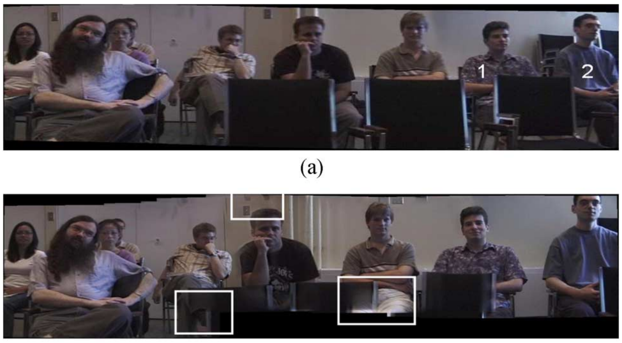
Toward Dynamic Image Mosaic Generation With Robustness to Parallax

Qi Zhi and Jeremy R. Cooperstock

***Abstract*—Mosaicing is largely dependent on the quality of registration among the constituent input images. Parallax and object motion present challenges to image registration, leading to artifacts in the result. To reduce the impact of these artifacts, traditional image mosaicing approaches often impose planar scene constraints or rely on purely rotational camera motion or dense sampling. However, these requirements are often impractical or fail to address the needs of all applications. Instead, taking advantage of depth cues and a smooth transition criterion, we achieve significantly improved mosaicing results for static scenes, coping effectively with nontrivial parallax in the input. We extend this approach to the synthesis of dynamic video mosaics, incorporating foreground/background segmentation and a consistent motion perception criterion. Although further additions are required to cope with unconstrained object motion, our algorithm can synthesize a perceptually convincing output, conveying the same appearance of motion as seen in the input sequences.**

***Index Terms*—Depth-based image mosaicing (DBM), dynamic mosaic, object motion, parallax.**



(a)

(b)

Fig. 1. Comparison of mosaic results from inputs with and without parallax effects. The results of (b) exhibit artifacts within the highlighted bounding boxes. (a) Panorama using parallax-free inputs: an extreme wide-angle effect is seen in the views of human subjects 1 and 2. (b) Panorama using nonparallax-free inputs from translationally offset cameras.

I. Introduction

IMAGE mosaicing refers to a process of combining multiple images with overlapping field of view (FOV) to produce a panorama. In theory, it compensates for the limited FOV of a single camera without sacrificing resolution or introducing undesirable distortions, as does the use of a wide angle or omnidirectional lens, to achieve a comparable panoramic view.

To ensure perceptually correct results, traditional systems typically impose constraints of planar scenery, a parallax-free camera configuration [10], [13], [45], or a dense sampling of the scene[1] [35], [60]. However, these requirements may be impractical or prohibitively expensive to satisfy. In particular, indoor environments typically exhibit nonplanar contents with complex depth distributions. Although a parallax-free configuration can be used to construct a valid mosaic, this may produce perceptually unacceptable results. An example is the extreme wide-angle effect evident in the views of the people labeled 1 and 2, positioned toward the right edge of the output panorama in Fig. 1(a). The appearance of those individuals would be ineffective in a group videoconferencing scenario. In such cases, a configuration with the cameras directly facing the objects of interest is sometimes preferable.

If the above constraints are violated, parallax effects between inputs make it difficult to achieve an accurate *image registration* that is applicable for the entire scene. Resulting misalignments may cause local intensity or structural inconsistencies, which induce visual artifacts in output mosaics, as illustrated in the highlighted bounding boxes in Fig. 1(b). In this example mosaic, generated by Autostitch with nonparallax-free inputs, the multiple instances of the light switch on the background wall are considered *duplication errors*, whereas the gradual blending of human legs with the chairs constitute *ghost artifacts*.

The presence of moving objects in the scene is another issue that frustrates traditional image mosaicing. The dynamic content results in changes between successive frames, either of texture or features used for image alignment. Moreover, in a sequence of mosaic frames, such object motion can induce interframe *jitter* in the appearance of the background and discontinuities in the movement of these objects.

Despite the difficulty of establishing universally accepted image quality metrics, we suggest the following three

Manuscript received November 12, 2010; revised April 30, 2011; accepted July 05, 2011. Date of publication July 22, 2011; date of current version December 16, 2011. The work was supported in part by the Advanced Network Applications, Services and Technologies Program of Canarie, Inc., and in part by the Natural Sciences and Engineering Research Council of Canada. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Kiyoharu Aizawa.

Q. Zhi is with the National Engineering Research Centre for ASIC, Southeast University, Nanjing 210096, China (e-mail: qizhi@cim.mcgill.ca).

J. R. Cooperstock is with the Centre for Intelligent Machines, McGill University, Montreal, QC H3A 2A7, Canada (e-mail: jer@cim.mcgill.ca).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2011.2162743

[1] A dense sampling is obtained when the set of input images exhibit significant overlap with each other.

properties that should be respected by an image mosaicing algorithm.

- **Feature** or **structure preservation**. The result should not violate existing features or create new ones[2] in the scene; otherwise, artifacts may result, as illustrated in Fig. 1(b).
- **Shape preservation**. Objects should appear free of distortion, i.e., maintain a shape consistent with the inputs.
- **Motion preservation**. The motion of dynamic objects in the mosaic video should be both spatially and temporally consistent with the one that occurs in the input.

To address the above issues, we introduce a novel image mosaicing algorithm formulated as a view synthesis problem. This renders the panorama as a single frame using depth estimates from the viewpoint of a virtual mosaicing camera. It notably includes the contents in nonoverlapping regions between sources by using a depth propagation procedure. Provided that the depth values are sufficiently accurate, the output mosaic result is perceptually satisfactory and free of parallax-related artifacts.

If combined with the technique of foreground–background segmentation and a consistent motion perception criterion, we can extend this paper to the synthesis of dynamic video mosaics, i.e., faithfully representing dynamic events in the environment. To cope with the added challenge of moving objects, our novel depth-based image mosaicing (DBM) technique divides the task into two steps, i.e., first, projecting the segmented background onto the image plane and, second, constructing the added foreground layers on a frame-by-frame basis. The result preserves the correct display of static scene content, as well as the spatiotemporal coherence of object motion. We are unaware of any comparable mosaicing technique that has proven capable of addressing nontrivial parallax effects from sparse sampling in either static or dynamic scenes.

The remainder of this paper is organized as follows: Previous work in image mosaicing is summarized in Section II. An overview of our approach is then presented in Section III, before delving into the details of each step of the algorithm. Section IV describes our DBM approach, on which the dynamic mosaic algorithm, described in Section V, relies. Section VI provides a comparison of experimental results with those of Autostitch, and Section VII concludes with a discussion of desirable improvements.

II. Literature Review

Considerable effort has been invested to increase the robustness of mosaicing to illumination variation [53], [56], exposure differences [16], lens distortion [25], [26], improvement of the computational efficiency [14], [31], [37], and other such challenges. However, for the purpose of this paper, we concentrate on image mosaicing techniques that address the issues of parallax and object motion in the scene.

Image mosaicing algorithms traditionally follow a structural alignment approach, involving warping and stitching. These steps can be complicated by the introduction of parallax, which degrades the quality of image alignment. To avoid such complications, some algorithms impose constraints of planar scenes or parallax-free camera configurations. For example, *QuickTime VR* [13] generated a panoramic view of the environment based on images from a rotating camera. Shum and Szeliski [45] introduced global and local image alignments to reduce accumulated image registration errors when given inputs of approximately planar scenes.[3] Further progress in dealing with the challenges of artifacts due to object motion and exposure differences was achieved [51].

Recently, through the improvements of distinctive feature detection and matching [5], [20], [32], more robust structural alignment mosaicing algorithms have been proposed. For example, Brown and Lowe described *Autostitch* [11], i.e., a fully automatic method for constructing panoramas, which was insensitive to the ordering, the orientation, the scale, and the illumination of input images. Similarly, for the construction of high-quality X-ray panoramic images, Wang *et al.* [52] strictly control the motion of both the camera and the surgery table. The system compensates for slight camera translations to ensure that the inputs remain free of parallax effects.

Another group of structural alignment mosaicing algorithms, i.e., *manifold mosaicing*, makes use of dense sampling to compensate for depth variance in the environment and thus offers greater robustness to input camera motion. Following the definition in [35] and [36], a manifold mosaic, i.e., a multiperspective image [39], was built by projecting thin warped strips from input images onto the mosaicing surface. Zomet *et al.* [60] introduced a new manifold mosaicing algorithm based on *crossed-slits projection*, whose results are closer to perspective images than those of traditional pushbroom mosaics [19].

Agarwala *et al.* [1] reduced the artifacts of manifold mosaicing by minimizing error functions based on the criteria of smooth and continuous strip connections. Their approach is representative of an alternative group of mosaicing algorithms, based on image intensity alignment, inspired by texture composition methods [17], [28]. In general, these algorithms first select a seam in the overlapping regions between inputs and then smoothly blend the stitched images. The choice of the seam is controlled by the optimization of an energy function, which prefers junction curves minimizing intensity differences across the seam. Levin *et al.* [30] proposed Gradient-domain Image STitching (GIST) to overcome the problem of intensity disagreement between input images. However, this cannot cope with large misalignment of the structure. More recently, Jia and Tang [24] simultaneously achieved the alignment of image structure and intensity by applying a method based on structure deformation and propagation. Although these methods deal with local or global intensity difference between inputs, they cannot handle the obvious structural misalignment that arises from parallax limited overlap between inputs or dynamic contents in the scene, which limit the number of reliable corresponding features.

The aforementioned algorithms are intended for constructing panoramas of mainly static scenes. Even when combined with

[2] Feature, i.e., structure, is a general concept in computer vision. This can be a group of points, edges, or complex structures representing objects. The choice of features in a particular computer-vision system is highly problem dependent.

[3] The scene contents were assumed to be far from the camera center, thus approximating the planar condition.

the optimization of energy functions as implemented by the intensity alignment mosaicing algorithms, they remain limited in their ability to compensate for motion-related artifacts.

In contrast, other algorithms apply robust image alignment procedures to minimize the impact of dynamic objects. These allow for the accurate estimation of camera motion and, hence, improved image registration, even when the scene contains time-varying content. Irani *et al.* [22] propose the distinction between *static* and *dynamic* mosaics. The former provides a single view of the full scene over the entire duration of the input video, whereas the latter results in a sequence of images, each of which updated according to the most recent frame in the input.

In the category of dynamic mosaics, Sawhney and Ayer [41] introduced a motion-separated layered representation of the input video. Based on dominant motion separation, the input sequence was divided into a layer of fixed background and multiple layers of moving objects. Bartoli *et al.* [4] instead applied a combined feature-based and direct image registration procedure to address the object motion in inputs. Although this resulted in three useful components, namely, a background panorama, a registered input sequence, and dynamic layers containing moving objects, the authors did not describe how such components could be combined to produce a true dynamic mosaic. Taking advantage of dense sampling inputs and the manifold mosaicing synthesis technique, Alex *et al.* [3] achieved the smooth appearance of the object motion in the final dynamic mosaic video at the expense of a faithful chronological ordering of the object motion. In addition, all of these algorithms relied on parallax-free (rotational) camera configurations or a dense sampling of the scene. Mills and Dudek [33] extended previous work in intensity alignment mosaicing to overcome the presence of moving objects in the scene by applying a heuristic seam selection procedure. However, their approach was not intended to deal consistently with dynamic contents in the temporal domain and is thus unsuitable for dynamic video mosaicing.

If depth estimates are available for both static and dynamic contents, the dynamic mosaic can be easily constructed. However, obtaining such depth estimates for the dynamic content when the video sources exhibit only partial overlap remains a difficult problem. In such a scenario, the structure from motion can be used to reconstruct the shape and the position of moving objects from a monocular video sequence. The method is based on the factorization algorithm in [49], which is only applicable to rigid objects. A newer technique, i.e., the nonrigid structure from motion [9], [50], can estimate time-varying 3-D shapes from 2-D point tracks in the monocular video input but is still limited to objects with comparatively low degrees of freedom of movement. However, a full-body human subject may generate significantly more complex motion patterns and often impose the added challenge of self-occlusions during movement. In this situation, the 3-D reconstruction of a scene that includes moving people is thus difficult at best and, more often, an ill-posed problem.

III. Summary of our Approach

These limitations, coupled with the perceptually unacceptable results presented in Section I, motivated our research to develop new techniques that overcome the traditional constraints of mosaicing algorithms, particularly for the construction of the true panoramic video of moving objects. Our efforts contribute to high-quality mosaics when given a sparse sampling of either a static or dynamic environment, with potentially significant depth variance, and in situations where the camera configuration may introduce parallax.

Our mosaic result is first divided into an overlapping region R_o, which is observed by multiple input cameras, and the nonoverlapping regions R_{non}, for which no stereo information is available. The view synthesis algorithm plane sweep is used to separate these two regions for the static background scenes based on whether the matching score, which implies the reliability of the correspondences among sources, is below a certain threshold. However, for the dynamic scene content, we simply consider the moving object (or person) to be in R_o when its silhouette is visible to both cameras. This can be performed through the analysis of the foreground silhouettes between overlapping cameras. A more detailed description of this procedure appears in Sections IV and V.

The synthesis of our mosaicing results for both static and dynamic scenes is then achieved by the optimization of various energy functions, each responsible for ensuring that relevant constraints are satisfied in different parts of the mosaic. The construction of R_o makes use of the correspondence constraint implemented in (1). The static contents of R_{non} are addressed using the smooth connection criterion of (5), whereas the dynamic foreground contents of R_{non} employ the consistent motion perception criterion, defined by (18). It also bears mention that the synthesis techniques used for a static scene are insufficient for the more complex situation where dynamic foreground objects are present, as explained in Section V-C.

IV. Depth-Based Mosaicing for a Static Scene

In order to overcome the parallax-related issues discussed above, we introduce our DBM method [38]. This uses a camera projection procedure combined with depth estimates generated from a virtual mosaicing camera to construct the final panorama.

Theoretically, the virtual camera can be placed at any position between the input cameras. However, for simplicity and without loss of generality, we usually choose a position coincident with either of the inputs. For illustrative purposes, of course, this choice of virtual camera position offers the benefit of allowing a comparison against a mosaic generated from a ground-truth depth map, as demonstrated in Section VI.

The virtual camera parameters that are used to build the output panorama are smoothly interpolated between those of the input cameras. However, because the output mosaic has a larger FOV than any of the sources, the corresponding elements in the internal parameter matrix of the virtual camera must be accordingly adjusted. Without loss of generality, we only consider the increase in the image size along the horizontal (x) direction.

The synthesis of the mosaic in the overlapping regions, i.e., the estimation of color and depth information for every pixel in R_o, is performed using the plane sweep algorithm [15] with *graph cut* [7] optimization. The synthesis of the mosaic in R_{non} is achieved by the propagation of reliable depth estimates from R_o into neighboring areas where no such information is available, using a smooth-appearance connection criterion.

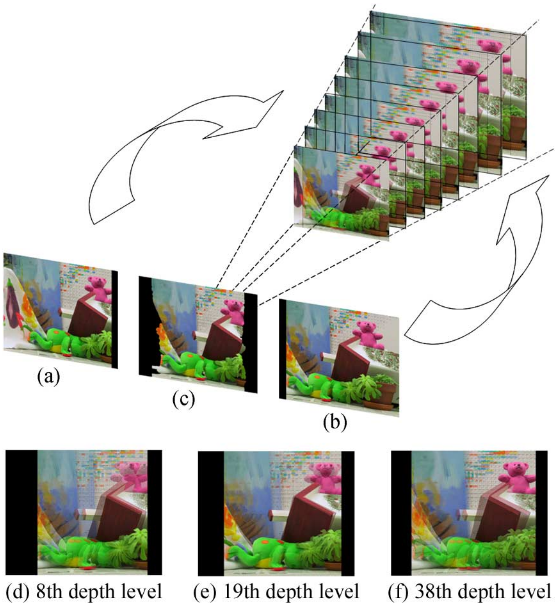


Fig. 2. Illustration of the plane sweep algorithm. Given two input images (a) and (b), we first warp them onto the parallel sampling planes, stacked at different depth levels along the z-axis. Comparing the images of these sampling planes, i.e., the intermediate images (d)–(f), we synthesize a virtual image (c) at a position between the inputs. Note that different objects are in focus at the different depth levels, where their projections coincide well with their real 3-D positions.

A. Synthesis of Overlapping Regions

Among various image-based rendering algorithms, we choose to use the plane sweep [15], [27], [47] to synthesize the overlapping regions R_o. This was primarily motivated because it divides the space into layered depth levels, which also proves valuable for the synthesis of R_{non}, where stereo information is not available. This is discussed in further detail in Section IV-B. Moreover, although we have not yet done so ourselves, the plane sweep algorithm can benefit from implementation on a programmable graphics processing unit (GPU) [54], [55], which may suffice to permit the real-time operation of our processing pipeline, even for high-definition video input.

Using the plane sweep algorithm, inputs are first projected onto parallel planes located at different depths to generate a set of N intermediate images, $\{I_{d_i}\}_{i=1}^{N}$, as illustrated in Fig. 2(d)–(f). Each intermediate image represents a sampling plane that discretizes the 3-D space at depth level d_i, as observed by the virtual mosaicing camera. The number of planes we use to divide the depth range can be varied depending on the desired accuracy and the allowable computational expense. For the experimental results provided here, we use 200 planes, which results in interplane separations in the range of 5–10 mm.

Each intermediate image contains RGB color channels, which are computed by weighted pixel-wise averaging of warped source images. Every intermediate image also contains an associated matching score channel that measures the similarity between the projections from the input images.

Let P denote the set of pixels in R_o of the output mosaic image and L be the set of depth levels $\{d_1, \ldots, d_N\}$. The problem of building the mosaic in R_o is defined as follows.

Problem 1: Given N intermediate images, find labeling $f : P \to L$ that assigns each output pixel a proper depth level $d_{i,i\in[1,N]}$ so that the energy of labeling f is minimized.

The energy of the labeling is defined as

$$E(f) = E_{\text{data}} + E_{\text{smoothness}}. \tag{1}$$

The first term, i.e., E_{data}, measures the cost of assigning the set of depth labels to pixels in R_o of the mosaic image. Generating the mosaic is, in essence, a procedure that maps pixels from the output mosaic image to their correspondences in intermediate images. In our application, E_{data} is related to the matching cost in R_o of the mosaic as

$$E_{\text{data}} = \sum_{p\in R_o} A(p). \tag{2}$$

Here, $A(p)$ is defined as

$$A(p) = \begin{cases} \phi\left(p, d(p)\right), & \text{if } \phi\left(p, d(p)\right) < \tau \\ \tau, & \text{otherwise} \end{cases} \tag{3}$$

where $\phi(p, d(p))$ is the matching cost of pixel p in the intermediate image $I_{d(p)}$ if p is assigned the depth value $d(p)$, and τ is a constant penalty for occluded pixels, empirically chosen based on the color distribution of input images.[4] The matching score is calculated based on the aggregated sum of squared differences between the projections over supporting windows. To obtain a similarity measurement that is robust not only in the textureless regions but also across depth boundaries, we apply the approach of multiresolution aggregation, as described in [55]. A lower matching score represents a higher possibility of having the pixel correspond to an object point right on the sampling plane at the assigned depth level.

The second term $E_{\text{smoothness}}$ measures the cost of assigning depth values to a pair of neighboring pixels and is used to indicate the smoothness of the depth transition from 1 pixel to its neighbors. This assumes that neighboring pixels with similar colors should typically have similar depths as well, i.e.,

$$E_{\text{smoothness}} = \sum_{p\in R_o}\left(\sum_{\{q\in N_p | d(p)\neq d(q)\}} V(p,q)\right) \tag{4}$$

where (p, q) is a pair of neighboring pixels with different depth values. N_p is a neighboring system around pixel p, such that $|x_p - x_q| + |y_p - y_q| = 1$. $V(p, q)$, which we take from Boykov *et al.* [8], is a monotonically increasing function of the color difference between the pair of pixels (p, q). $V(p, q)$ generates a high penalty value when two neighboring pixels happen to have similar colors but different depth values.

Using the energy definition of (1), an alpha-expansion *graph cut* algorithm [7], [8] is applied to find the optimal labeling f. Then, for a certain pixel p in the output image, the color of its corresponding pixel in the intermediate image at depth $f(p)$ is applied to the final synthesized virtual image, as illustrated in Fig. 2(c). This renders the set of pixels constituting R_o, as shown

[4] A pixel with matching cost exceeding τ is considered to be occluded. The value of τ should be larger than the maximal matching cost of visible pixels to ensure that the energy function penalizes occlusions appropriately and avoids labeling them with a valid depth value.

in Fig. 2(c). The remaining nonoverlapping regions R_{non} are filled in through the algorithm discussed in the following section.

B. Synthesis of Nonoverlapping Regions

The synthesis of nonoverlapping regions R_{non} in the mosaic result must be calculated by a different method from that applied to R_o, since only the latter can make use of stereo correspondences. We take advantage of the observation that depth discontinuities rarely occur in regions of uniform texture but typically coincide with color segment boundaries [48], [58]. Thus, starting from color segmentation of the source images, reliable depth information of color segments in R_o can be propagated to adjacent color segments in R_{non}, ensuring that the results preserve a smooth-appearance connection between them.[5]

Input images are first decomposed into color segments using the mean shift segmentation algorithm, incorporating edge information, as proposed in [18]. These color segments are then divided into two groups, depending on whether they contribute to R_o, for which depth estimates are available, or R_{non} otherwise. The assignment of segments that straddle both R_o and R_{non} is determined by the percentage of pixels in the segment for which depth estimates are available, as obtained from the plane sweep algorithm.[6] An illustrative segmentation result of the data in [43] is shown in Fig. 7(b), where hashed color segments represent R_{non} in the output mosaic.

Next, the color segments of R_{non} are mapped to their corresponding segments in the intermediate images $\{I_{d_i}\}_{i=1}^{N}$ from the plane sweep operation. Due to the lack of available $3-D$ information for establishing the shape functions in nonoverlapping regions, we naively assume that each color segment in R_{non} is of uniform depth. While this may not be true in practice, our results suggest that the mosaic results based on this assumption are nevertheless perceptually correct.

Let S denote the set of color segments $\{s_1, s_2, \ldots, s_M\}$ in R_{non} and L be the set of depth levels $\{d_1, \ldots, d_N\}$. Based on the assumption of uniform depth, the problem of synthesizing the nonoverlapping regions in the output mosaic is defined precisely as follows.

Problem 2: Given N intermediate images, find a minimum energy labeling $\rho : S \rightarrow L$ assigning each color segment in the nonoverlapping regions of the output mosaic a proper depth level d_i.

The energy of the labeling is defined as

$$E(\rho) = E_{\text{smoothness}} + E_{\text{occlusion}}. \tag{5}$$

The first term, i.e., $E_{\text{smoothness}}$, evaluates the overall connection cost for color segments in R_{non} defined as

$$E_{\text{smoothness}} = \sum_{i=1}^{M} \sum_{(p,q)\in\Psi_i} C_i(p,q) \tag{6}$$

where $\sum_{(p,q)\in\Psi_i} C_i(p,q)$, with respect to one color segment $s_i \in S$, is equal to the total connection cost of all pairs of neighboring pixels (p,q) within region Ψ_i, which represents border areas between segment s_i and its neighbors that already have depth estimates.[7] This smooth-appearance connection criterion between neighboring color segments plays an important role in determining the quality of the output mosaic.

In (6), the connection cost of one pair of neighboring pixels is calculated as

$$C(p,q) = \left|I_{D(p)}(p) - I_{D(q)}(p)\right|^2 + \left|I_{D(p)}(q) - I_{D(q)}(q)\right|^2 \tag{7}$$

where $D(p)$ and $D(q)$ are the depths of p and q, respectively. $I_{D(p)}$ refers to the intermediate image at depth $D(p)$, and $I_{D(p)}(p)$ represents the intensity of pixel p in this image. The transition between p and q is smooth when the local region containing this pair of pixels in the output mosaic resembles the corresponding areas in the intermediate images. $C(p,q)$ is thus minimized when $I_{D(p)}$ and $I_{D(q)}$ exhibit similar texture around (p,q). Once the smooth texture connection is preserved between neighboring color segments, the depth information can be propagated into R_{non} step by step.

If, at every possible depth level, the smooth connection cost $\sum_{(p,q)\in\Psi_i} C_i(p,q)$ of color segment s_i exceeds a threshold[8] or if the number of neighboring pixel pairs in its Ψ_i region is insufficient to generate a reliable depth estimate, s_i is defined to be *occluded*. Accordingly, its smooth connection cost in (6) is set to zero.

In order to ensure that the energy function in (5) is not biased toward a trivial optimum by considering all color segments as occluded, i.e., $E_{\text{smoothness}} = 0$, we assign a constant penalty λ_{occ}[9] to every occluded segment. The second term, i.e., $E_{\text{occlusion}}$, in (6) accounts for the case of occlusion as

$$E_{\text{occlusion}} = \sum_{i=1}^{M} P_{\text{occ}}(s_i) \tag{8}$$

where

$$P_{\text{occ}}(s_i) = \begin{cases} \lambda_{\text{occ}}, & \text{if } s_i = \text{occluded} \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

Finding the optimal labeling that minimizes the energy function is a nontrivial problem. Given M color segments, each of which has N distinct depth levels, there are a total of M^N possible solutions to labeling ρ. Such a large solution space is indicative of the complexity of this optimization problem. Moreover, the approach of spreading depth estimates from overlapping regions into neighboring nonoverlapping regions implies a slow updating procedure. These factors suggest that the optimization of the energy function by global search would be inordinately expensive.

[5] The smooth appearance connection criterion guarantees resemblance between local regions in the inputs and their corresponding locations in the resulting mosaic, particularly where patches from different inputs are merged. The definition of this criterion is inspired by the concept of seamless image composition [2], [28].

[6] For our purposes, we assign the segment to R_o if the plane sweep succeeded in establishing depth estimates for the majority of its pixels.

[7] Ψ_i contains the pairs of (p,q), where p belongs to the border of s_i and $q \in N(p)$ that already has the depth values.

[8] We choose a threshold of 100, which corresponds to the value of λ_{occ}, as explained below. A connection cost exceeding this value indicates that the segments do not match.

[9] This is simply a positive value, chosen empirically, depending on the color distribution of the input images. In our case, $\lambda_{\text{occ}} = 100$.

However, the procedure that propagates reliable depth estimates from R_o into R_{non} exhibits an *optimal substructure*,[10] suitable for a *greedy algorithm* implementation. Furthermore, as demonstrated below in Section VI, a locally optimal solution, calculated by a greedy algorithm, can produce reasonable mosaic results for different data sets.

The greedy algorithm starts with $S_{M1} = \{s_1, s_2, \ldots, s_{M1}\}$, i.e., the group of color segments immediately adjacent to the overlapping regions, and calculates labeling $\rho_{M1} = \{\rho_{M1}(s_i), i \in [1, M1]\}$ that minimizes their energy, as defined in (5). The initial depth value for each $s_i \in S_{M1}$ is *occluded*, and the initial energy of the entire group is $E(\rho_{M1}) = M1 \cdot \lambda_{\mathrm{occ}}$. The calculation proceeds as follows, repeating until either the change of labeling energy between iterations becomes insignificant or the number of iterations exceeds a specified threshold.

1) For each color segment $s_i \in S_{M1}$, all its depth candidates $d_j \in L$ are tested, and the corresponding $E(\rho_j) = E(s_i \rightarrow d_j)$ values are calculated with the depth estimates of all the other color segments fixed. The best depth value for s_i that minimizes the labeling energy is then chosen and recorded.
2) Once the best depth candidate is found for each color segment in the group, if the new depth assignments reduce the total labeling energy $E(\rho_{M1})$, the depth estimates and labeling ρ of the entire group are accordingly updated. Otherwise, these are left unchanged.

This process is applied in a similar manner to the neighbors of the previously processed group of segments, for which depth estimates have not yet been computed. This continues until no unprocessed color segments remain. Once all the depth estimates are obtained, the final panorama in R_{non} is rendered by copying the corresponding color segments from intermediate images into the mosaicing image plane.

V. Depth-Based Dynamic Mosaics

We now elaborate on the generation of a perceptually correct mosaic video for a scene that includes moving objects.

Obtaining accurate image alignment or registration remains difficult when the depth variance in the scene causes obvious parallax effects in the inputs. Furthermore, it is difficult to preserve the temporal coherence of these image registration results when objects at different depths are in motion.

To cope with these issues, our DBM approach performs a segmentation of foreground and background layers using a mixture-of-Gaussians (MoGs) model and then projects these layers separately onto the mosaicing plane according to their respective depth estimates. This guarantees both the temporal and spatial coherences of the resulting mosaic video.

A. Foreground–Background Segmentation

The segmentation process, which, for our purposes, consists of separating moving objects from the static background, can be based on motion, depth, or stochastic background models. A successful segmentation algorithm should exhibit robustness to both sudden and gradual variance of illumination, be adaptive to the nonstationary background content, and provide a clear description of the background scene, even when confronted with dynamic foreground objects in the training data. An ideal motion-based segmentation algorithm, which satisfies all of these requirements, is hard to achieve. Furthermore, the dependence of depth-based segmentation algorithms on stereo information prevents them from processing inputs corresponding to nonoverlapping regions of the scene. As a result, we favor the use of the stochastic background model. This proves particularly well suited to distinguish between the rapidly changing foreground contents and the slowly varying background scene, even under the challenging conditions of varying illumination and limited overlap between source cameras. In our paper, we apply an improved version of the MoGs algorithm [46] in [29], which uses an adaptive, rather than fixed, learning rate for each Gaussian to obtain improved convergence speed without sacrificing stability. This is explained in further detail below.

1) MoGs Background Model Construction: Suppose all the pixels from the frames in a given time interval satisfy the distribution of a MoGs background model. In this case, the probability that a pixel assumes a value X is given by [46]

$$P(X) = \sum_{i=1}^{K} \omega_i P(X|G_i) \tag{10}$$

where

$$P(X|G_i) = \frac{1}{\sqrt{(2\pi)^n |\Sigma_i|}} \exp -\frac{1}{2}(X - \mu_i)^T \Sigma_i^{-1} (X - \mu_i) \tag{11}$$

where μ_i is the mean of the ith Gaussian, Σ_i is the diagonal covariance matrix, and ω_i is its weight.

When training the background model, each pixel value is compared with the available models. If the value can be represented by an element of the MoGs, it is used to update the model. Otherwise, the least likely Gaussian element, i.e., with the smallest ω_i, is replaced by a new Gaussian initialized with the new pixel value. Our present implementation uses $K = 3$ Gaussians.

2) Segmentation: The probability that each pixel X of a given frame belongs to background B is calculated using the trained MoGs background model G_i, $i = [1, K]$, i.e.,

$$\begin{aligned} P(X_t) &= \sum_{i=1}^{K} P(B|G_i) P(G_i|X) \\ &= \frac{\sum_{i=1}^{K} P(X|G_i) P(G_i) P(B|G_i)}{\sum_{j=1}^{K} P(X|G_j) P(G_j)} \end{aligned} \tag{12}$$

where $P(G_i) = \omega_i$ and $P(B|G_i)$, $i = [1, K]$, is defined in [29]. If probability $P(B|X)$ exceeds some threshold, the pixel is considered as an element of the background and, otherwise, as a foreground (or nonstatic) object. This process produces a binary mask, with white pixels representing possible foreground regions.

After thresholding and shadow removal, the raw mask may contain many isolated foreground points, as illustrated in the example in Fig. 3(c). Combining this mask with the color segmentation of the input frame, as shown in Fig. 3(b), we integrate these isolated points into a group of color segments contributing

[10] A problem exhibits optimal substructure if an optimal solution to the problem contains optimal solutions to its subproblems.

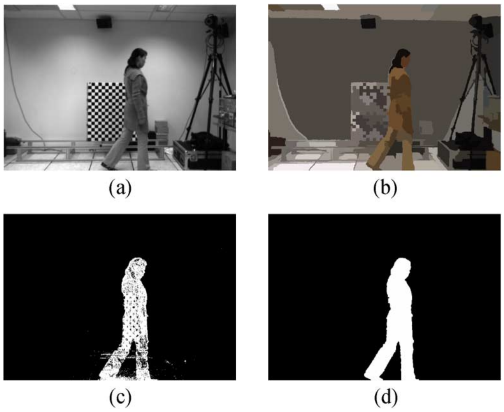


Fig. 3. An illustration of foreground–background segmentation, resulting in a cleaned mask, integrating color information. (a) Original frame. (b) Color segmentation. (c) Raw mask of the foreground layer. (d) Final cleaned mask.

to a silhouette with clean boundaries closely matching the outlines of the foreground object, as illustrated in Fig. 3(d). This silhouette will later assist in the generation of foreground mosaicing frames.

3) Background-Image Generation: Whereas the complement of a silhouette is incomplete wherever there are occlusions from foreground objects, the *background image* provides a full description of the static regions in the environment. It is computed as a weighted average of the means of each Gaussian element of the MoGs as the expected value $E[X|B]$ of observation X, assuming it to be the background [29], i.e.,

$$\begin{aligned} E[X|B] &= \sum_{i=1}^{K} E[X|G_i]P(G_i|B) \\ &= \frac{\sum_{i=1}^{K} \mu_i P(B|G_i)P(G_i)}{\sum_{j=1}^{K} P(B|G_j)P(G_j)} \end{aligned} \tag{13}$$

where μ_i represents the mean of the ith Gaussian G_i.

Two of such background images are shown in Fig. 4(b) and (c). These are used to construct the background mosaic that is common to all frames, as explained in the next section.

B. Mosaic Construction of Background Layer

Using the DBM technique introduced in Section IV, both overlapping and nonoverlapping regions of the mosaic are rendered as if seen by a virtual camera that has all of its parameters fixed during the synthesis of the entire dynamic mosaicing video.

Fig. 4 illustrates the background images and the depth-based mosaic based on them. Because the dynamic foreground objects have been removed, they exercise no influence over the image alignment process, and thus, the appearance of the background in the mosaic video remains static over time. In our tests, a multiband blending strategy [12] is required to balance the color differences between cameras. This smooths the boundary between overlapping and nonoverlapping regions in the background mosaic.

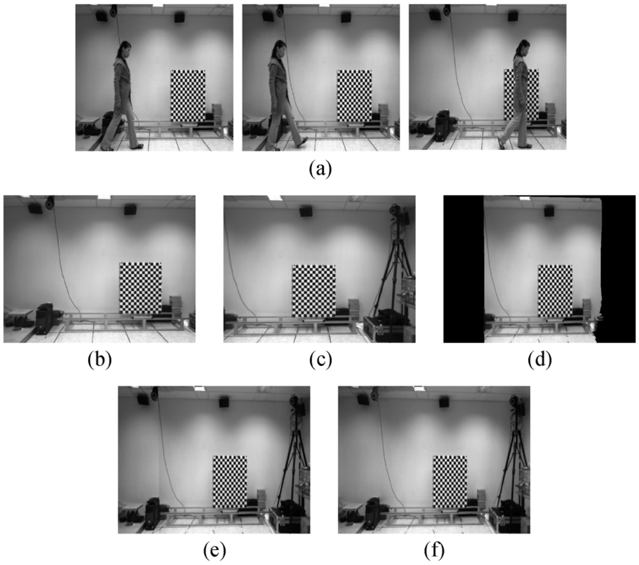


Fig. 4. Mosaic construction of background layer. (a) Sample training frames. (b) Left camera background image. (c) Right camera background image. (d) R_o of the background mosaic. (e) Background mosaic with R_{non} filled. (f) After applying multiband blending.

C. Rendering Foreground Mosaic

As before, we divide the foreground contents of the video input into R_o and R_{non} based on whether these are observed by both or only one of the input cameras. However, unlike the case for static scene content, as described in Section IV, we can perform this division simply by analyzing the areas of the foreground silhouettes between input cameras.[11]

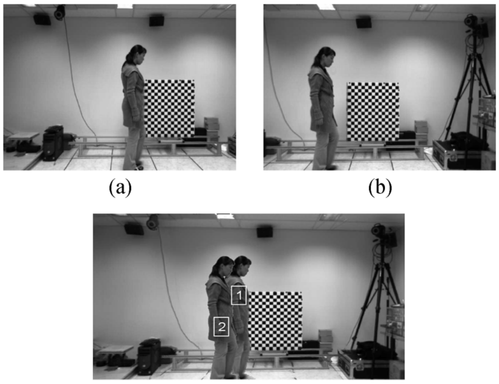
The foreground mosaic in R_o, as observed by the virtual mosaicing camera, can be synthesized in the same manner as in the static case. Using the associated depth estimates, we project these foreground contents onto the proper positions in the mosaicing image plane.

However, the challenge arises when dealing with a dynamic foreground object that moves, without loss of generality, from R_{non} into R_o. While the object is in R_{non}, the mosaic sequence would be generated from the contents of one camera using the smooth-appearance connection criterion used for static contents in R_{non}, as described in Section IV-B. At some point following the entry into R_o,[12] it is necessary to switch to input from another camera, and the mosaic sequence then continues using the stereo-based algorithm described in Section IV-A. However, because of the parallax between the camera views, i.e., one that covers R_{non} and the second used when the switch between cameras occurs in R_o, the transition between the two mosaicing algorithms typically results in inconsistency of both appearance and motion.

This problem is illustrated in Fig. 5(c). Ideally, the foreground layer from both sources should be projected to the identical position in the mosaicing frame, but in this case, the foreground layer from the left source projects onto position 1, whereas that from the right source projects onto position 2. The effects can be observed as jitter in the foreground objects in the output mosaic.

[11] The standard deviation measures the variability of the data.

[12] We assume that the object continues moving further away from R_{non}.



(a) (b)

(c)

Fig. 5. Illustration of parallax effects when projecting from different sources onto the foreground layer. The parameters of the virtual mosaicing camera are chosen to coincide with the right source camera. (a) Left source image. (b) Right source image. (c) Projection of foreground layers onto the background mosaic.

This motivates the use of a different process to estimate depth values and project the foreground dynamic objects in R_{non} onto the mosaicing plane.

1) Motion Perception and its Mathematical Representation: View synthesis techniques based on stereo information are obviously inapplicable when depth values cannot be computed, as in the case of foreground contents in R_{non}. Moreover, as noted in Section II, the 3-D reconstruction of a scene from the monocular video input is, at best, a difficult problem. However, an exact reconstruction may not be necessary to achieve perceptually correct results.

Previous studies of motion perception reveal that humans are sensitive to significant physical feature changes, i.e., *speed* and *direction* in motion trajectories [23]. Observers understand motion activities based on when and where these changes occur [57]; moreover, the order of these events is another important cue for perception [34]. Rao *et al.* [40] translated these results into a mathematical representation. A 2-D trajectory, which represents the path of an object in a video sequence, is defined as a spatiotemporal curve with the following function:

$$r(t) = [x(t),\ y(t),\ t] \quad 1 \le t \le n \tag{14}$$

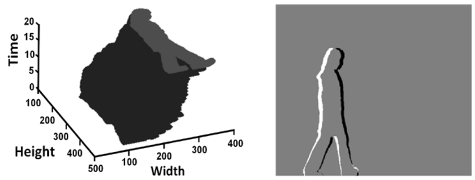
where t represents the frame index and $[x(t), y(t)]$ indicates the pixel coordinates of the object centroid in the tth frame. Curvature $k(t)$, which is responsive to discontinuities in velocities and accelerations of $r(t)$, is given by

$$k(t) = \frac{||r'(t) \times r''(t)||}{||r/(t)||^3} \tag{15}$$

where $r'(t)$ or $r''(t)$, i.e., the first and second derivatives of $r(t)$, represent its velocity and acceleration, respectively.

Significant changes of physical motion features, referred to as *instants*, are defined as the maxima in the curvature of a 2-D trajectory [40]. Each trajectory can be decomposed into a sequence of such instants, i.e., the mathematical representation of motion that is understood and distinguished by humans.

As long as all of the cameras remain in the same half-hemisphere of the viewing space, instants are independent of the view direction. In other words, although the 2-D trajectories of the motion may differently appear from various camera positions, they share a common sequence of instants, and thus, an identical perception of object movement.



(a) (b)

Fig. 6. The velocity of a 2-D motion trajectory is related to the difference between two successive frames (b), where regions are evaluated with values of (white) -1, (black) 1, and (gray) 0. (a) History of silhouettes as a 2-D trajectory. (b) Difference between two successive silhouettes.

For simplicity, we ignore the depth variance of foreground objects, treating them as parallel to the image plane of the viewing camera. Consequently, constructing a foreground-mosaic video in R_{non} can be naively performed. We determine a suitable depth estimate d for each foreground object at every time t, as described in the next section, and replicate velocity $r(t)$ from the monocular input video by projecting these objects onto the mosaicing image plane according to their depth estimates. In this manner, curvature and, thus, motion consistency, in both spatial and temporal domains, are preserved in the mosaic.

We note that the output constructed in this manner does not faithfully emulate the projection of 3-D motion that would be seen by the virtual camera. However, as noted above, it is computationally expensive, if not impossible at times, to generate reliable 3-D information when presented only with the monocular input video. Thus, we pursue a more practical alternative to estimating the 3-D shape and position of moving foreground objects in R_{non}. Instead, we consider the task of generating a representation of foreground motion in a dynamic video that provides spatial- and temporal-motion consistency as objects move across the FOVs of different cameras.

We now describe the details of our novel dynamic mosaicing algorithm for the foreground contents of R_{non}. Our method generates perceptually correct results by propagating reliable foreground layer depth information from overlapping regions into neighboring nonoverlapping regions, in such a manner that the sequence of motion instants is preserved.

2) Foreground-Mosaic Generation in R_{non}: Let L denote the set of depth levels $\{d_1, \ldots, d_N\}$. In our application, the 2-D motion trajectories in the input video $r(t)$ and the synthesized mosaic video $r_{\text{mosaic}}(t)$, are represented by the history of silhouettes, i.e., $H(t)$, as shown in Fig. 6. Velocity $v(t)$ of a 2-D trajectory is related to the first derivative of the silhouette sequence, i.e., $\Delta H(t) = H(t+1) - H(t)$, as shown in Fig. 6(b). The speed of these trajectories is given by the sum of two area ratios, i.e.,

$$\text{speed}\,(\Delta H(t)) = \frac{\sum_{p(i,j)=1} \Delta H(t)}{\sum_{p(i,j)=1} H(t+1)} + \frac{\sum_{p(i,j)=-1} |\Delta H(t)|}{\sum_{p(i,j)=1} H(t)} \tag{16}$$

where $p(i,j)$ refers to the value of pixel (i,j) in the silhouette. Thus, speed$(\Delta H(t))$ depends on the area ratios between the regions of removed and newly exposed pixels to the entire silhouette of the object. The direction of velocity is defined as

$$\vec{\gamma}(t) = \text{sgn}\left(\text{CM}_1(t) - \text{CM}_{-1}(t)\right) \tag{17}$$

where CM_1 is the centroid of 1-valued regions, CM_{-1} is that of -1-valued regions in $\Delta H(t)$, and $\text{sgn}(\mathbf{x})$ of vector $\mathbf{x}$ returns a vector composed of the sgn values of each of its elements.

The problem of generating the tth depth-based foreground-mosaic frame in R_{non} can be now stated as follows.

Problem 3: Given a monocular input video of a foreground object in R_{non} and assuming that the object is of uniform depth, find the best depth estimate $d(t)$ so that the difference of velocity, i.e., $\text{dif}_v(t)$, between $r(t)$ and $r_{\text{mosaic}}(t)$ is minimized.

The difference of velocity, i.e., $\text{dif}_v(t)$, is defined as

$$\text{dif}_v(t) = \text{dif}_{\text{speed}}(t) + \text{dif}_{\text{direction}}(t). \tag{18}$$

The first term, i.e., $\text{dif}_{\text{speed}}(t)$, measures the difference in speed between $r(t)$ and $r_{\text{mosaic}}(t)$, i.e.,

$$\text{dif}_{\text{speed}}(t) = \text{speed}\left(\Delta H(t)\right) - \text{speed}\left(\Delta H_{\text{mosaic}}(t)\right) \tag{19}$$

where speed$(\Delta H(t))$ is defined in (16). speed$(\Delta H_{\text{mosaic}}(t))$, based on $r_{\text{mosaic}}(t)$, is constructed in the same manner.

The second term in (18) accounts for the difference between directions of velocities in $r(t)$ and $r_{\text{mosaic}}(t)$, i.e.,

$$\text{dif}_{\text{direction}}(t) = \|\vec{\gamma}(t) - \vec{\gamma}_{\text{mosaic}}(t)\| \cdot \lambda_{\text{direction}} \tag{20}$$

where $\lambda_{\text{direction}}$ is a constant penalty.[13] $\text{dif}_{\text{direction}}(t)$ generates a cost value proportional to the measurement of direction differences between the velocities of $r(t)$ and $r_{\text{mosaic}}(t)$. If the velocities coincide in all directions, $\text{dif}_{\text{direction}}(t)$ is zero.

Without loss of generality, we test all the depth candidates starting from frame t, immediately prior to the entry of a foreground object into the overlapping region, where reliable depth estimates are already available. The value that provides the best match of motion velocity between $(H_{\text{mosaic}}(t+1), H_{\text{mosaic}}(t))$ and $(H(t+1), H(t))$ and consequently minimizes the energy cost in (18) is selected. We then apply that same calculation to estimate the foreground mosaic at time $(t-1)$ and continue until the first frame of the sequence is reached. In this manner, the foreground-mosaic video in R_{non} is created over the time domain $[0, t]$. A simple merging of the background mosaic [e.g., see Fig. 4(f)] with the foreground mosaic results in the final dynamic mosaic video.

VI. Experimental Results

As a well-known representative of conventional mosaicing techniques, we use Autostitch [10] to compare and evaluate the quality of our algorithms. This is applied to data sets in [42], [43], and those we have collected ourselves, all of which contain structured indoor scenes that exhibit a wide range of depth distribution. Camera calibration was performed using Zhang's method [59], as implemented in [6] for our own data and by the *structu*re*-from-motion* algorithm [21] for externally supplied data.

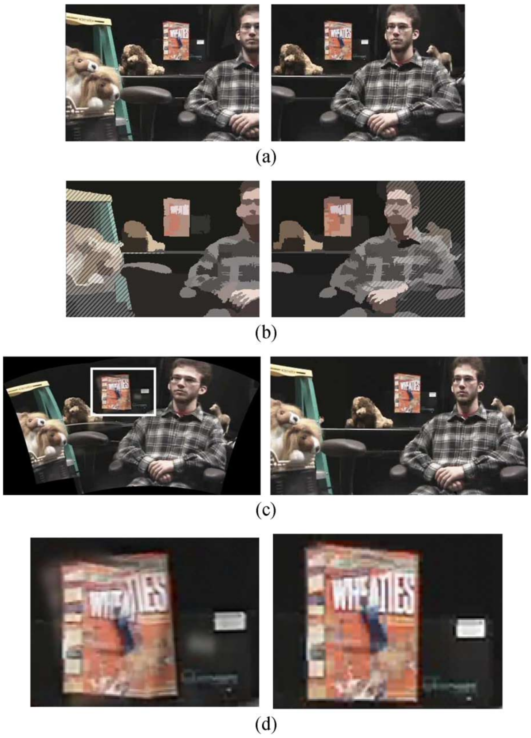


Fig. 7. DBM results using the stereo data set from [43]. Hashed regions of (b) contribute to nonoverlapping regions in the final mosaic. (a) Left and right input images. (b) Color segmentation of left and right images. (c) Mosaicing results of (left) Autostitch exhibit (in overlaid rectangles) distortion and artifacts compared with (right) our DBM results. (d) Magnified region of results from (c).

Our own data set was generated from two synchronized video cameras with a wide baseline and included a single moving person in the foreground. However, the results can be extended in a straightforward manner to take advantage of additional cameras. It is not necessary to rectify the inputs; indeed, some panning or tilting rotations are acceptable as long as the configuration is predominantly translational.

A. DBM of Static Scenes

First, we validate the DBM on static scenes. In the results shown in Figs. 7 and 8, Autostitch produces artifacts in its mosaic results when provided two sparse samples of a scene with complex depth variation. The first type of artifact, i.e., *distortion*, is mainly caused by disparity variance between objects at different depths from the camera. Autostitch has to deform the input images, compressing close objects and expanding distant objects, to equalize their respective disparities. The normalized amount of overlap between the images allows for a smooth combination of the two deformed inputs. As shown in Fig. 7(c), such distortion errors (e.g., the "twisted" human subject) are noticeable.

The second type of artifact, i.e., *ghost errors*, seen in the highlighted bounding boxes, results from the difficulty of achieving accurate image alignment, even after compensating for disparity differences, as previously described. Ultimately, this is due to

[13] In our case, λ is chosen to ensure balance between the contributions of speed and direction components to the measurement of differences of velocity.

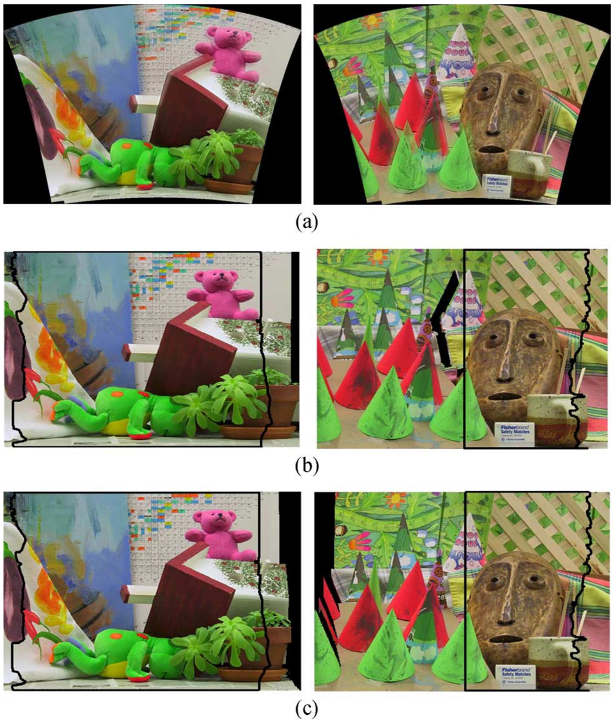


Fig. 8. Comparison of mosaic results. The black lines in (b) and (c) denote the boundaries between overlapping and nonoverlapping regions. (a) Mosaicing result of Autostitch exhibit distortion and artifacts. (b) DBM result, generally free of such artifacts. (c) Reference DBM result using ground-truth depth values of (left) Teddy and (right) Cone data sets.

the depth variance in the scene and the significant view disagreement between two cameras with a wide baseline. In contrast, the results of our DBM algorithm are free of such artifacts, as illustrated in Fig. 7(d).

For further comparison, reference mosaics generated with the DBM method using the ground-truth depth values from the Teddy and Cone data sets are shown in Fig. 8(c). The position of the virtual mosaicing camera is chosen to coincide with the left source, with intrinsic parameters adjusted to provide a wider FOV. The overlapping regions in our mosaic results, i.e., the regions within the black boundaries in Fig. 8(b), exhibit reasonable coherence with the reference mosaic. However, slight appearance differences due to the variance of depth estimates are observed. Specifically, the right side of the nonoverlapping regions, i.e., outside of the black boundaries, exhibits some expansion relative to the reference mosaic but does not otherwise exhibit any perceptual disagreement with the reference mosaic, as shortly discussed.

It is worth noting that, unlike image-based rendering algorithms [44], our DBM approach does not attempt to determine the real depth in nonoverlapping regions. Indeed, using the naive assumption of the uniform depth of each color segment in R_{non}, the estimates produced by the DBM method do not conform to the ground-truth topology of most scenes. Nevertheless, the depth estimates obtained with a smooth-appearance connection criterion guarantee sufficient resemblance between local regions in the mosaic and those of the inputs. Most importantly, the resulting outputs are perceptually acceptable.

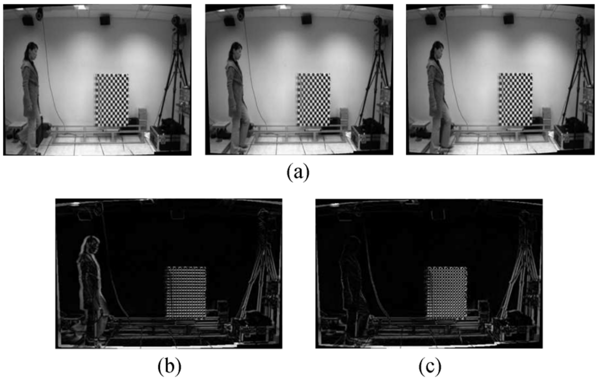


Fig. 9. Illustration of jitter in the dynamic mosaic results of Autostitch. The difference images (b) and (c) exhibit changes of content not only caused by the object motion but also by the jitter of static background regions. (a) Successive frames F_{21}, F_{22}, and F_{23} produced by Autostitch. (b) $|F_{22} - F_{21}|$. (c) $|F_{23} - F_{22}|$.

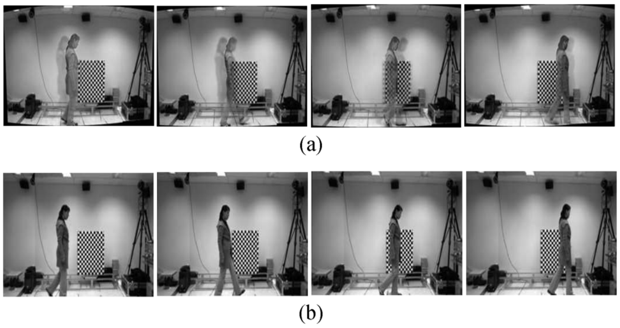


Fig. 10. Frames generated by (b) our algorithm are free of the ghost errors exhibited by Autostitch. (a) Autostitch results with ghost errors. (b) Corresponding frames of our DBM algorithm.

B. DBM of Dynamic Scenes

The depth-based dynamic mosaicing method was tested on our own data. Since Autostitch was not designed for video sequences, its results exhibit interframe jitter effects in response to the moving objects in the scene. Fig. 9 illustrates the difference images between pairs of successive mosaicing frames. These differences arise from nonrigid movements of dynamic objects, as well as from certain areas in the static background regions with inconsistent image registration between neighboring frames. In contrast, our algorithm applies a common mosaicing background [see Fig. 4(f)] in each frame of the dynamic mosaic and thus avoids inducing jitter.

More importantly, parallax effects in the inputs lead to ghost errors observed in the output of Autostitch, as illustrated in Fig. 10(a). As summarized in Section II, this also poses a challenge for conventional dynamic mosaicing algorithms. Although their improved image registration techniques may avoid jitter problems by reducing the impact of dynamic objects in the scene, they cannot overcome parallax entirely, which remains evident even in the static environment. In contrast, Fig. 10(b) demonstrates that our approach generates results free of these errors.

In order to demonstrate that our algorithm preserves motion consistency in the spatiotemporal domain, we compare our result to the output of a reference camera that was located behind the baseline between the two input cameras. This allows us to obtain a FOV similar to that of the virtual mosaicing camera.

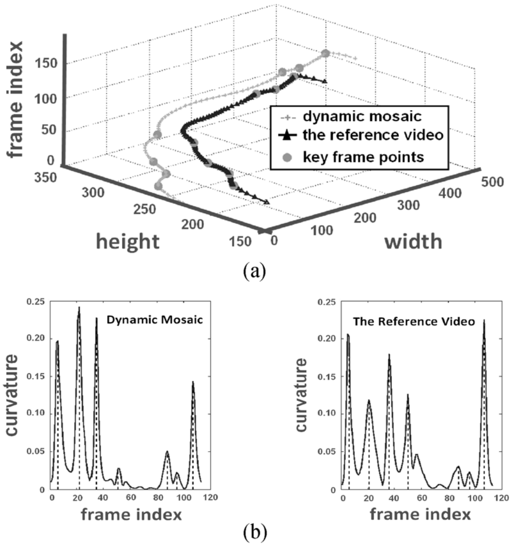


Fig. 11. (a) Motion trajectories of the reference video and the dynamic mosaic video. (b) Corresponding curvature values. The strong coherence of both the trajectories and the sequence of instants is observed between the two paths.

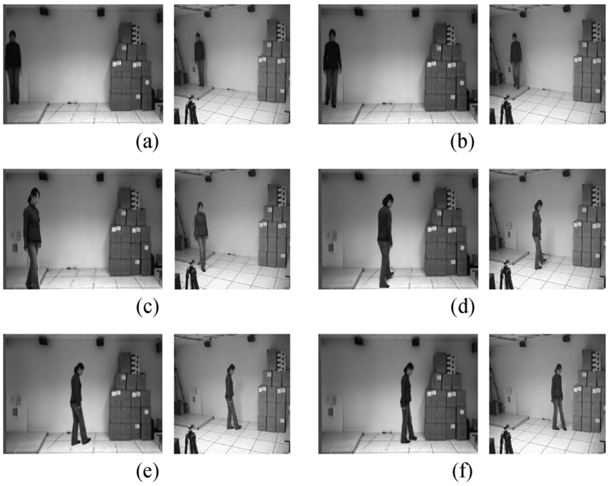


Fig. 12. Comparison of frames from the dynamic mosaic with those from the reference video. Note that the reference camera had a different position and focal length from that of the (virtual) mosaicing camera. (a) F_6. (b) F_{21}. (c) F_{60}. (d) F_{89}. (e) F_{99}. (f) F_{110}.

As illustrated in Fig. 11, the video acquired by the reference camera and the synthesized dynamic mosaicing video present similar motion trajectories, with the same sequence of instants, or maxima in the curvature of trajectory. The frame indexes corresponding to these maxima are marked on the trajectories as key frame points. The similarity of trajectories and instants between the reference video and the dynamic mosaic confirms that the latter not only preserves motion consistency in the spatiotemporal domain but also provides a similar perception of motion. Samples from both the reference video and the dynamic mosaic,[14] taken from identical frame indexes, are provided in Fig. 12 to illustrate motion consistency. Similarly, samples demonstrating the absence of ghost errors can be seen in Fig. 10.

C. Limitations

If the virtual mosaicing camera has a different position and orientation from the source camera covering the nonoverlapping regions, holes due to occlusion may occur in R_{non} of the mosaic result. These holes do not have any correspondence in the inputs and thus violate the smooth-appearance connection criterion, i.e., a local region in the output mosaic should resemble some corresponding region in the sources. Since the DBM method is based on a smooth-appearance connection criterion, filling these hole regions using DBM may prove problematic unless the holes actually correspond to textureless scene content.

We use the *Cone* data set [42] as a more challenging input to test the limitations of our algorithm with regard to occlusions. Furthermore, we deliberately restrict the algorithm from employing depth estimates from only half of the total area of R_o. This reduces the amount of the mosaicing result that is directly obtained from the plane sweep algorithm, ensuring that a greater proportion depends on propagated depth estimates. As expected, noticeable holes appear in the result [see Fig. 8(b)]. Due to the texture in the corresponding scene contents, filling these holes may result in artifacts. It should be noted that holes also appear in the reference mosaic result using the ground-truth depth values [see Fig. 8(c)].

VII. Conclusion

Conventional image mosaicing techniques must warp inputs by estimating the geometric relationship, i.e., the explicit camera motion models, so that input images are aligned in their regions of overlap. However, by incorporating depth cues, our new approach has instead applied the smooth-appearance connection criterion to ensure natural transitions between contents in panoramas of static scenes.

Notwithstanding the holes due to occlusion, our DBM algorithm has constructed significantly improved results compared with traditional algorithms, as shown in Figs. 7 and 8, with respect to its avoidance of distortions and synthesis artifacts due to image alignment errors. In general, the new DBM technique introduced in this paper has dramatically overcome the parallax problem and has produced high-quality panoramas for most real-world applications.

Moreover, we have demonstrated a successful processing pipeline that generates reasonable mosaic video containing a single moving object. Our algorithm has imposed a consistent motion perception criterion with respect to moving objects, which preserves the continuity of object movements in dynamic mosaicing videos. Experiments have demonstrated promising results, even when given challenging inputs that cause traditional image mosaicing algorithms to fail.

However, our present implementation has not addressed more complicated motion patterns, such as those involving multiple objects with occlusion and disocclusion effects, a foreground object becoming stationary, or the movement of an otherwise static object in the scene. We expect that these challenges may be addressed by taking advantage of motion tracking or 3-D models of foreground objects, obtained prior to employing the algorithm, but such investigation is left for future work.

Unfortunately, in the current implementation, shadows of moving objects have been removed. This is due to our training

[14] Illustrative video available at http://www.cim.mcgill.ca/sre/videos/mosaic/dynamicmosaic_regions.mpg

of the statistical background model, which regards such shadows as part of the static background. In order to preserve shadows and render them properly in the dynamic mosaicing output, a shadow and light source detection algorithm must be included.

Another potential area for future work is apparent when considering the energy functions summarized in Section III, required to satisfy the constraints of various regions of the mosaicing result. Developing a single "super" energy function that suffices to address all of these constraints for both static and dynamic scenes in a unified manner could well provide an improvement in the computational efficiency of our algorithm.

Our investigation of techniques to overcome parallax and motion issues has exposed opportunities for improvement and has also raised further open questions. Considerable work remains, particularly to cope with the arbitrary movement of multiple objects in the scene. Furthermore, the underlying steps of depth estimation, color segmentation of inputs, identification of overlapping and nonoverlapping regions, depth propagation in the nonoverlapping regions, and mosaicing synthesis of the dynamic foreground object are computationally demanding. At present, these preclude achieving anywhere near real-time video rates and cannot thus match the performance of mosaicing methods that ignore parallax issues. However, this might be significantly improved by taking advantage of the parallel computation abilities of a GPU or perhaps exploiting the efficient depth map generation abilities of precalibrated stereo cameras or time-of-flight cameras. We believe that these directions offer considerable opportunities for further research and will lead to exciting advances in computational photography in the years ahead.

REFERENCES

[1] A. Agarwala, M. Agrawala, M. Cohen, D. Salesin, and R. Szeliski, "Photographing long scenes with multi-viewpoint panoramas," in *Proc. 33th Annu. Conf. Comput. Graph. Interactive Techn.*, 2006, pp. 853–861.

[2] A. Agarwala, M. Dontcheva, M. Agrawala, S. Drucker, A. Colburn, B. Curless, D. Salesin, and M. Cohen, "Interactive digital photomontage," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 294–302, Aug. 2004.

[3] R. A. Alex, Y. Pritch, D. Lischinski, and S. Peleg, "Dynamosaicing: Mosaicing of dynamic scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 10, pp. 1789–1801, Oct. 2007.

[4] M. Bartoli, N. Dalal, B. Bose, and R. Horaud, "From video sequence to motion panoramas," in *Proc. Workshop Motion Video Comput., MOTION*, 2002, pp. 201–207.

[5] H. Bay, A. Ess, T. Tuytelaars, and L. J. Van Gool, "Speeded-up robust features (SURF)," *Comput. Vis. Image Process.*, vol. 110, no. 3, pp. 346–359, Jun. 2008.

[6] J. V. Bouguet, Camera Calibration Toolbox for Matlab 2003 [Online]. Available: http://www.vision.caltech.edu/bouguetj/calib_doc/

[7] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 9, pp. 1124–1137, Sep. 2004.

[8] Y. Boykov, R. Zabih, and S. Gortler, "Multi-camera scene reconstruction via graph cuts," in *Proc. 7th ECCV—Part III*, 2002, pp. 82–96.

[9] C. Bregler, A. Hertzmann, and H. Biermann, "Recovering non-rigid 3d shape from image streams," in *Proc. Comput. Vis. Pattern Recog.*, 2000, pp. 690–696.

[10] M. Brown and D. G. Lowe, "Recognising panoramas," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2003, pp. 1218–1225.

[11] M. Brown and D. G. Lowe, "Automatic panoramic image stitching using invariant features," *Int. J. Comput. Vis.*, vol. 74, no. 1, pp. 59–73, Aug. 2007.

[12] P. J. Burt and E. H. Adelson, "A multiresolution spline with application to image mosaics," *ACM Trans. Graph.*, vol. 2, no. 4, pp. 217–236, Oct. 1983.

[13] S. C. Chen, "Quicktime VR: An image-based approach to virtual environment navigation," in *Proc. 22nd Annu. Conf. Comput. Graph. Interactive Techn., SIGGRAPH*, 1995, pp. 29–38.

[14] J. Civera, A. J. Davison, J. A. Magallón, and J. M. Montiel, "Drift-free real-time sequential mosaicing," *Int. J. Comput. Vis.*, vol. 81, no. 2, pp. 128–137, Feb. 2009.

[15] R. Collins, "A space-sweep approach to true multi-image matching," in *Proc. Comput. Vis. Pattern Recog.*, 1996, pp. 358–363.

[16] C. Doutre and P. Nasiopoulos, "Fast vignetting correction and color matching for panoramic image stitching," in *Proc. IEEE Int. Conf. Image Process.*, 2009, pp. 709–712.

[17] A. A. Efros and W. T. Freeman, "Image quilting for texture synthesis and transfer," in *Proc. ACM SIGGRAPH*, 2001, pp. 341–346.

[18] B. Georgescu, I. Shimshoni, and P. Meer, "Mean shift based clustering in high dimensions: A texture classification example," in *Proc. 9th IEEE ICCV*, 2003, pp. 456–463.

[19] R. Gupta and R. I. Hartley, "Linear pushbroom cameras," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 9, pp. 963–975, Sep. 1997.

[20] C. Harris and M. J. Stephens, "A combined corner and edge detector," in *Proc. 4th Alvey Vis. Conf.*, 1988, pp. 147–152.

[21] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. New York: Cambridge Univ. Press, 2000.

[22] M. Irani, P. Anandan, and S. Hsu, "Mosaic based representation of video sequence and their applications," in *Proc. IEEE Int. Conf. Comput. Vis.*, 1995, pp. 605–611.

[23] R. J. Jagacinski, W. W. Johnson, and R. A. Miller, "Quantifying the cognitive trajectories of extrapolated movements," *J. Exp. Psychol.: Hum. Percept. Perform.*, vol. 9, no. 1, pp. 43–57, Feb. 1983.

[24] J. Jia and C. K. Tang, "Image stitching using structure deformation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 4, pp. 617–631, Apr. 2008.

[25] H. L. Jin, "A three-point minimal solution for panoramic stitching with lens distortion," in *Proc. Comput. Vis. Pattern Recog.*, 2008, pp. 1–8.

[26] M. H. Ju and H. B. Kang, "A new simple method to stitch images with lens distortion," in *Proc. Adv. Vis. Comput.*, 2010, pp. 273–282.

[27] K. N. Kutulakos and S. M. Seitz, "A theory of shape by space carving," *Int. J. Comput. Vis.*, vol. 38, no. 3, pp. 199–218, Jul./Aug. 2000.

[28] V. Kwatra, A. Schodl, I. Essa, G. Turk, and A. Bobick, "Graphcut textures: Image and video synthesis using graph cuts," *ACM Trans. Graph.*, vol. 22, no. 3, pp. 277–286, Jul. 2003.

[29] D. S. Lee, "Effective Gaussian mixture learning for video background subtraction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 5, pp. 827–832, May 2005.

[30] A. Levin, A. Zomet, S. Peleg, and Y. Weiss, "Seamless image stitching in the gradient domain," in *Proc. Eur. Conf. Comput. Vis.*, 2004, pp. 377–389.

[31] S. Lovegrove and A. J. Davison, "Real-time spherical mosaicing using whole image alignment," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 73–86.

[32] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004.

[33] A. Mills and G. Dudek, "Image stitching with dynamic elements," *Image Vis. Comput.*, vol. 27, no. 10, pp. 1593–1602, Sep. 2009.

[34] D. Newtson and G. Engquist, "The perceptual organization of ongoing behavior," *J. Exp. Social Psychol.*, vol. 12, no. 5, pp. 436–450, Sep. 1976.

[35] S. Peleg and J. Herman, "Panoramic mosaics by manifold projection," in *Proc. Comput. Vis. Pattern Recog.*, 1997, pp. 338–343.

[36] S. Peleg, B. Rousso, A. Rav-Acha, and A. Zomet, "Mosaicing on adaptive manifolds," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 10, pp. 1144–1154, Oct. 2000.

[37] K. Pulli, M. Tico, and Y. Xiong, "Mobile panoramic imaging system," in *Proc. ECVW*, 2010, pp. 108–115.

[38] Z. Qi and J. R. Cooperstock, "Depth-based image mosaicing for both static and dynamic scenes," in *Proc. 19th ICPR*, 2008, pp. 1–4.

[39] P. Rademacher and G. Bishop, "Multiple-center-of-projection images," in *Proc. 25th Annu. Conf. Comput. Graph. Interactive Techn., SIGGRAPH*, 1998, pp. 199–206.

[40] C. Rao, A. Yilmaz, and M. Shah, "View-invariant representation and recognition of actions," *Int. J. Comput. Vis.*, vol. 50, no. 2, pp. 203–226, Nov. 2002.

[41] H. S. Sawhney and S. Ayer, "Compact representations of videos through dominant and multiple motion estimation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 8, pp. 814–830, Aug. 1996.

[42] D. Scharstein and R. Szeliski, "High-accuracy stereo depth maps using structured light," in *Proc. Comput. Vis. Pattern Recog.*, Jun. 2003, pp. 195–202.

[43] S. M. Seitz and J. Kim, "The space of all stereo images," *Int. J. Comput. Vis.*, vol. 48, no. 1, pp. 21–38, Jun. 2002.

[44] H. Y. Shum, S. C. Chan, and S. B. Kang, *Image-Based Rendering*. Secaucus, NJ: Springer-Verlag, 2006.

[45] H. Y. Shum and R. Szeliski, "Construction and refinement of panoramic mosaics with global and local alignment," in *Proc. Int. Conf. Comput. Vis.*, 1998, pp. 953–960.

[46] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *Proc. Comput. Vis. Pattern Recog.*, 1999, pp. 23–25.

[47] R. Szeliski and P. Golland, "Stereo matching with transparency and matting," *Int. J. Comput. Vis.*, vol. 32, no. 1, pp. 45–61, Aug. 1999.

[48] H. Tao, S. H. Sawhney, and R. Kumar, "A global matching framework for stereo computation," in *Proc. 8th IEEE ICCV*, 2001, pp. 532–547.

[49] C. Tomasi and T. Kanade, "Shape and motion from image streams under orthography: A factorization method," *Int. J. Comput. Vis.*, vol. 9, no. 2, pp. 137–154, Nov. 1992.

[50] L. Torresani, A. Hertzmann, and C. Bregler, "Nonrigid structure-from-motion: Estimating shape and motion with hierarchical priors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 5, pp. 878–892, May 2008.

[51] M. Uyttendaele, A. Eden, and R. Szeliski, "Eliminating ghosting and exposure artifacts in image mosaics," in *Proc. Comput. Vis. Pattern Recog.*, 2001, pp. 509–516.

[52] L. Wang, J. Traub, S. Weidert, S. M. Heining, E. Euler, and N. Navab, "Parallax-free intro-operative X-ray image stitching," *Med. Image Process.*, vol. 14, pp. 674–686, 2010.

[53] W. Xu and J. Mulligan, "Performance evaluation of color correction approaches for automatic multi-view image and video stitching," in *Proc. Comput. Vis. Pattern Recog.*, 2010, pp. 263–270.

[54] Y. Liu, G. Chen, N. Max, C. Hofsetz, and P. McGuinness, "Undersampled light field rendering by a plane sweep," *Comput. Graph. Forum*, vol. 25, no. 2, pp. 225–236, Jun. 2006.

[55] R. Yang, M. Pollefeys, H. Yang, and G. Welch, "A unified approach to real-time multi-resolution multi-baseline 2D view synthesis and 3D depth estimation using commodity graphics hardware," *Int. J. Image Graph.*, vol. 4, no. 4, pp. 627–651, Oct. 2004.

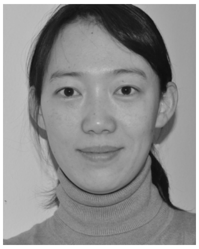
[56] W. Yang, J. Zheng, J. Cai, S. Rahardja, and C. W. Chen, "Natural and seamless image composition with color control," *IEEE Trans. Image Process.*, vol. 18, no. 11, pp. 2584–2592, Nov. 2009.

[57] J. M. Zacks and B. Tversky, "Event structure in perception and conception," *Psychol. Bull.*, vol. 127, no. 1, pp. 3–21, Jan. 2001.

[58] Y. Zhang and C. Kambhamettu, "Stereo matching with segmentation-based cooperation," in *Proc. 7th ECCV—Part II*, 2002, pp. 556–571.

[59] Z. Y. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 11, pp. 1330–1334, Nov. 2000.

[60] A. Zomet, D. Feldman, S. Peleg, and D. Weinshall, "Mosaicing new views: The crossed-slits projection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 6, pp. 741–754, Jun. 2003.



Qi Zhi received the B.Eng. degree from Southeast University, Nanjing, China, the M.Eng. degree from the National University of Singapore, Singapore, and the Ph.D. degree from the McGill University, Montreal, QC, Canada.

She is currently an Assistant Professor with the Department of Electrical Engineering, Southeast University. Her research interests include computer vision, image processing, and pattern recognition, recently with a focus on reconfigurable computing, embedded computer vision, and network-based human–computer interaction.

Jeremy R. Cooperstock received the Ph.D. degree from the University of Toronto, Toronto, Canada, in 1996.

He is with the Centre for Intelligent Machines, McGill University, Montreal, QC, Canada. He currently directs McGill University's Shared Reality Laboratory, which focuses on computer mediation to facilitate high-fidelity human communication and the synthesis of perceptually engaging, multimodal, and immersive environments. His accomplishments include the world's first Internet streaming demonstrations of Dolby Digital 5.1; uncompressed 12-channel 96 kHz/24 bit; multiple simultaneous streams of uncompressed high-definition video; and a simulation environment that renders graphic, audio, and vibrotactile effects in response to footsteps.

Mr. Cooperstock's work on the ultravideoconferencing system was recognized by an award for the Most Innovative Use of New Technology from the Association for Computing Machinery/IEEE Supercomputing and a Distinction Award from the Audio Engineering Society.