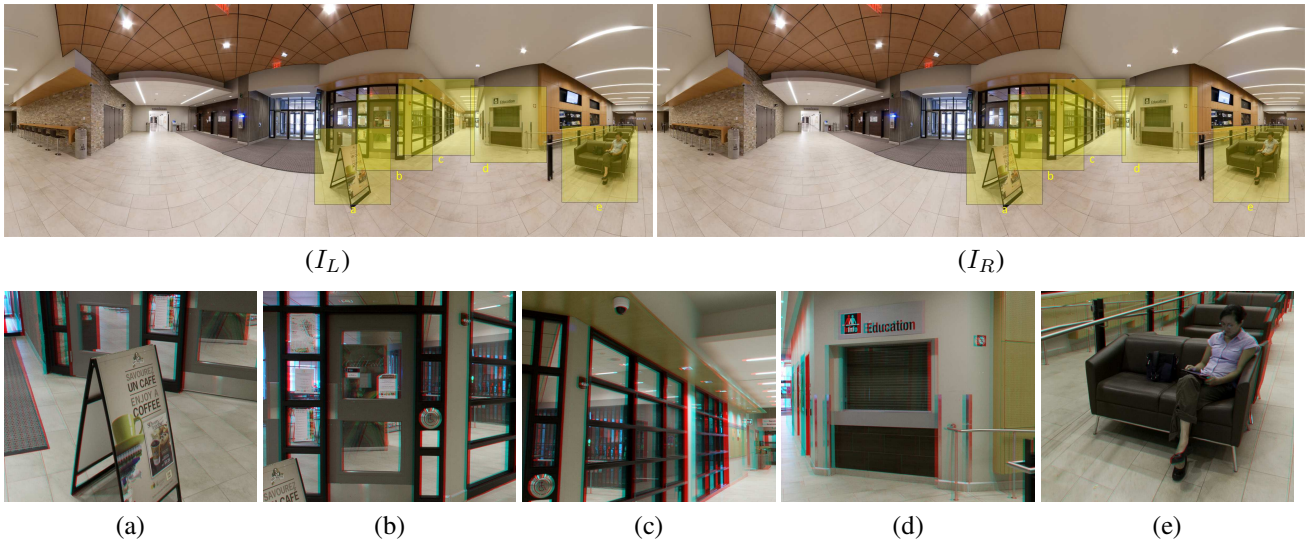


Figure 12. Examples of stereoscopic views generated using the micro-stereopsis method: (top) left and right cylindrical panoramas, (below) red-cyan anaglyphs of the stereoscopic views indicated on the top omnistereoscopic pair.

The acquired set of images were processed separately dividing them in a subset corresponding to subsets A and B . The lens correction was done using the lens profile provided by the manufacturer. The images from the subset A were used to create a cylindrical panorama I_L as described in Section 3.1, while the images corresponding to the subset B were processed to generate the right-eye panorama I_R using the procedure described in Section 3.2. An example of these omnistereoscopic cylindrical pair is shown in Fig. 12.

Whenever a single camera was used, the calibration was limited to obtain the intrinsic camera parameters. The aligning transformation was estimated one time using features extracted from the overlapped areas of two randomly selected images and then applied to the whole set of images, e.g., $P_i = P$ for all i . The stereoscopic registration was done using features extracted from each image pair since no global stereoscopic calibration was possible in this case.

The left eye-panorama I_L , which is based on mosaicking the images acquired by the subset of images A , exhibits less parallax errors in comparison with the right-eye panorama I_R based on mosaicking images acquired by the horizontal displaced subset of cameras B . This is expected given the radial distribution and relative skew of the subset of cameras A favours the approximation of a singular viewpoint panorama with respect to the center of the arrangement O . This can be seen in Fig. 13-(a) and (b).

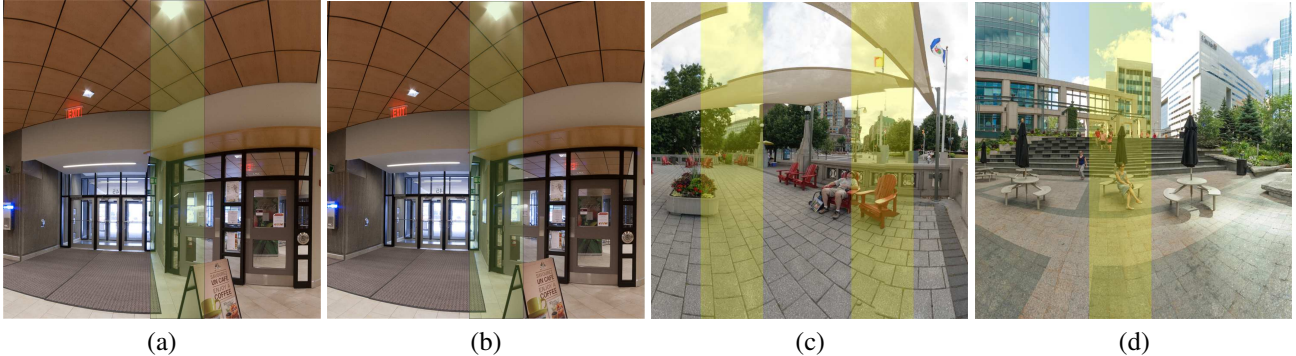


Figure 13. Mosaicking errors: (a) and (b) are the left and right views on a stitching area for the microstereopsis case ($r_c = 3$ cm), and (c)-(d) two examples of stitching error due to a larger radial distance ($r_c = 9$ cm).

4.2 Method 2

The stereoscopic camera rig was positioned in the panoramic head to respect the configuration of the subsets A (left camera) and B (right camera) presented in Fig. 2-(b). In this case, the position of the nodal point of left camera in the stereoscopic rig was first obtained. Then the stereoscopic rig was displaced off-center $r_c = 9$ cm. In order to emulate the omnistereoscopic camera proposed, we used this large radial displacement. The radial displacement is not arbitrary: it was determined by the cameras' physical dimensions, which constrains the minimum baseline b . Using these off-the-shelf cameras, the baseline was $b = 9$ cm. This wide-baseline was sufficient to emulate the *hyper-stereoscopic* effect we were looking for, but it constrains the test to outdoor scenes, which satisfies the desired distance between the scene and the camera.

In order to reduce the noticeable parallax for the left-eye panorama I_L , each image in the image set A was aligned defining a transformation P_i as described in Section 3.1.2. In this case, it was not necessary to apply the alignment transformation to the images in the subset B , it was enough to know the stereoscopic rectification transformation ($R_i \circ T_i$) and P_i to define a correspondence between the pixels in I_L and the disparity and occlusion information. Examples of I_L created for different outdoor locations are presented in Fig. 14.

The stereoscopic registration was done one time since the same stereoscopic rig was rotated to acquire six pair of images. The transformation obtained from this stereoscopic registration, ($R \circ T$), was applied to the whole set of images B . The procedure described in Section 3.3 was applied to obtain the horizontal disparity and occlusion maps. A 2D-to-3D transformation was applied to selected views of the scene to render synthetic right-eyes views from the rendered left-eye views. An example of this is illustrated in Fig. 15.

A companion website for this stereoscopic camera project with more navigable stereoscopic views and examples is available at: <http://luisgurrieri.net/publications/spcamera/>

5. CONCLUSIONS

We have proposed a multi-camera configuration that is suitable to acquire omnistereoscopic images and videos of dynamic scenes. The camera concept is based on acquiring information to render a geometrically correct monoscopic panorama and



Figure 14. Examples of left-eye cylindrical panoramas created with the proposed camera for the rendering method 2 ($r_c = b = 9$ cm).

additional information to render stereoscopic views in any gazing directions. We presented two omnistereoscopic rendering strategies for the proposed camera. The rendering method 1, based on mosaicking narrow baseline stereoscopic images, is attractive for its simplicity. The results of using microstereopsis are effective in creating binocular renditions of a scene that are credible to a human viewer. The inexpensive computational complexity of method 1 makes it attractive for real-time rendering of omnistereoscopic videos and imagery. We also presented a rendering method 2, which is based on synthesizing binocular views in arbitrary gazing directions. This alternative requires a wider baseline to be able to estimate disparity and occlusion maps. The parallax between neighbour cameras increases in method 2 as well as the stitching problems. Despite its disadvantages, the method 2 is suitable for creating appealing stereoscopic views, adapting the acquired material for different displaying scenarios and depth budgets. In addition, vergence and accommodation mechanisms can be simulated using the second method. A general strategy for the acquisition of omnistereoscopic images and videos can be conceived using the proposed camera and reducing the radial distance to an intermediate value ($r_c = 6$ cm). This will enable to acquire moderately wide-baseline omnistereoscopic images without incurring in excessive parallax. This camera will enable creating a left-eye panoramic view I_L by mosaicking and a right-eye panorama I_R by any of the rendering methods described in this paper. For instance, method 1 can be preferred in real-time applications such as omnistereoscopic video, while method 2 can be used for off-line rendering.

ACKNOWLEDGMENTS

The authors would like to thank Quyen Sy for their invaluable assistance in the acquisition of panoramic samples. This work was supported by the Ontario Graduate Scholarship (OGS) fund and by the Natural Sciences and Engineering Research Council of Canada (NSERC).

REFERENCES

1. Uyttendaele, M., Criminisi, A., Kang, S. B., Winder, S., Szeliski, R., and Hartley, R., “Image-based interactive exploration of real-world environments,” *IEEE Computer Graphics and Applications*, 52–63 (2004). 2
2. Tanimoto, M., “FTV (free viewpoint television) creating ray-based image engineering,” *IEEE International Conference on Image Processing 2*, 25–32 (2005). 2
3. Southwell, D., Basu, A., Fiala, M., and Reyda, J., “Panoramic stereo,” *Proc. IEEE Int. Conf. Pattern Recognition*, 378–382 (1996). 2
4. Gluckman, J., Nayar, S., and Thoresz, K., “Real-time omnidirectional and panoramic stereo,” *Proc. of DARPA Image Understanding Workshop 1*, 299–303 (1998). 2
5. Kawanishi, T., Yamazawa, K., Iwasa, H., Takemura, H., and Yokoya, N., “Generation of high-resolution stereo panoramic images by omnidirectional imaging sensor using hexagonal pyramidal mirrors,” *Proc. of the 14th International Conference on Pattern Recognition*, 485–489 (1998). 2

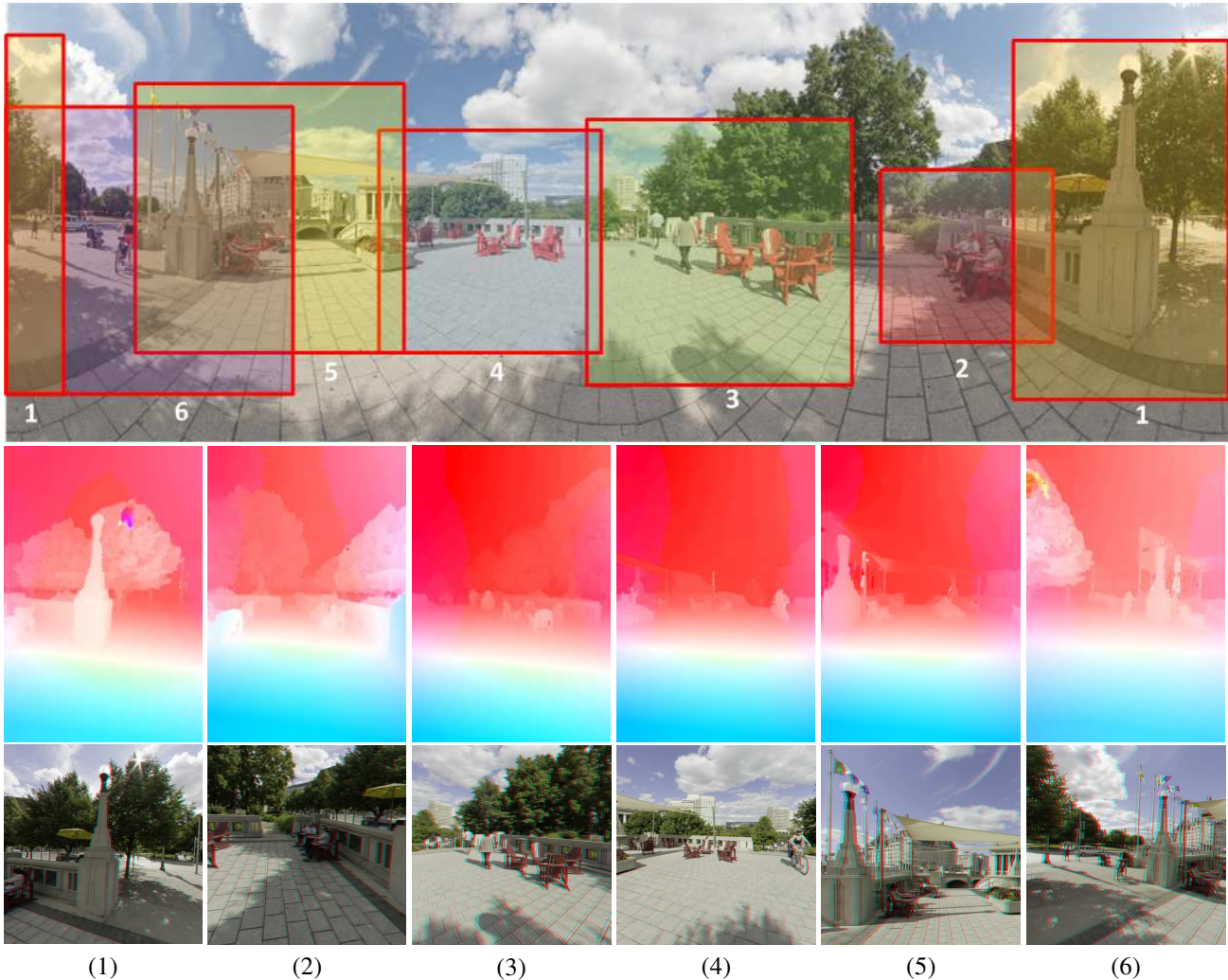


Figure 15. Examples of stereoscopic views generated using the method 2: (top) left-eye cylindrical panorama (I_L) indicating the target regions to render, (center) color coded optical flow maps, and (bottom) the synthetic stereoscopic views (red-cyan anaglyphs) corresponding to the gazing directions indicated in I_L .

6. Spacek, L., “Coaxial omnidirectional stereopsis,” in [Computer Vision - ECCV 2004], Lecture Notes in Computer Science 3024, 354–365 (2004). [2](#)
7. Weissig, C., Scherr, O., Eisert, P., and Kauff, P., “The Ultimate Immersive Experience: Panoramic 3D Video Acquisition,” in [Advances in Multimedia Modeling] 7131, 671–681 (2012). [2](#)
8. Huang, F., Klette, R., and Scheibe, K., [Panoramic Imaging: Sensor-Line Cameras and Laser Range-Finders], Wiley (2008). [2](#)
9. Peleg, S. and Ben-Ezra, M., “Stereo panorama with a single camera,” Proc. IEEE Conf. Computer Vision Pattern Recognition, 395–401 (1999). [2](#)
10. Gurrieri, L. E. and Dubois, E., “Efficient panoramic sampling of real-world environments for image-based stereoscopic telepresence,” Proc. SPIE Stereoscopic Displays and Applications XXIII 8288, 1–14 (2012). [2](#), [6](#)
11. Vanijja, V. and Horiguchi, S., “A stereoscopic image-based approach to virtual environment navigation,” The Computer the Internet and Management 14, 68–81 (2006). [2](#)
12. Tzavidas, S. and Katsaggelos, A., “Multicamera setup for generating stereo panoramic video,” Proc. SPIE Three-Dimensional Image Capture and Applications V 4661, 47–58 (2002). [2](#)
13. Baker, H. H. and Constantin, P., “Panoramic stereoscopic camera,” US Patent (0105574), (2012). [2](#), [5](#)

14. Grover, T., "Multi-dimensional imaging," US patent (7796152B2), (2008). [2](#)
15. Baker, R. G., Baker, F. A., and Conellan, J. A., "Panoramic stereoscopic camera," US patent (2008/0298674 A1), (2008). [5](#)
16. Szeliski, R., "Image alignment and stitching: A tutorial," *Foundations and Trends in Computer Graphics and Vision* 2(1), 1–10 (2006). [6](#), [7](#), [8](#)
17. Szeliski, R., "Video mosaics for virtual environments," *IEEE Computer Graphics and Applications* 16(2), 22–30 (1996). [7](#)
18. Hughes, C., Glavin, M., Jones, E., and Denny, P., "Review of geometric distortion compensation in fish-eye cameras," *IET Irish Signals and Systems Conference*, 162–167 (2008). [7](#)
19. Xiong, Y. and Pulli, K., "Color correction for mobile panorama imaging," *Proc. of the 1st Int. Conf. on Internet Multimedia Computing and Service*, 219–226 (2009). [8](#)
20. Ha, S. J., Koo, H., Lee, S. H., Cho, N. I., and Kim, S. K., "Panorama mosaic optimization for mobile camera systems," *IEEE Transactions on Consumer Electronics* 53, 1217-1225 (2007). [9](#)
21. Zhang, L. and Tam, W., "Stereoscopic image generation based on depth images for 3D TV," *IEEE Transactions on Broadcasting* 51, 191-199 (2005). [11](#)
22. Zhang, L., Vazquez, C., and Knorr, S., "3D-TV content creation: Automatic 2D-to-3D video conversion," *IEEE Transactions on Broadcasting* 57, 372-383 (2011). [11](#)
23. Chambolle, A. and Pock, T., "A first-order primal-dual algorithm for convex problems with applications to imaging," *Journal of Mathematical Imaging and Vision* 40(1), 120-145 (2011). [11](#)
24. Lang, M., Hornung, A., Wang, O., Poulakos, A., Smolic, A., and Gross, M., "Nonlinear disparity mapping for stereoscopic 3D," *ACM Transactions on Graphics (TOG)* 29(4), 75-85 (2010). [12](#)