Scene Collages and Flexible Camera Arrays

Yoshikuni Nomura,[1] Li Zhang,[2] and Shree K. Nayar[2]

[1]Sony Corporation, Yoshikuni.Nomura@jp.sony.com
[2]Columbia University, {lizhang,nayar}@cs.columbia.edu

Abstract

This paper presents an automatic method for creating a collage from a collection of photos of a scene taken from different viewpoints. The collage is constructed by aligning the images (in terms of their positions, rotations and scales) using a least-squares formulation. We have developed a graph-based optimization algorithm for layering the images so as to minimize the fragmentation of the collage. A collage can be displayed with opaque layers, with transparent layers, or with blended image boundaries. A scene collage can be viewed as a piece-wise perspective representation of a scene with visible seams. This representation has not only aesthetic value but also conveys scene structure and camera motion in an intuitive way. To capture live-action collages of dynamic scenes we have developed camera arrays that can be physically flexed by the user to continuously vary the composition of the scene. The design of our camera arrays enables a user to reconfigure them in terms of the spatial arrangement of the cameras in a matter of minutes. We show several still and dynamic examples that demonstrate that scene collages provide a new and interesting way to experience scenes.

Categories and Subject Descriptors (according to ACM CCS): I.4.1 [Image Processing and Computer Vision]: Digitization and Image Capture; I.4.8 [Image Processing and Computer Vision]: Scene Analysis

1. Introduction

With the advent of digital cameras, taking many pictures of a scene from different viewpoints has turned into a common practice. The abundance of such image sets has motivated researchers to develop algorithms to create photomosaics with wide fields of view, that can be used with an interactive viewer to more richly experience the scene (for examples, see [Che95, IAH95, SS98, SKG*98, BL03, AAC*06]). Although the previous works approach this problem in different ways, they share the same goal: to create a single *seamless* image of the scene.

The goal of "seamlessness" in creating a single representation from a collection of images raises two key problems. First, it reduces the operating range of the resulting representation. All mosaicing methods require the scene to be distant or consisting of a dominant plane. If these conditions are not met, the computed mosaic includes parallax artifacts such as blurring. Second, when the set of input images represent a wide field of view, the mosaic appears distorted as it attempts to represent a large portion of a sphere onto a plane (see Figure 1(b)). As a result, the mosaic is not easy to interpret and a viewer (such as QuickTime VR [Che95]) must

be used to select narrower perspective views from it. This second problem was recently discussed in [ZMPP05].

We believe that seamlessness is not a necessary criterion in creating a visual representation of an image collection for human consumption. In fact, images with seams have their own aesthetic value. This is exemplified by the photographic collages created by the artist David Hockney. See http://www.ibiblio.org/wm/paint/auth/hockney/. His *Pearblossom Highway* and *Place Furstenberg* collages include many patches selected from photos taken from different viewpoints. While the boundaries of the patches are clearly visible, these collages give us a more comprehensive view of the scene without the use of a software viewer. In Hockney's opinion, this is because such collages are more amenable to human visual perception than seamless but smoothly-distorted photos taken using wide-angle lenses. Recently, collages of this type have piqued the interest of photographers. For example, the photo-sharing website, http://www.flickr.com, has hundreds of such collages created by members in the group named "Panography." Even though the images used in these collages are taken in an uncontrolled fashion and the scenes are arbitrary, the final col-
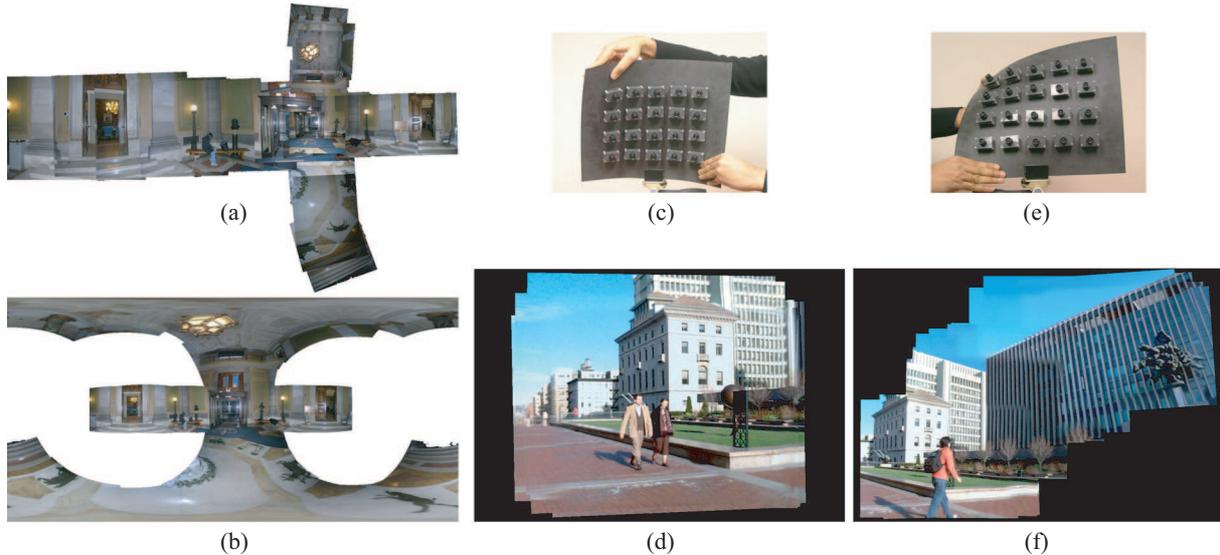
Figure 1: *Illustration of Scene Collage and Flexible Camera Array. (a) A scene collage computed from 33 images of a scene captured from similar viewpoints. (b) A panorama obtained by applying image stitching to the same set of images. Due to the wide field of view covered by the input images, the panorama is highly distorted. We have developed flexible 1D and 2D (shown in (c,e)) camera arrays that can be used to continuously vary the composition of a scene and create a dynamic (video) collage, like the one shown in (d,f).*

lages are impressive to look at. Currently, all of these collages are being manually crafted, for example, using software available at http://www.photojojo.com/content/tutorials/panographies/.

In this paper, we present an automatic approach to compute a Hockney-style collage from a set of input images, which we call a *scene collage*. A lay-outing algorithm uses matched features to align the input images. Then, a layer ordering algorithm automatically orders the input images. Finally, the collage can be displayed with opaque layers, transparent layers, or blended boundaries. When looking at such a collage, a person can comfortably perceive the overall structure of the scene as well as imagine the camera's motion during the capture of the images (see Figure 1(a)). For many scenes, the end result is a richer experience than what a seamless mosaic provides. Specifically, this paper makes the following contributions.

Automated Creation of a Scene Collage: We present a simple method that automatically creates scene collages from a set of images. Our method has two steps. First, the rotations, translations, and scales of the input images are found from matched SIFT features [Low04] using least squares. Then, a graph-based optimization is used to determine the layering of the images so as to minimize the fragmentation of the collage. While SIFT feature matching has been used for panorama stitching [BL03], we use it for the scene collage, a representation that has not been created or used in vision or graphics. We demonstrate the three key advantages of our collages over photomosaics: they convey scene structure and camera motion in a more intuitive way; they are more tolerant to scene parallax; last, but not least, they can be used

to organize photos of a scene and enable photo-browsing at various levels-of-detail.

Flexible and Reconfigurable Camera Arrays: To create video collages of dynamic scenes, we present a way to design flexible camera arrays. These arrays can be used to simultaneously capture videos of a scene from different viewpoints. Our design consists of a plastic frame onto which a set of cameras can be easily attached, very much like Lego^{TM}building blocks. The spatial layout of the cameras can be reconfigured in a matter of minutes to achieve a variety of configurations, such as "L" and "T" shaped ones. The plastic frame can be physically flexed to vary the shape of the array (see Figure 1(c,e)). This gives a photographer significant creative control – the composition of the scene can be smoothly varied as the scene changes.

Dynamic Scene Collage: Using the videos captured by a flexible camera array, we compute dynamic collages whose layouts change smoothly with the deformations applied to the array (see Figure 1(d,f)). Dynamic collages represent a new visual medium. Unlike mosaics computed from a single video stream, in a dynamic collage, multiple moving objects are captured from multiple and changing viewpoints.

We have created still and dynamic collages for a wide variety of scenes. These examples illustrate that scene collages can serve as an attractive and effective medium for conveying scene structure.

2. Related Work

In this section, we review methods that create a single image representation from a set of acquired photos as well as camera arrays that have been used for producing such representations.

2.1. Mosaics and Collages

Many methods have been developed for generating high quality mosaics from photos or videos, e.g., [Che95, IAH95, SS98, SKG*98, BL03]. All these methods seek to compute a single seamless mosaic, and therefore require either the scene to be planar or distant, or the camera viewpoints to be closely located. For cases when these requirements are not adequately met, local warping [SS98] and plane sweeping [KSU04] have been proposed to reduce the parallax artifacts. However, these methods are computationally expensive and can sometimes generate blurry results. To address these problems, Agarwala et al. [AAC*06] use graph cuts [BVZ01] to generate piece-wise perspective mosaics. (The graph cuts were used earlier by Kwatra et al. [KSE*03] for seaming images in texture synthesis.) This method still assumes that the scene consists of a dominant plane. However, the method avoids the distortions seen in strip panoramas, e.g., [Zhe03]. To avoid the distortions inherent to panoramas, Zelnik-Manor et al. [ZMPP05] manually segment the scene into foreground and background layers and generate a mosaic with different perspectives for each layer.

Our work is motivated by the recent popularity on http://www.flickr.com of photo-collages of the type created by David Hockney. These collages are all manually created and the goal of our work is to automate this process. At the expense of having seams, our collages are free of local distortions and can be produced for scenes with strong parallax. The weak alignment between the patches of a collage convey a stronger impression of scene structure and camera motion. This observation was also made in [GCSS06] and was used to build a storyboard from a video clip. To this end, our work is related to the multi-perspective panorama [WFH*97], which produces the illusion of 3D motion when viewed through a small moving window.

Recently, several interesting methods have been proposed to create collages, e.g., [RKKB05, RBHB06, WQS*06, DE05]. Each method produces a different type of collage, but in all cases the collage is made from images taken in different scenes. There also exist online services for creating such collages. (See http://www.procollage.com.) In contrast, we are interested in creating a collage from images of the same scene.

2.2. Camera Arrays

Many camera arrays, e.g., [KRN97, WJV*05, JMA06, YEBM02], have been proposed in graphics and vision research to capture images simultaneously from multiple viewpoints. In amateur photography, Lomographic cameras (http://www.lomography.com) with multiple (4~9) fixed lens have been developed. While all these arrays, except for the multi-lens Lomographic cameras, can be rearranged for different applications, they cannot be reconfigured as quickly as our arrays. The only exception is the array built by Zhang and Chen [ZC04], in which each camera is driven by a servo-motor. The positions of the cameras can be controlled to change the light field captured by the array. However, this system does not have the flexibility of ours. Our array can be physically flexed by a photographer to compose a scene in unconventional ways, and the composition can be varied as the scene changes.

3. The Scene Collage

In this section, we present our method for creating a scene collage from a set of photos of an arbitrary scene taken from different viewpoints. Our method has three stages: collage layout, layer ordering, and layer composition.

3.1. Collage Layout

Given a set of input images, $\mathcal{I} = \{I_1, I_2, \cdots, I_N\}$, we associate with each image a rotation angle θ, a translation vector $[u, v]$, and a scale factor s. The rotation and translation are used to approximate camera motion and the scale factor is used to model lens zoom. These four parameters can be represented by a matrix of similarity transform

$$\mathbf{G} = \begin{bmatrix} a & -b & u \\ b & a & v \\ 0 & 0 & 1 \end{bmatrix}, \qquad (1)$$

where $a = s\cos\theta$, $b = s\sin\theta$, and $s = \sqrt{a^2 + b^2}$. We seek to compute an optimal similarity transform for each input image to determine the layout of the collage.

We estimate the similarity transforms using a feature-based approach. Specifically, we extract SIFT features [Low04] in each input image, and then match the features in each pair of images based on the feature descriptors. The matched features often contain outliers, and we prune the outliers by using RANSAC with the Homography model. To tolerate parallax, we use a loose inlier threshold of 11 pixels. A more principled way of handling parallax would be using the method in [TFZ99], which automatically switches between Homography and Fundamental matrices using a statistical model selection test. After running RANSAC, we have a set of matched features between each pair of the input images.

Given the matched feature pairs, we compute the similarity transforms by minimizing the sum of the squared distances between the locations of corresponding features in the coordinate system of the collage. In short, we minimize

$$E_m(\{a_i, b_i, u_i, v_i\}) = \sum_{i=1}^{N} \sum_{j=i+1}^{N} \sum_{k \in F(i,j)} \|\mathbf{G}_i \mathbf{x}_{ik} - \mathbf{G}_j \mathbf{x}_{jk}\|^2, \qquad (2)$$

where $\mathbf{G}_i$ has the parameters (a_i, b_i, u_i, v_i) for image i, $F(i, j)$ is the set of features matched between images i and j, and $\mathbf{x}_{ik}$ and $\mathbf{x}_{jk}$ are the locations of the k-th feature in images i and j, respectively. To obtain a unique solution, we select one image as the reference image with an identity $\mathbf{G}$ matrix. The minimization of Eq. (2) is a linear, least-squares problem that can be solved efficiently [GV96].

The similarity transform is a special case of Homography. From a geometric point of view, we can only use the
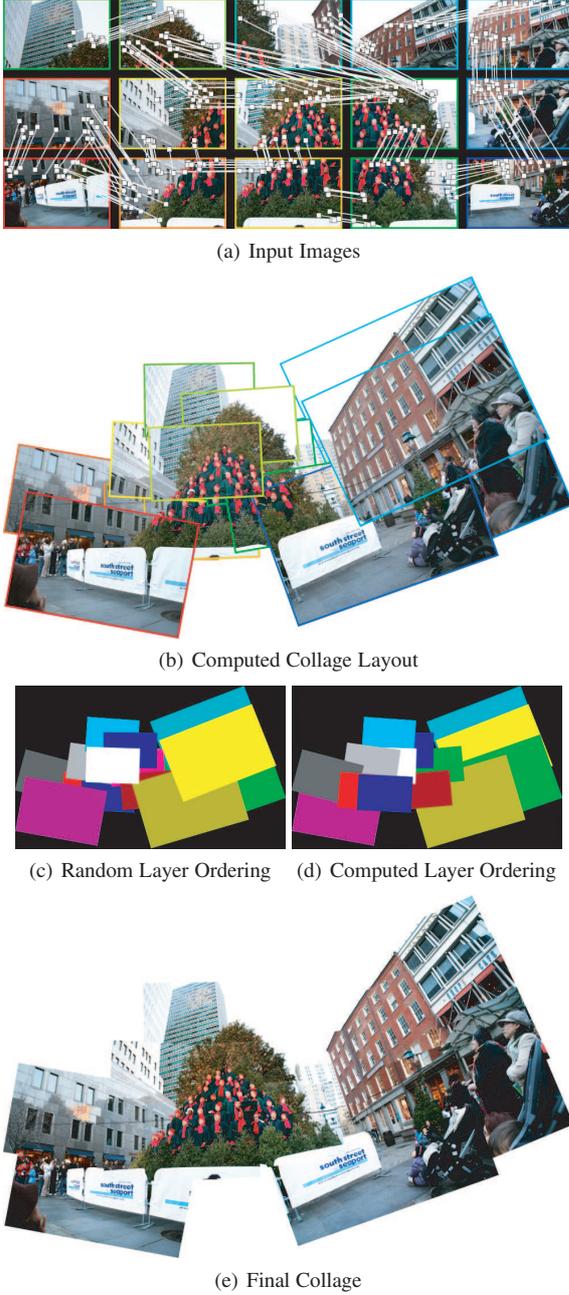
(a) Input Images

(b) Computed Collage Layout

(c) Random Layer Ordering (d) Computed Layer Ordering

(e) Final Collage

Figure 2: *Illustration of collage generation. (a) A set of 15 input images. The white lines show a few of the matched features between pairs of images. (b) The collage layout found by minimizing Eq. (2). (c) A random layer ordering with $E_l = 326.3$ in Eq. (3). Some of the visible regions are very small in this case. (d) The final (optimized) layer ordering with $E_l = 0.4635$. Note that this ordering is less fragmented. (e) The final collage computed using the optimized layer ordering.*

similarity transform to represent image motion when the optical axis of the camera is perpendicular to a planar scene and the camera motion is parallel to this plane. The similarity transform gives an approximate alignment between the images but does not change the scene appearance in each image. Laying out all the images on a plane using this transform avoids the severe distortions seen in spherical mosaics near the two poles. This is the main reason we chose to use similarity transforms for computing the layout.

For the set of photos shown in Figure 2(a), our method computes the collage layout shown in Figure 2(b). Although the image boundaries are visible in the collage, we get a good feel for the scene as the local structures of the individual images are preserved.
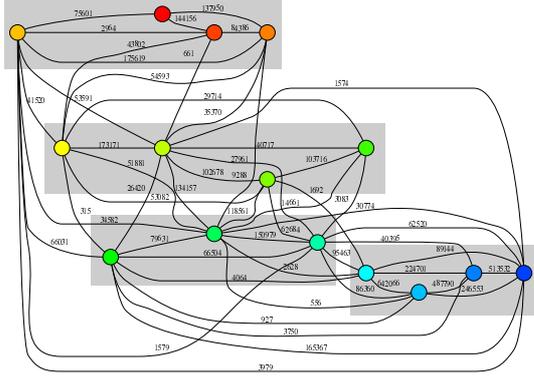
3.2. Layer Ordering

For a given layout, different layer orderings of the input images will result in different collage appearances, since the contents of the images are not perfectly aligned. We now present a method that automatically orders the images such that the collage appears least fragmented.

The layering problem can be formulated as an energy minimization that penalizes the creation of small visible patches in the final collage. Let $\{l_i\}$ be a layer ordering that assigns layer l_i to image i. We find the ordering that minimizes the following objective function:

$$E_l(\{l_i\}) = \sum_{1 \leq m \leq M} \frac{1}{\sum_{\mathbf{x} \in V(m)} w(\mathbf{x})}, \qquad (3)$$

where $V(m)$ is the m'th visible segment for ordering $\{l_i\}$, and $w(\mathbf{x})$ is a weight associated with each pixel. If we set $w(\mathbf{x}) = 1$, the denominator in Eq. (3) is the area of the visible segment $V(m)$. Since layer ordering does not change the *total* collage area, Eq. (3) encourages an ordering that results in visible regions that have similar areas[†]. In our implementation, we also encourage image regions with high frequency information to be visible. To this end, we set $w(\mathbf{x})$ to be the local intensity variance within a window of 3×3 pixels around $\mathbf{x}$. Alternative choices for w could be based on image saliency [IKN98] or objects of interest [VJ01], which we have not used in our current implementation. Next, we describe an approximate algorithm that efficiently minimizes Eq. (3).

[†] As a simple example, suppose we have two layers and their total visible area is 1. There are only two possible layer orderings in this case. Let one ordering have two visible regions with areas p and $1 - p$, and the other with q and $1 - q$. It is easy to verify that $\frac{1}{p} + \frac{1}{1-p} > \frac{1}{q} + \frac{1}{1-q}$ if $p < q < 0.5$. This inequality suggests that our objective function in Eq. (3) favors the ordering for which the two regions have similar areas. In general, if $\sum_i p_i = 1$, $\sum_i \frac{1}{p_i}$ attains its minimum when all the p_i are equal.

(a) The Layering Graph
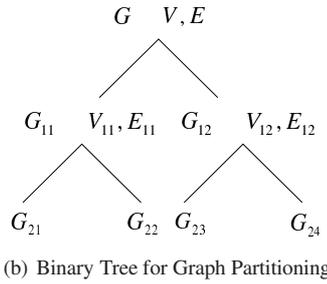
(b) Binary Tree for Graph Partitioning

Figure 3: *Illustration of the layer ordering algorithm. (a) A graph representation of the input images in Figure 2(a). Each node represents an image. The vertex color here is the color of the frame of the corresponding image in Figure 2(b). The weight of an edge between two images is the number of overlapping pixels. We recursively partition the graph into two subgraphs, till each subgraph has no more than 7 nodes. The final subgraphs are shown by the grey rectangles. (b) The partitioning process can be represented by a binary tree, where each leaf node represents a subset of the images. Please see text for details.*

3.2.1. Graph-Based Optimization

When the number N of input images is large, it is intractable to evaluate all $N!$ possible layer orderings to find the optimal one. Instead, we take a *divide-and-conquer* approach. Specifically, we first divide the whole image set into many small subsets and compute the optimal layer ordering for each subset while ignoring the interactions between the subsets. Then, we fix the relative ordering within each subset and find the optimal ordering between the subsets. We now describe the details of this algorithm.

Image Set Decomposition: We wish to divide the image set into subsets between which the interactions are small. For this, we build an undirected graph in which each node is an input image and each edge has a weight that equals the number of overlapping pixels between two images. Figure 3(a) shows an example of such a graph. Dividing the image set into subsets is equivalent to k-way partitioning the graph such that the total weight of edges that connect the subgraphs

is minimum. This graph partitioning is an NP-hard problem. As an approximate solution, we use the METIS package [KK98] to recursively split the graph into two subgraphs until all subgraphs contain no more than 7 nodes. This procedure can be represented using a binary tree, as shown in Figure 3(b).

Layer Order Generation: Given the decomposed image subsets, we exhaustively search for the best layer ordering for each subset, while ignoring the interactions between the subsets. Then, we search for the ordering of the subsets by fixing the ordering within the subsets. Instead of doing an exhaustive search[‡], we use the binary tree structure obtained during the graph partitioning state to make the search efficient. Starting from a pair of leaf nodes, we compare the two possible orderings between them and choose the one that gives a smaller value for the layer ordering objective function. Then, we merge these two nodes to generate a larger image subset, within which the ordering is decided. We recursively collapse the leaf nodes to obtain the final layer ordering of the collage.

While this heuristic search does not necessarily give the globally optimal solution, we find it works quite well in practice. Figure 2(d) shows the result of layer ordering, which does not contain the small fragments of visible regions seen in the random ordering shown in Figure 2(c). Figure 2(e) shows the collage obtained using the optimized ordering. It appears less fragmented than the collage in Figure 2(b), which has the random ordering shown in Figure 2(c).

3.3. Layer Composition

Given the collage layout and layer ordering, we are ready to generate the collage. We first use the procedure in [AAC*06] to compensate for color and brightness differences between the input images due to the use of different exposure settings. We then use α-blending to synthesize the collage.

Specifically, let α_i be the α value associated with image i. We compute the collage I as

$$I = \alpha_1 I_1 + (1 - \alpha_1)(\alpha_2 I_2 + (1 - \alpha_2)(\cdots + (1 - \alpha_{N-1})I_N)). \tag{4}$$

In general, α_i can be different for different images, and can even be spatially varying. We have experimented with the following three schemes for setting the α values.

(A) If $\alpha_i = 1$ for all layers (images), only the first layer will contribute to each pixel in the collage. We call this type of a collage an *opaque collage*. **(B)** If $\alpha_i = 0.5$, all images contribute to the final collage. We call this type of a collage a *transparent collage*. Transparent collages have aesthetic value and a majority of the collages found on http://www.flickr.com are created this way. **(C)** Finally, for each image i, we can set α_i to be 1 in a central region

---

[‡] If we have N images divided into subsets of size L, the number of possible orderings for each subset is $\lceil \frac{N}{L} \rceil!$.
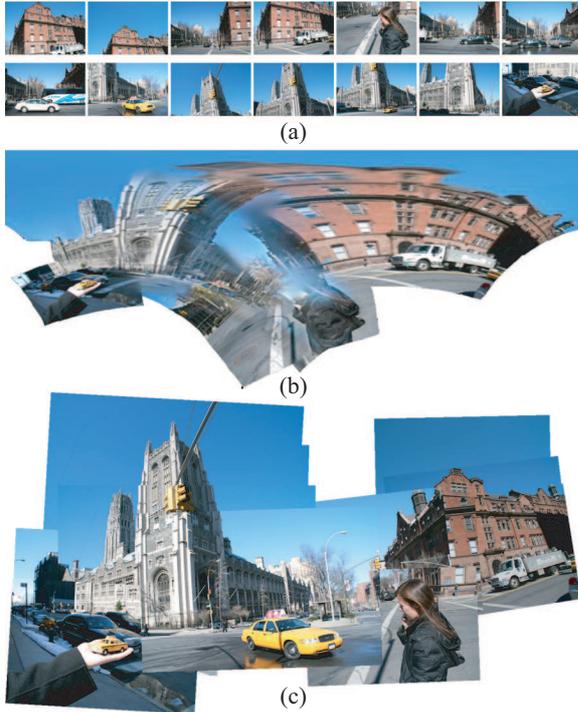
(a)

(b)

(c)

Figure 4: *Comparison between scene collages and seamless panoramas. (a) Samples of 15 images as one crosses a street of about 25 meters wide. (b) A panorama obtained by applying image stitching to the set of images. This panorama is severely blurred in several areas due to significant parallax caused by the camera movement. (c) An opaque collage computed from the same set of images. While the collage has many boundaries, each of its components is perspective and the spatial arrangement of the components conveys the structure of the scene and the viewpoint movement. Compared to seamless panoramas, scene collages have boundaries (seams) but do not have distortions between the boundaries and are more tolerant to parallax effects.*

of the image and taper it to 0 at the boundaries. This setting smoothly blends the boundaries of the input images but keeps the interiors crisp. We call such a collage a *blended-boundary collage*.

3.4. Collage Editing

While our method automates the process of creating a scene collage, the collage is an artistic representation and a user should make the final decision on its appearance. For this, our system also supports user interaction for adjusting the layout, the layer ordering, and the composition options. However, we must emphasize that all our results shown in the paper are automatically generated, except for the layout of Figure 10(i) and the layer ordering of Figure 5(b). In Figure 10(i), we provide the similarity transform between one pair of cameras (the 13'th and 14'th), because their images are largely composed of the white tablecloth and do not have enough features for matching. In Figure 5(b), we manually
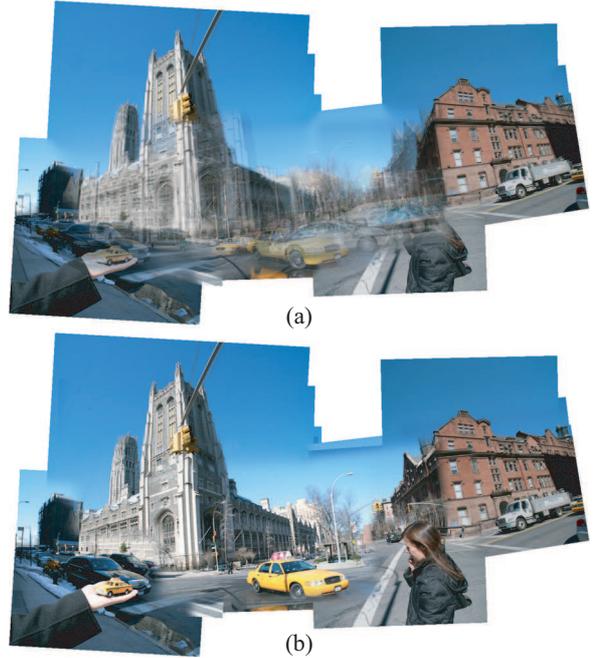

(a)

(b)

Figure 5: *Comparison between two different blending methods for collage synthesis. (a) The weighted average using $I = \frac{\sum \alpha_i I_i}{\sum \alpha_i}$. (b) The α-blending using Eq. (4). Using the same set of α-maps (described in Section 3.3, scheme (C)), the α-blending better keeps the crispness of the top layer than the weighted average.*

choose the layer ordering to avoid the yellow taxi at the center of the topmost layer from being partially occluded by other layers. This is because our layering algorithm does not model the semantic meaning of the input images.

3.5. Results

We first compare our scene collages with the seamless images obtained using a mosaicing algorithm. Figure 1(a) shows a scene collage with opaque layers computed from a collection of 33 images taken from similar viewpoints in a church. While the collage has many boundaries, each of its components is perspective and the spatial arrangement of the components conveys the structure of the scene and the camera viewpoints used to capture the images. We also generate a spherical panorama of the scene (using the AutoStitch [BL03] mosaicing software), which is shown in Figure 1(b). Due to the wide field of view covered by the acquired images, the stitched panorama is highly distorted. While such an image can be used to explore the scene with a software viewer, it is difficult to perceive the structure of the scene or the camera viewpoints from it.

Figure 4(a) shows a few of 15 photos taken when a person crosses a street of about 25 meters wide. Due to the significant parallax, the panorama created from these photos (also using AutoStitch) is blurry and distorted, shown in (b). Figure 4(c) shows the scene collage using our method. Note that, the image content and scene structure are preserved,
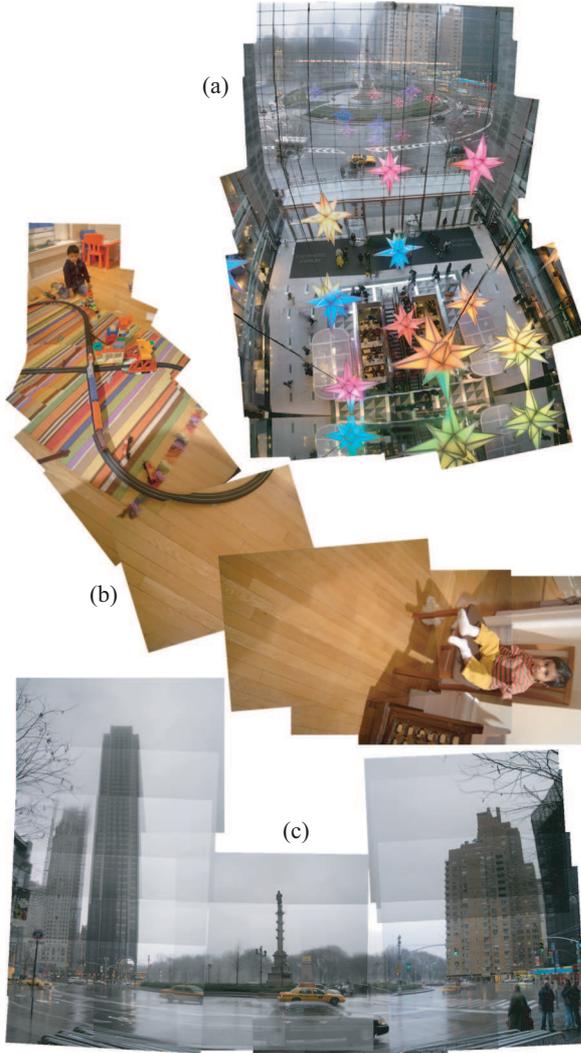
Figure 6: *More examples of scene collages using blended boundaries (a), opaque layers (b), and transparent layers (c). Please see text for details.*
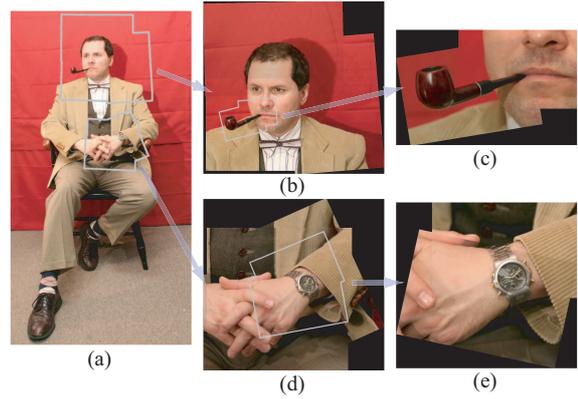

Figure 7: *A nested collage. (a) A photo of a person with boxes that convey the locations of sub-collages that have greater detail. (b,d) Sub-collages of the person's face and hands. (c,e) Sub-collages that include close-ups of the pipe and the watch.*

This is because topmost layers have dominant weights for the α-blending[§].

In Figure 6, we show a few more examples of scene collages. The collage with blended boundaries in Figure 6(a) is made of 35 images taken inside a large atrium. The structure of the atrium is easily perceived as each component of the collage is perspective. Figure 6(b) shows a collage with opaque layers computed from 8 images of a playroom. The motion of the camera is easily perceived in this case. Finally, Figure 6(c) shows a transparent collage of a skyline made of 15 images. All the component images are revealed due to the transparency. **Please see supplementary materials for high resolution versions of all the scene collages in the paper.**

3.6. Browsing Photos with Nested Collages

A scene collage provides a single layout of a set of images. When images of a scene are taken with a very wide range of zoom settings (focal lengths), the close-up images will appear as small regions in the final collage and the details they carry will be lost. To facilitate the browsing of scene images that have different levels of detail, we propose a hierarchical structure that we call a *nested collage*.

A nested collage is created with the same layout method used for scene collages. We find the image with the largest scale factor s_1 estimated by the layout algorithm and all other images whose scale factors are at least half of s_1. We compute a "sub-collage" of these images, which is used as the lowest level (resolution) of the nested collage. Then, from the remaining images, we take the image with the largest scale factor s_2 and those with scale factors that are at least half of s_2. We make a sub-collage of these images and used it

albeit the boundaries of the input images are clearly visible. This example demonstrates that the scene collage is more tolerant to parallax effects than the spherical panorama.

If a user wishes to reduce the boundary effect in the scene collage, she/he can choose to use the spatially-varying α-maps (described in Section 3.3, scheme (C)) and apply the α-blending in Eq. (4) to synthesize the collage. Based on the same set of α-maps, Figure 5 compares the traditional weighted-average blending using $I = \frac{\sum \alpha_i I_i}{\sum \alpha_i}$ and the α-blending using Eq. (4), for collage synthesis. The former is widely used in the mosaicing literature, e.g. [SS98]. Notice that the α-blending better keeps the crispness of the top layer and the weighted-averaging blurs out details heavily.

[§] The weighted-averaging method works well for mosaicing applications because they are often operated on input images that can be more or less aligned; in the case of collage synthesis, the alignment of input images is often very coarse.
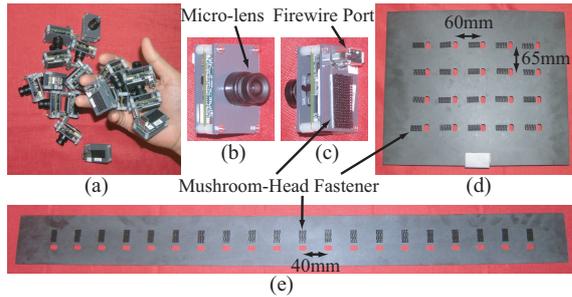
Figure 8: *Flexible Camera Arrays. (a) 20 camera modules, each including a PointGrey Firefly MV camera. (b-c) Front and back views of a camera module. (d-e) Flexible plastic sheets onto which the camera modules are mounted to realize 2D and 1D arrays. Mushroom-head fasteners on the camera modules and the plastic sheets enable a user to create an array with any desired configuration in less than 5 minutes.*

as the second level of the nested collage. This process is applied recursively to obtain a complete nested collage, which is essentially a set of collages with increasing levels of detail.

A nested collage provides a simple way to navigate through images of the same scene taken at different focal lengths (and hence, resolutions). This is illustrated in Figure 7, where a user can start with a full-body image of the man and quickly find a close-up shot of his pipe. Similar ways of browsing image sets are used in the Pseudo-3D Photo Collage system [TAS03] and the Photo Tourism system [SSS06]. In terms of browsing photos, our approach is less sophisticated than these previous systems, as it does not explicitly compute the 3D structure of the scene or morph the images between user-initiated transitions. **Please see the submitted video for a demonstration of collage-based browsing.**

4. Flexible Camera Arrays

To create collages of dynamic scenes, we have developed flexible camera arrays for simultaneously capturing multiple videos from different viewpoints. Using such an array, a user can smoothly vary the composition of a dynamic collage as the scene changes.

4.1. Array Design

We have build 1D and 2D camera arrays by attaching 20 camera modules to flexible plastic sheets (see Figure 8(a,d,e)). Each camera module includes a PointGrey Firefly®MV camera and a micro-lens with 6.0mm focal length¶. The cameras produce 8-bit color images with a resolution of 640x480 pixels at 15fps. The plastic sheets onto which the camera modules are attached are made of Acetal

---

¶ This micro-lens produces more severe distortions than a typical digital camera lens. We have calibrated the distortions for each camera module using the method described in [Zha00]. Since the focal length is fixed, we only need to perform this calibration once.
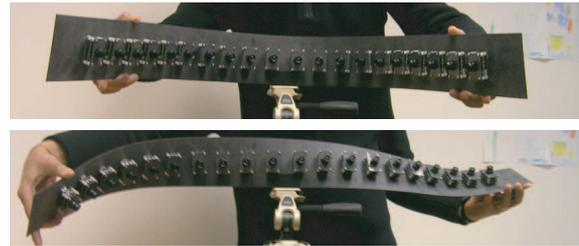

Figure 9: *Two flexed states of the 1D array.*

Copolymer (McMaster-Carr part #8492K511). These sheets can be severely flexed without breaking them.

To quickly mount the camera modules onto the plastic sheets, we have used self-locking, mushroom-head fasteners (McMaster-Carr part #96055K23) to both the back side of the camera modules and the front surfaces of the plastic sheets (see Figure 8(c,d,e)). Once the cameras are mounted on a sheet, they are connected to a host PC via 3 Firewire buses. The PC has a Pentium4 CPU with 4GB RAM, and can store in the RAM approximately 20 seconds of videos from all the 20 cameras. Our design enables a user to configure an entire 1D or 2D array in less than 5 minutes. **(Please see the submitted video for a demonstration.)**

This design also enables a user to flex the sheet to vary the camera viewpoints during scene capture. In the case of a 2D array, one can flex the corners and sides of the array in many different ways. In case of a 1D array, one can make the array convex, concave, or even twisted. Figure 9 and Figure 1(c,e) show the 1D and 2D arrays in various flexed states.

4.2. Dynamic Collage

Given multiple video streams, we can generate a video collage by applying the method in Section 3 to the images captured at each of the time instants. To improve the temporal coherence of the collage layout, we compute the layout parameters for every 10 frames and interpolate the parameters for intermediate frames using Catmull-Rom splines [FvDFH96]. This interpolation also reduces the computations by a factor of 10.‖ Since the cameras are more or less uniformly spaced on the 1D and 2D arrays, the videos of neighboring cameras overlap and these overlap regions vary smoothly. Hence, in the case of dynamic collages, we fix the layer ordering for any given array and maintain this ordering through the entire dynamic collage. This not only reduces computations but also avoids flickering in the computed collage due to sudden changes in the ordering. Specifically, for a 1D array, the ordering is from left to right, and for a 2D array, the ordering is from left to right and top to bottom (as

---

‖ At the time of this submission, SIFT feature detection and matching are the main computation bottleneck in our implementation, which takes about 700 seconds for each collage frame that consists of 20 input images. We are currently accelerating the system by using fast nearest neighbor search [Low04].

Figure 10: *Three examples of dynamic collages captured using a 1D array with 20 cameras. In each example, a shadow effect is added around the image boundaries to highlight the collage structure. In the first example, the array is twisted to capture the church at one end and the bench at the other, and it is held more or less rigid as shown in (a) during the capture of the scene. In the second example, the array starts in the convex state shown in (d) to capture both the people in the scene, and is gradually flexed to be concave as the people approach each other and eventually sit down on the bench. In the third example, the array is hung over a dining table to capture the two people having dinner, as shown in (g). The array is bent such that it captures the man's face on the left and the woman's face on the right. A few of the 20 input frames corresponding to one time instant of the capture are shown in (b), (e), and (h). Please see the submitted video.*
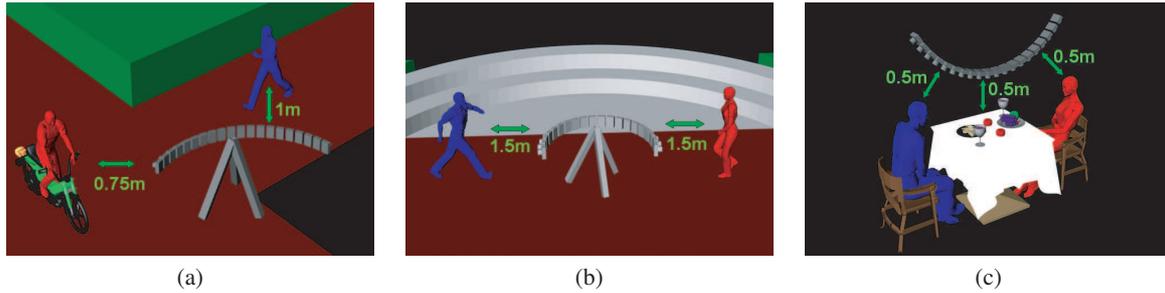
(a) (b) (c)

Figure 11: *Illustration of the configuration of the 1D camera array relative to the subjects for the three collages shown in Figure 10(c,f,i). Notice that due to the proximity of the subjects to the array, appreciable parallax exists for the scenes.*
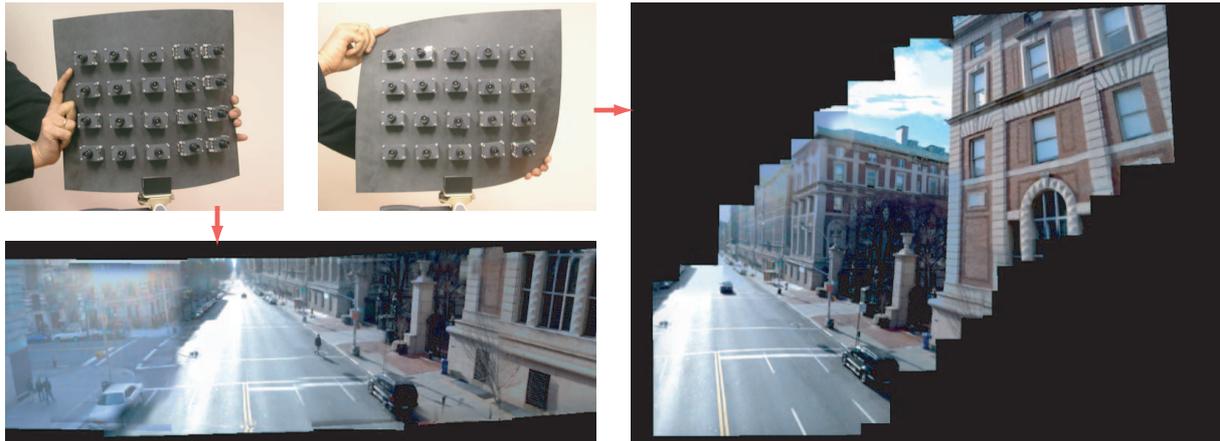

Figure 12: *The dynamic collage of a street scene created using a 2D array of 20 cameras. The array is flexed in various ways during the capture to change the composition of the scene. Please see submitted video.*

in raster scanning). **Please see the submitted video for all the dynamic collages.**

Figure 10(a-c) shows one frame of a dynamic collage created from videos captured using a 1D array with 20 cameras. In this case, the array is held more or less rigid. It is twisted such that it captures a head-on view of the church on the right and an inclined view of the bench on the left. Notice that the two subjects, the walking person and the bicyclist, are simultaneously captured by the array. This type of collage cannot be obtained using previous video mosaicing methods that use a single video camera, e.g. [IAH95]. While a wide-angle camera can also be used to acquire a large field of view (with distortions), it captures the world from a single viewpoint. In contrast, the dynamic collage is a multi-viewpoint video of the scene – the bicyclist is captured by the cameras at one end of the array while the walking person is captured by cameras at the other end. In this example and the following two, shadow effect is added around the image boundaries to highlight the collage structure.

Figure 10(d-f) shows another dynamic collage created by using the 1D array. In this case, the array is continuously flexed such that the collage is framed around the two people who are in motion. The array starts out in an outward-

looking configuration. As the two people get closer, the array is straightened out. Finally, it is flexed inward.

Figure 10(g-i) shows an example of collage taken inside a room. In this case, the 1D array is hung over a dining table shown in (g). The array is bent such that it captures the man's face on the left and the woman's face on the right. The resulting collage is a multi-perspective video that can not be captured using a conventional video camera.

Figure 11 illustrates the configuration of the array relative to the subjects for the three collages shown in Figure 10(c,f,i). Notice that due to the proximity of the subjects to the array, appreciable parallax exists for the scenes.

Figure 12 shows two frames from a dynamic collage of a street scene created using a 2D array with 20 cameras. The array was first flexed to capture a wide horizontal view, as shown in Figure 12(a). Then, the top right corner of the array was bent to capture the buildings on the right, as shown in Figure 12(b). The above examples, and the one in Figure 1(d,f), show that our camera arrays can be used to compose changing scenes in unconventional ways.

5. Discussion

In this paper, we have presented an automatic method for creating a Hockney-style collage from a collection of pho-

Figure 13: *In this example, the seamless mosaic on the top obtained using image stitching may be considered to be more appealing than the scene collage with opaque layers at the bottom.*

Figure 14: *In this example, an opaque collage on the left and a transparent collage on the right are produced from a set of photos taken by moving the camera around the head of a person. Such collages do not reveal all the information embedded in the input images. This limitation of our current approach may be addressed by developing an algorithm that can automatically partition the input images while constructing the collage.*

tos of a scene taken from different viewpoints. We also presented flexible camera arrays that enable us to create dynamic collages with varying scene composition. We now discuss the limitations of our work and suggest directions for future work.

User Study of Collages vs. Mosaics: We have shown several examples that demonstrate that scene collages often convey scene structure in a more intuitive way than spherical mosaics. However, there are many cases where this judgment can be expected to vary between people. For example, Figure 13 shows a collage created by using a set of photos taken in a mall. In this case, the seamless mosaic may be deemed to be better than a collage. This judgment will also depend on the application. For example, a collage provides a more natural way to browse or organize collections of photos. To quantify the comparison between collages and seamless mosaics, we plan to perform a user study using a large number of examples and subjects.

Collages of Inward Views of a Scene: Collages are more tolerant to input images with parallax effects, because they

do not strive for seamlessness. However, collages are not as compelling in the case of a set of inward-looking images. To illustrate this, we captured images of a person's head from viewpoints distributed on a half-circle around the head. Figure 14 shows the opaque and transparent collages computed from this image set. While they may still have aesthetic value, they do not convey all the information embedded in the images. In this case, a cyclograph [SK02] would do a better job, but it would require the capture of a large number of images and the resulting image would be highly distorted. We believe it is possible to create a compelling collage from inward-looking images by using regions from the input images rather than the complete images. This problem is discussed below.

Collage Using Image Patches: In his original work, Hockney used patches from images instead of entire images for creating collages. As a natural extension of our work, we would like to explore optimization methods for decomposing input images into patches and then laying out these patches to create a collage. This can be done by incorporating an automatic image partitioning algorithm into our method. The idea is to take from each input image mainly information that is not available in other images. The main issue here is designing an objective function that would result in visually appealing collages. This is a hard and interesting problem that we plan to explore.

Consumer Flexible Camera Arrays: We have seen many successful applications of camera arrays in vision and graphics research. However, camera arrays are not yet commonplace in consumer photography. We believe our modular/flexible camera array design can be used to develop low-cost, credit-card sized devices that an amateur photographer can quickly reconfigure (much like Lego® blocks) and use to capture new types of images. We see this as the ultimate goal of our work.

References

[AAC*06] AGARWALA A., AGRAWALA M., COHEN M., SALESIN D., SZELISKI R.: Photographing Long Scenes with Multi-viewpoint Panoramas. In *SIGGRAPH Conference Proceedings* (2006), pp. 853–861.

[BL03] BROWN M., LOWE D. G.: Recognising Panoramas. In *Proc. Int. Conf. on Computer Vision* (2003), pp. 1218–1225.

[BVZ01] BOYKOV Y., VEKSLER O., ZABIH R.: Fast Approximate Energy Minimization via Graph Cuts. *IEEE Trans. on Pattern Analysis and Machine Intelligence 23*, 11 (2001), 1222–1239.

[Che95] CHEN S. E.: Quicktime VR: an Image-Based Approach to Virtual Environment Navigation. In *SIGGRAPH Conference Proceedings* (1995), pp. 29–38.

[DE05] DIAKOPOULOS N., ESSA I.: Mediating Photo Collage Authoring. In *Proc. ACM Symp. on User Interface Software and Technology* (2005), pp. 183–186.

[FvDFH96] FOLEY J., VAN DAM A., FEINER S., HUGHES J.: *Computer Graphics: Principles and Practice*, 2 ed. Addison-Wesley, 1996.

[GCSS06] GOLDMAN D. B., CURLESS B., SEITZ S. M., SALESIN D.: Schematic Storyboarding for Video Visualization and Editing. In *SIGGRAPH Conference Proceedings* (2006), pp. 862–871.

[GV96] GOLUB, G. H., VAN LOAN, C. F.: *Matrix Computations*, 3rd ed. Johns Hopkins University Press, Baltimore, 1996.

[IAH95] IRANI M., ANANDAN P., HSU S.: Mosaic Based Representations of Video Sequences and Their Applications. *Proc. Int. Conf. on Computer Vision* (1995), 605–612.

[IKN98] ITTI L., KOCH C., NIEBUR E.: A Model of Saliency-Based Visual Attention for Rapid Scene Analysis. *IEEE Trans. on Pattern Analysis and Machine Intelligence 20*, 11 (1998), 1254–1259.

[JMA06] JOSHI N., MATUSIK W., AVIDAN S.: Natural Video Matting Using Camera Arrays. In *SIGGRAPH Conference Proceedings* (2006), pp. 779–786.

[KK98] KARYPIS G., KUMAR V.: A Fast and High Quality Multilevel Scheme for Partitioning Irregular Graphs. *SIAM J. Sci. Comput. 20*, 1 (1998), 359–392.

[KRN97] KANADE T., RANDER P., NARAYANAN P. J.: Virtualized Reality: Constructing Virtual Worlds from Real Scenes. *IEEE MultiMedia 4*, 1 (1997), 34–47.

[KSE*03] KWATRA V., SCHODL A., ESSA I., TURK G., BOBICK A.: Graphcut textures: Image and video synthesis using graph cuts. *ACM Transactions on Graphics, SIGGRAPH 2003 22*, 3 (July 2003), 277–286.

[KSU04] KANG S. B., SZELISKI R., UYTTENDAELE M.: Seamless Stitching Using Multi-Perspective Plane Sweep. Microsoft Research Technical Report MSR-TR-2004-48, June 2004.

[Low04] LOWE D. G.: Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Comput. Vision 60*, 2 (2004), 91–110.

[RBHB06] ROTHER C., BORDEAUX L., HAMADI Y., BLAKE A.: Autocollage. In *SIGGRAPH '06: ACM SIGGRAPH 2006 Papers* (New York, NY, USA, 2006), ACM Press, pp. 847–852.

[RKKB05] ROTHER C., KUMAR S., KOLMOGOROV V., BLAKE A.: Digital Tapestry. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition* (2005), pp. 589–596.

[SK02] SEITZ S. M., KIM J.: The Space of All Stereo Images. *Int. J. on Computer Vision 48*, 1 (2002), 21–38.

[SKG*98] SAWHNEY H. S., KUMAR R., GENDEL G., BERGEN J., DIXON D., PARAGANO V.: Videobrush: Experiences with Consumer Video Mosaicing. In *Proc.*

of the 4th IEEE Workshop on Applications of Computer Vision (1998), pp. 56–63.

[SS98] SHUM H.-Y., SZELISKI R.: Construction and Refinement of Panoramic Mosaics with Global and Local Alignment. In *Proc. Int. Conf. on Computer Vision* (1998), p. 953.

[SSS06] SNAVELY N., SEITZ S. M., SZELISKI R.: Photo Tourism: Exploring Photo Collections in 3D. In *SIGGRAPH Conference Proceedings* (2006), pp. 835–846.

[TAS03] TANAKA H., ARIKAWA M., SHIBASAKI R.: Design Patterns for Pseudo-3D Photo Collage. In *ACM SIGGRAPH Web Graphics* (2003), pp. 1–1.

[TFZ99] TORR P., FITZGIBBON A., ZISSERMAN A.: The Problem of Degeneracy in Structure and Motion Recovery from Uncalibrated Images. *Int. J. on Computer Vision 32*, 1 (1999), 27–44.

[VJ01] VIOLA P., JONES M.: Rapid Object Detection Using a Boosted Cascade of Simple Features. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition* (2001).

[WFH*97] WOOD D. N., FINKELSTEIN A., HUGHES J. F., THAYER C. E., SALESIN D. H.: Multiperspective panoramas for cel animation. In *Proceedings of SIGGRAPH 97* (Aug. 1997), Computer Graphics Proceedings, Annual Conference Series, pp. 243–250.

[WJV*05] WILBURN B., JOSHI N., VAISH V., TALVALA E.-V., ANTUNEZ E., BARTH A., ADAMS A., HOROWITZ M., LEVOY M.: High Performance Imaging Using Large Camera Arrays. In *SIGGRAPH Conference Proceedings* (2005), pp. 765–776.

[WQS*06] WANG J., QUAN L., SUN J., TANG X., SHUM H.-Y.: Picture Collage. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition* (2006), pp. 347–354.

[YEBM02] YANG J. C., EVERETT M., BUEHLER C., MCMILLAN L.: A real-Time Distributed Light Field Camera. In *Proc. Eurographics Workshop on Rendering* (2002), pp. 77–86.

[ZC04] ZHANG C., CHEN T.: A Self-Reconfigurable Camera Array. In *Proc. Eurographics Workshop on Rendering* (2004), pp. 243–254.

[Zha00] ZHANG Z.: A Flexible New Technique for Camera Calibration. *IEEE Trans. on Pattern Analysis and Machine Intelligence 22*, 11 (2000), 1330–1334.

[Zhe03] ZHENG J. Y.: Digital Route Panoramas. *IEEE MultiMedia 10*, 3 (2003), 57–67.

[ZMPP05] ZELNIK-MANOR L., PETERS G., PERONA P.: Squaring the Circles in Panoramas. In *Proc. Int. Conf. on Computer Vision* (2005), pp. 1292–1299.