Depth-based Image Mosaicing for Both Static and Dynamic Scenes

Qi Zhi
Centre for Intelligent Machines
McGill University
qizhi@cim.mcgill.ca

Jeremy R. Cooperstock
Centre for Intelligent Machines
McGill University
jer@cim.mcgill.ca

Abstract

Traditional image-based mosaicing deals with the problem of parallax by imposing constraints of a parallax-free camera configuration or requiring a dense sampling of the scene. These solutions are often impractical or fail to address the needs of the application. Instead, taking advantage of depth cues and a criterion of smooth transitions, we achieve significantly improved mosaicing results for static scenes, coping effectively with non-trivial parallax in the input. Furthermore, by incorporating a criterion of consistent motion perception, we demonstrate progress on mosaicing of dynamic scenes without introducing artifacts. Although further additions are required to cope with unconstrained object motion, our algorithm can synthesize perceptually convincing dynamic mosaics, conveying the same appearance of object motion as seen in the original sequences.

1. Introduction

Traditional image mosaicing techniques operate by first aligning the inputs and then warping and stitching them together. However, in the presence of considerable disparity variance or dynamic objects in the scene, image registration is frustrated by parallax effects and object motion. This results in misalignments, which lead to artifacts in both static and dynamic mosaics.

In order to cope with the parallax problem, most algorithms impose constraints of either a parallax-free camera configuration [3][5][12], or a dense sampling of the scene, such as that provided by manifold mosaics [9][14]. However, these requirements may be impractical or prohibitively expensive to satisfy.

The application of mosaicing to dynamic video sequences was first proposed by Irani [7] and Sawhney [11]. Both approaches were based on the assumption of a parallax-free input video, provided by a single rotating camera, which cannot capture dynamic events continuously in both spatial and temporal dimensions. Rav-Acha et al. [1] described the production of non-chronological mosaic video. Although effective for its purpose, this unfortunately does not preserve the perception of chronologically continuous motion for objects in the scene.

To overcome the aforementioned problems and limitations, we introduce a novel image mosaicing algorithm, which considers $2D$ image mosaicing as a depth-based view synthesis problem. When integrated with foreground-background segmentation and motion perception analysis, our algorithm can generate reasonable dynamic mosaics when given inputs exhibiting non-trivial parallax.

The remainder of this paper is organized as follows. Section 2 describes our depth-based image mosaic approach, on which the dynamic mosaic algorithm, described in Section 3, relies. Section 4 provides a comparison of experimental results with those of Autostitch, and Section 5 concludes with a discussion of desirable improvements.

2. Depth-Based Image Mosaicing for a Static Scene

In contrast with traditional mosaicing techniques, we synthesize the panorama as if seen through a virtual camera with a wider field-of-view (FOV) than the inputs. This assumes availability of a depth map of the entire scene, a requirement we discuss in further detail below. Provided that the depth estimates are reasonable, we may build a panorama free of parallax-related artifacts. While traditional view synthesis techniques only consider content that is viewed by multiple input cameras, i.e., is contained in overlapping regions of the input where stereo information is available, our approach employs a depth propagation procedure to include the contents of non-overlapping regions, visible only to a single camera, as well.

2.1 Synthesis of overlapping regions

Mosaicing in the overlapping regions, R_o, is performed using the plane sweep algorithm [6]. Given a selected position and orientation of the virtual camera, the inputs are then projected onto parallel planes located at different depths to generate a set of intermediate images. Each intermediate image contains RGB color channels and an associated matching score channel that denotes the similarity between the projections from different input images.

Let P be the set of pixels in the output mosaic, and L be the set of depth levels. We apply the *graph cut* [2] algorithm to estimate color and depth information for every pixel in the overlapping regions of the mosaic image by minimizing the following equation:

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

where labelling $f : P \rightarrow L$ assigns each pixel of the synthesized frame a discrete depth level, E_{data} represents the color consistency between various sources and $E_{smoothness}$ indicates the smoothness of the transition between estimated depths of neighboring pixels in the virtual image.

2.2 Synthesis of non-overlapping regions

Because of the lack of stereo correspondence information, mosaic pixels of non-overlapping regions, R_{non}, must be calculated by a different method from that applied to R_o. We observe that depth discontinuities rarely occur in regions of uniform texture but typically coincide with color segment boundaries. Taking advantage of this fact, we propagate the (reliable) depth information of color segments from R_o to adjacent color segments in R_{non}, provided that this results in the appearance of a smooth transition between them.

Synthesis of the mosaic in R_{non} is actually a procedure that maps color segments of the output mosaic to their correspondences in intermediate images, $\{I_{d_i}\}_{i=1}^{N}$, built during synthesis of the mosaic in R_o. 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 $\{d_1, \cdots, d_N\}$. Based on the assumption of uniform depth, a greedy algorithm is used to calculate the best labelling $\rho : S \rightarrow L$ that minimizes the energy function:

$$E(\rho) = E_{smoothness} + E_{occlusion} \tag{2}$$

The first term, $E_{smoothness}$, evaluates the overall connection cost between neighboring color segments as follows:

$$E_{smoothness} = \sum_{i=1}^{M} \sum_{(p,q)\in\Psi} C_i(p,q) \tag{3}$$

where Ψ represents the border areas between the color segment s_i and its neighbors, and $\sum_{(p,q)\in\Psi} C_i(p,q)$, with respect to one color segment s_i ($s_i \in S$), is the total smooth connection cost [8] of all pairs of neighboring pixels, (p,q), within Ψ. The second term, $E_{occlusion}$, accounts for occlusion by applying a constant penalty value, λ_{occ}, for each occluded s_i.

Starting with the subset of segments S_{M1} in R_{non} and immediately adjacent to R_o, all depth candidates are tested for each $s_i \in S_{M1}$, noting the best candidate. The update of depth values is reserved until the end of the iteration, and is applied only if the overall energy of the entire group is improved over the current depth values obtained using $\rho_{M1} = \{\rho(s_i), s_i \in S_{M1}\}$. This process is performed until either the change of total connection cost between iterations is insignificant or the number of iterations exceeds a threshold. Then, the process is applied to the immediate neighbors of S_{M1} for which depth estimates have not yet been computed. This continues until no unprocessed color segments remain. Once depth estimates are obtained, the mosaic in R_{non} is rendered by copying the corresponding color segments from intermediate images into the mosaicing image plane.

3. Depth-based Dynamic Mosaics

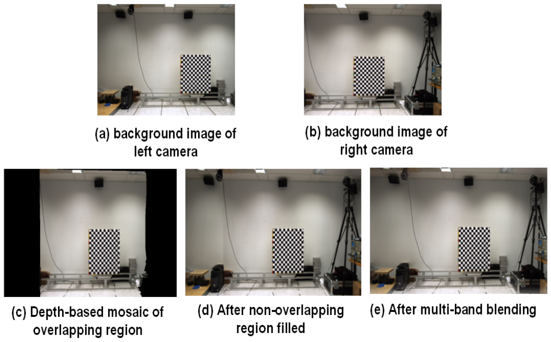
We now turn to the problem of generating a perceptually correct mosaic that includes moving objects in the scene. Unlike earlier dynamic mosaicing approaches [7][11][1], which employ a single rotating video camera to acquire the scene content, we use a multiple camera configuration, with a large baseline. The resulting parallax effects pose a significant challenge to the mosaicing task.

Our approach first performs a segmentation of foreground and background layers using a Mixture-of-Gaussians (MoGs) model [13], which offers robustness to potentially complex illumination conditions, such as non-uniform lighting and dynamic shadows. It then projects these layers separately onto the mosaicing plane, according to their respective depth estimates, to render the final result.

3.1 Foreground-background segmentation

Suppose the intensity of each pixel in the frames satisfies the distribution of an MoG model, which contains $K = 3$ Gaussian elements in our present implementation. For a given frame, the probability that each pixel belongs to the background is calculated according to the trained models. If this probability exceeds some threshold, the pixel is considered as an element of the static background, and otherwise, as a dynamic foreground object.

In addition to the binary foreground-background segmentation result, the *background image*, a weighted sum of the means of each Gaussian element from the MoG, is also constructed. This image retains only the


(a) background image of left camera

(b) background image of right camera

(c) Depth-based mosaic of overlapping region

(d) After non-overlapping region filled

(e) After multi-band blending

Figure 1. Construction of the mosaic background

static portions of the scene while dynamic foreground objects are removed. Two such background images are pictured in Figure 1a-b, along with the depth-based mosaic (DBM) based on these, seen in Figure 1d. A multi-band blending strategy [4] is applied to balance the color differences between camera responses, with the result shown in Figure 1e.

3.2 Rendering of dynamic mosaic

The foreground mosaic in R_o as observed by the virtual mosaicing camera is synthesized in the same manner as that used in the static case, as described in Section 2.1. However, for R_{non}, where no stereo information is available to estimate depth values, a different process is necessary.

People understand motion as a sequence of *instances* [10], which are sensitive to characteristics of the motion trajectories, specifically, velocity and its first derivative, as they appear in the 2D video sequences. Furthermore, $2D$ trajectories with the same sequence of instances are perceived as corresponding to the same motion, regardless of the poses of the corresponding video cameras.

As in the case of static mosaicing, for which we require the appearance of a smooth transition between contents from different sources, here, for dynamic mosaicing, we require the consistent perception of motion if objects move across the FOVs between different cameras. So that, in R_{non}, if the output mosaic video observed by the virtual mosaicing camera, preserves the same velocities as the input video, thus maintains the same sequence of instances, this output mosaic video ensures an identical perception of motion, as well as spatiotemporal motion consistency as presented in input video.

These principles motivate our approach to the construction of mosaic video. Construction of the t'th frame of foreground contents in R_{non} is a procedure that projects the foreground layer onto the mosaicing image plane according to its proper depth estimate, $d(t)$. This should be done in such a manner that the motion trajectory in the output mosaic preserves the same velocity as that in the input. Finally, a simple merging of the background mosaic, as in the example of Figure 1e, with the foreground mosaic video, results in the final dynamic mosaic.

4. Experimental Results

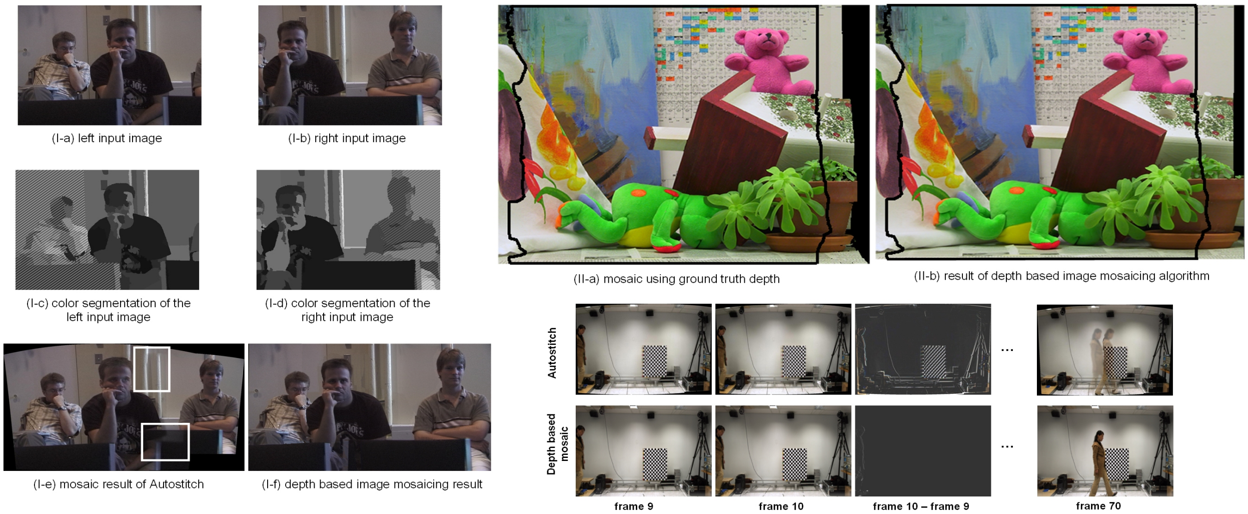
We are unaware of any comparable mosaicing technique that has proven capable of addressing non-trivial parallax effects from sparse sampling in either static or dynamic scenes. As a well-known representative of conventional mosaicing techniques, we use Autostitch as a comparison against which to evaluate the quality of our algorithms.

To obtain an alignment ensuring the best matching performance in R_o between sources, Autostitch must deform the input images, compressing objects closer to the cameras and expanding distant objects to equalize for their respective disparities. The effects of this process can be seen in Figure 2(I-e). In contrast, our depth-based image mosaicing results, shown in Figure 2(I-f), are free of such deformations. Furthermore, our mosaics do not suffer from the unpleasant ghosting effects seen in the Autostitch result, as indicated by the highlighted bounding boxes in Figure 2(I-e), stemming from misalignments due to parallax effects. Note that these artifacts are evident, even after applying deformations to compensate for disparity variance.

For further comparison, a reference DBM result generated using the ground truth depth values from the teddy data set is shown in Figure 2(II-a). The overlapping regions, i.e., within the black boundaries of Figure 2(II-b), exhibit reasonable coherence with the reference mosaic, although appearance differences due to depth estimate variance are observed in the non-overlapping regions, in particular toward the right of the mosaic result.

It bears comment that unlike view synthesis algorithms, our DBM approach does not attempt to determine the real depth in non-overlapping regions. Indeed, with the naive assumption of uniform depth of each color segment in R_{non}, the DBM method generates depth estimates that usually do not conform to the ground truth topology of most scenes. Nevertheless, the smooth appearance connection criterion guarantees a resemblance between local regions in the mosaic results with those of the inputs. As such, the mosaicing outputs based on these depth estimates still appear reasonable and perceptually acceptable.

For video inputs containing moving foreground objects, Autostitch, like other traditional image-based mo-


(I-a) left input image

(I-b) right input image

(I-c) color segmentation of the left input image

(I-d) color segmentation of the right input image

(I-e) mosaic result of Autostitch

(I-f) depth based image mosaicing result

(II-a) mosaic using ground truth depth

(II-b) result of depth based image mosaicing algorithm

Figure 2. Comparison of our depth-based mosaicing algorithm to Autostitch.

saicing algorithms, generates results in which consecutive frames may exhibit jitter, as seen in the difference frame in Figure 2. Furthermore, the ghost errors in frame$_{70}$, resulting from parallax, are a significant issue even for a single mosaic image. By comparison, our depth-based dynamic mosaicing approach produces results that are free of parallax-related artifacts.

5. Conclusion and Future Work

Traditional algorithms are prevented from generating perceptually acceptable panoramas under fairly common conditions. In response, we introduced techniques that treat image mosaicing as a view synthesis problem that must exploit depth information. We demonstrated the use of a smooth motion perception criterion, which guarantees not only the appearance of correct motion but also motion consistency in both spatial and temporal dimensions. Our algorithms are applicable to both static and dynamic scenes.

The results presented here are, of course, only a start. Considerable work remains, in particular to cope with arbitrary movement of multiple objects in the scene. Furthermore, dynamic mosaicing at video rates, requires taking advantage of the parallel computation abilities of a GPU, or exploiting the efficient depth map generation abilities of pre-calibrated stereo cameras or laser rangefinders.

References

[1] R. Alex, Y. Pritch, D. Lischinski, and S. Peleg. Dynamosaicing: Mosaicing of dynamic scenes. *IEEE Trans. PAMI*, pages 1789–1801, 2007.

[2] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. In *IEEE Trans. PAMI*, pages 1124–1137, 2004.

[3] M. Brown and D. Lowe. Recognising panoramas. In *Int. Conf. Comput. Vision (ICCV)*, pages 1218–1225, 2003.

[4] P. Burt and E. Adelson. A multiresolution spline with application to image mosaics. *Transactions on Graphics*, pages 217–236, 1983.

[5] S. Chen. Quicktime VR: an image-based approach to virtual environment navigation. In *SIGGRAPH*, pages 29–38. ACM, 1995.

[6] R. Collins. A space-sweep approach to true multi-image matching. *IEEE Trans. PAMI*, pages 358–363, 1996.

[7] M. Irani, P. Anandan, and S. Hus. Mosaic based representation of video sequence and their applications. In *Int. Conf. Comput. Vision (ICCV)*, pages 605–611, 1995.

[8] V. Kwatra, A. Schodl, I. Essa, G. Turk, and A. Bobick. Graphcut textures: Image and video synthesis using graph cuts. *ACM Transactions on Graphics, SIGGRAPH 2003*, 22:277–286, July 2003.

[9] S. Peleg and J. Herman. Panoramic mosaics by manifold projection. In *CVPR*, pages 338–343, 1997.

[10] C. Rao, A. Yilmaz, and M. Shah. View-invariant representation and recognition of actions. *Int. J. Comput. Vision*, 50(2):203–226, 2002.

[11] H. Sawhney and S. Ayer. Compact representations of videos through dominant and multiple motion estimation. *Trans. PAMI*, pages 814–830, 1996.

[12] H. Shum and R. Szeliski. Construction and refinement of panoramic mosaics with global and local alignment. In *Int. Conf. Comput. Vision (ICCV)*, pages 953–960, 1998.

[13] C. Stauffer and W. Grimson. Adaptive background mixture models for real-time tracking. In *CVPR '99*, pages 246–252, 1999.

[14] A. Zomet, D. Feldman, S. Peleg, and D. Weinshall. Mosaicing new views: The crossed-slits projection. *IEEE Trans. PAMI*, pages 741–754, 2003.