# Mosaics of Scenes with Moving Objects

James Davis
Department of Computer Science
Stanford University
Stanford, CA 94305
jedavis@cs.stanford.edu
http://graphics.stanford.edu/~jedavis/panorama/

## Abstract

*Image mosaics are useful for a variety of tasks in vision and computer graphics. A particularly convenient way to generate mosaics is by 'stitching' together many ordinary photographs. Existing algorithms focus on capturing static scenes. This paper presents a complete system for creating visually pleasing mosaics in the presence of moving objects. There are three primary contributions. The first component of our system is a registration method that remains unbiased by movement—the Mellin transform is extended to register images related by a projective transform. Second, an efficient method for finding a globally consistent registration of all images is developed. By solving a linear system of equations, derived from many pairwise registration matrices, we find an optimal global registration. Lastly, a new method of compositing images is presented. Blurred areas due to moving objects are avoided by segmenting the mosaic into disjoint regions and sampling pixels in each region from a single source image.*

## 1 Introduction

Image mosaics are useful for a variety of tasks in vision and computer graphics; applications include: virtual environments, panoramic photography, and video compression. Although a number of hardware devices exist for creating panoramic images, software methods are gaining popularity. Typical software algorithms register or 'stitch' a sequence of digital images, and then composite all registered images into a final mosaic. Ideally, a user takes a hand–held consumer camcorder, pans around a scene capturing the region of interest, and obtains a complete description of the surrounding environment. In practice, several issues must be considered when designing a system.

First, a method to register any pair of images is needed. Some previous methods only register image pairs related by simple transforms. For example [McMillan95, Chen95] can only capture cylindrical panoramas, requiring the camera to be carefully mounted on a tripod, and panned around a single axis of rotation. A spherical environment can be captured by relaxing the requirements on camera orientation to allow pan/tilt freedom. In the case of a fixed center of projection, pairs of photographs are related by a two dimensional projection. Several methods for registering deformations of this kind have been developed [Szeliski96, Mann94]. If the camera center of projection is translated in $\Re^3$, then changes in visibility due to parallax preclude a mosaic consistent with all images. Nevertheless, some researchers have attempted to create a meaningful mosaic, by selectively including only certain regions of each image in the final mosaic [Rousso98, Wood97]. These methods have been demonstrated for only restricted classes of motion. In our solution, we do not attempt to address large changes in the center of projection of the camera.

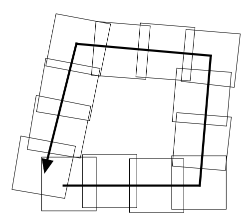


**Figure 1: Accumulated errors cause poor alignment near the end of an image sequence.**

Second, registering a large set of images introduces difficulties. Consider the sequence of images in Figure 1. Because most registration methods operate on images in a sequential, pairwise fashion, small errors in alignment accumulate from one pair of images to the next. In Figure 1, we expect the final image to coincide exactly with the first image. Although, each registration error is individually minute, the cumulative effect produces significant error in the position of the final image. This problem is commonly addressed by aligning images to a reference mosaic; after the first two images are registered, a mosaic is created; the third image is registered with respect to this mosaic, rather than the second image; Each new image is registered to the mosaic containing all previous images. A reference mosaic decreases the rate of error accumulation, however degradation persists. In addition to precise pairwise registration, a better method for producing globally consistent alignment is needed.

A third issue arises in scenes with moving objects. Estimation of image transform parameters can be biased by moving objects because moving regions of the image indicate a transformation different than the transformation due to the camera. For example, direct minimization of pixel intensity differences has been widely used to register image mosaics [Szeliski96, Chen95]. However, moving regions of high contrast contribute significant residual to the minimization, producing biased results. Feature tracking techniques are also widely used for image registration. Unfortunately, features arise on the boundary between foreground and background objects. These features move unpredictably with respect to the rest of the image, producing unreliable results. Motion analysis is an active area of research. For example, Sawhney and Ayer calculate the dominant motion of images, using an outlier mask to avoid bias [Sawhney96]. However, most current registration algorithms assume static scenes.

As a final issue, moving objects present challenges when compositing image sequences. Since the moving object translates with respect to the background, a mosaic consistent with every image in the sequence is not possible. Standard compositing techniques blend all available information and produce a blurred image in moving regions. Blurriness can occur to a lesser extent even if the scene remains static. Image distortions can arise due to the lens system, or inexact constraint of the center of projection; these local nonlinearities result in ambiguity and blurriness in the final mosaic. The addition of a local nonlinear adjustment after initial image registration substantially reduces minor artifacts [Shum97]. However, local registration adjustment is of limited use when the distortions are large or discontinuous, as is the case with moving objects. Although a theoretically correct mosaic is not possible when objects in the scene are moving, a typical user is only interested in a plausible reconstruction—a single copy of the moving object is often preferable to an indistinguishable smudge.

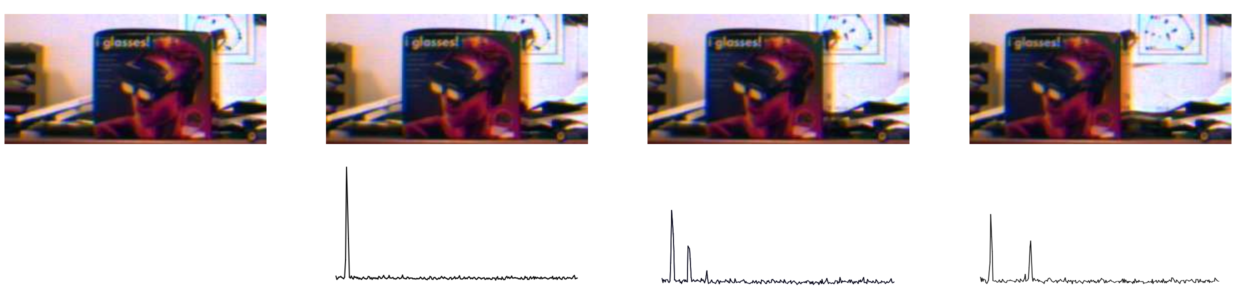
The remainder of this paper presents a complete system for creating mosaics in the presence of moving objects. Image sequences were captured with a hand–held consumer camcorder; no lens or camera calibration procedures were used. First, we review the Mellin transform, an existing technique that is unbiased by moving objects. The Mellin transform applies only to a limited class of image deformations, therefore a geometric extension is proposed that provides projective transformations. Next a new method for globally registering many images efficiently is presented. Global registration estimates are found by solving a system of linear equations constructed from the independently obtained pairwise projections. Finally, segmenting the mosaic into disjoint regions leads to a compositing method for scenes with moving objects.

## 2 Pairwise Registration

Mosaic creation requires that images be registered with respect to one another. Existing image registration techniques (see [Brown92] for a survey) are insufficient for creating mosaics with moving objects. As mentioned in section 1, many methods produce biased estimates of image registration when moving objects are present. The Mellin transform is a well known method for registering images which is not biased by moving objects, but is unfortunately restricted to aligning translation and rotation in the image plane [Cassant76, Reddy96]. After reviewing the Mellin transform, this limitation is addressed. A method to derive the actual projective registration parameters is presented.

### 2.1 Mellin Transform

The Mellin transform is based on phase correlation and the properties of Fourier analysis. We can find the



(a) (b) (c) (d)

**Figure 2: Image translation estimated with phase correlation is not biased by moving objects. Image (b) is a translated version of image (a). A single strong correlation peak is present. Moving a foreground object in image (c) degrades the amplitude of the background correlation peak, however the position of the peak is not affected. Instead, a smaller foreground correlation peak is created. Additional motion of the foreground object in image (d) changes the location of the foreground correlation peak.**

translational motion ($x_0$,$y_0$) between a pair of images {$I_1$, $I_2$} with phase correlation [Kuglin75]. The Fourier transform of each image {$\mathcal{I}_1$, $\mathcal{I}_2$} = {$\mathcal{F}[I_1]$, $\mathcal{F}[I_2]$}can be related using the phase shift theorem.

$$I_2(x,y) = I_1(x-x_0, y-y_0)$$
$$\mathcal{I}_2(\xi,\eta) = e^{-j2\pi(\xi x_0+\eta y_0)} \cdot \mathcal{I}_1(\xi,\eta)$$

The phase shift, $e^{-j2\pi(\xi x_0+\eta y_0)}$, can be computed as

$$\frac{\mathcal{I}_1^*(\xi,\eta)\,\mathcal{I}_2(\xi,\eta)}{\left|\mathcal{I}_1^*(\xi,\eta)\,\mathcal{I}_2(\xi,\eta)\right|} = e^{-j2\pi(\xi x_0+\eta y_0)}$$

where $\mathcal{I}^*$ is the complex conjugate of $\mathcal{I}$. Applying an inverse Fourier transform to the phase shift results in a delta function offset by exactly the amount of translational motion in the original image pair.

$$\delta(x-x_0, y-y_0) = \mathcal{F}^{-1}[e^{-j2\pi(\xi x_0+\eta y_0)}]$$

The resulting delta function may fall between samples if the actual translation is not a multiple of pixel width. Subpixel registration can be obtained by fitting the appropriate filter kernel to the pixels surrounding the image peak [Zieglar90]. I have found it sufficient to merely scan the resulting image for the greatest pixel intensity, and accept this as the delta location. When some region of the image is in motion, there are multiple correlation vectors present. As shown in figure 2, some energy from the primary delta function will be moved to a secondary peak, but the primary peak will remain, and its location will not be corrupted.

The Mellin transform extends phase correlation to handle images related by rotation. [Casasent76, Reddy96] According to the Fourier rotation and translation properties, the transforms will be related by

$$I_2(x,y) = I_1(x\cos\theta_0 + y\sin\theta_0 - x_0, -x\sin\theta_0 + y\cos\theta_0 - y_0)$$
$$\mathrm{I}_2(\xi,\eta) = e^{-j2\pi(\xi x_0+\eta y_0)} \cdot \mathrm{I}_1(\xi\cos\theta_0 + \eta\sin\theta_0, -\xi\sin\theta_0 + \eta\cos\theta_0)$$

We can see that the magnitude of spectra $\mathcal{I}_2$ is a rotated replica of $\mathcal{I}_1$. We can recover this rotation by representing the spectra of $\mathcal{I}_1$ and $\mathcal{I}_2$ in polar coordinates.

$$\left|\mathcal{I}_2(r,\theta)\right| = \left|\mathcal{I}_1(r,\theta-\theta_0)\right|$$

The Fourier magnitude images differ only by translation in this polar representation. Phase correlation as described above, can now be applied to the pair of polar Fourier images to recover $\theta_0$. Using $\theta_0$, a rotation can be applied so that $\mathcal{I}_1$ and $\mathcal{I}_2$ differ only by a phase shift. Now phase correlation is applied a second time to recover the original translation ($x_0$, $y_0$). At this point we have found all three parameters relating $I_1$ and $I_2$. Although my implementation assumes a fixed focal depth, the Mellin transform can additionally recover a scale factor by utilizing a log–polar transformation instead of a polar transformation in the above procedure.

In my experience, the Mellin transform is capable of robust registration with image translation up to one third the image width/height and rotation up to 45°. Greater image displacement or low quality source images causes significant degradation of the delta function. In this case the resulting peak is indistinguishable from surrounding noise. Robustness can be improved by observing that a poor estimate of rotation invalidates the second phase correlation procedure; a translation result without a peak is produced. The magnitude of the translation peak can be used to quantify the quality of a rotation estimate. If the translation peak is found to be below some threshold, the estimate of rotation is repeated using the next highest correlation peak.

## 2.2 Projection

Images photographed from a stationary center of projection are related by a projective transform. Registration obtained from the Mellin transform is limited to translation and rotation. Although [Szeliski96] uses phase correlation to initialize an iterative method, this limitation has prevented the Mellin transform from being used for the direct construction of mosaics. We present a derivation of the correct projective parameters using a few assumptions about camera geometry and the Mellin transform results.

Careful calibration of intrinsic camera parameters is possible, however we prefer to make the following standard assumptions which have proven adequate. Images are centered on the optical axis, have square pixels, and exhibit no skew. The following discussion assumes a known focal depth, however in section 3 a method for deriving this parameter is presented. The remaining degrees of freedom in the desired projection can be obtained as follows.

Image planes acquired from a stationary center of projection are tangent to a sphere. Figure 3a illustrates the camera geometry, drawn in two dimensions for clarity. The spherical angular rotation ($\alpha$,$\beta$) of the image plane $I_2$ can be trigonometrically determined from the focal depth $f$, and the lengths $x$ and $y$. One additional parameter relates $I_1$ and $I_2$: image plane rotation θ. This accounts for twisting the plane $I_2$ about the vector $\mathbf{f_2}$.

The image plane translation and rotation ($2x_0$,$2y_0$,$\theta_0$), estimated by the Mellin transform assumes coplanar images. Figure 3b shows the assumed geometry. For small (α,β) in the actual geometry, we can approximate ($x$,$y$,θ) ≈ ($x_0$,$y_0$,$\theta_0$). Assuming $f$ is known, this approximation allows ($\alpha$,$\beta$,θ) to be determined.

$$(\alpha,\beta,\theta) = \left(2\arctan\left(\frac{x_0}{f}\right), 2\arctan\left(\frac{\sqrt{f^2+x_0^2}}{y_0}\right), \theta_0\right)$$

The intrinsic camera parameters define a matrix $C$ that maps a 3D point onto the image plane. Because of the assumptions made above, $C$ reduces to a scaling matrix that accounts for the focal depth of the image plane. Furthermore, the motion given by ($\alpha,\beta,\theta$) can be described as a 3D rotation matrix $R$. Using these results, the desired projection matrix relating points in $I_2$ to points in $I_1$ can be written as $A = C\,R\,C^{-1}$ .

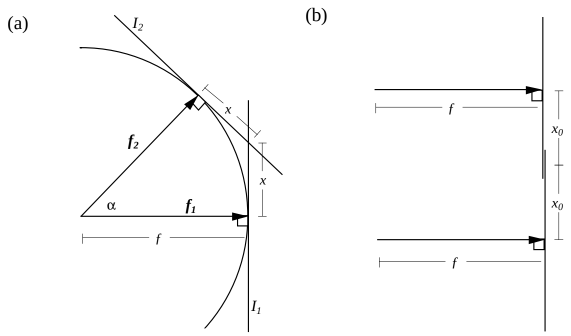



**Figure 3: (a) Actual camera geometry. (b) Geometry assumed by the Mellin transform.**

## 3 Global Registration

Image mosaics are often constructed from a sequence of many images. Registration errors accumulate over time so that images near the end of a sequence have a large cumulative error. Therefore, in addition to the previously presented method for registering a pair of images, we need to ensure that the final mosaic consisting of all images is globally registered.

Other researchers have addressed global registration. [Shum97] proposes a global registration strategy which establishes point correspondences in a set of images. Minimizing the projected difference of these points results in global alignment. The search required to determine many point correspondences can be quite slow, and as previously noted, feature based algorithms are potentially biased when working with moving objects. [Sawhney97] presents a method for iteratively registering multiple images while simultaneously correcting lens distortion. Although complex deformations are correctly registered with this technique, the large number of parameters makes computation prohibitive for more than a few images. The complexity of global alignment can be reduced by treating local registration independently. The following method has been used to globally align hundreds of images.

In the previous section we presented a method for finding a registration matrix $A_{ij}$ that will project image $I_i$ into the space of image $I_j$. It is possible to project $I_i$ directly into the space of $I_k$ by first projecting $I_i$ into the space of $I_j$ using $A_{ij}$, and then projecting the result into $I_k$ using $A_{jk}$. As illustrated in figure 4, projections chained in this fashion provide a method of mapping all images in the sequence $I_1 \ldots I_N$ into the space of $I_1$.

$$P_i = \prod_{j=2}^{i} A_{j,j-1}$$

The global registration matrix $P_i$ will project $I_i$ into the space of $I_1$. However, as noted previously, errors tend to accumulate and a better method is needed.

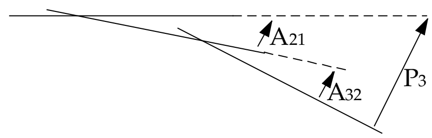



**Figure 4: The global registration matrix of an image can be expressed as the product of many local registration matrices.**

In order to improve the quality of global registration, let us suppose we have locally registered all spatially overlapping image pairs in addition to those that are adjacent in the image sequence. Furthermore, note that we can estimate the global projection $P_i$ of image $I_i$ by first projecting $I_i$ into any space $I_j$ and then using $P_j$ to project the result into the space of $I_1$.

$$\boldsymbol{A_{ij}\,P_j = P_i}$$

By considering many such pairs, we can build a sparse linear system of equations in which the $A_{ij}$ are known quantities obtained by pairwise image registration, and the matrix elements of $P_s$ $(1 \le s \le N)$ are unknowns to be found. Equating $P_i$ to the identity matrix will fix $I_i$ as the reference space. If we have locally registered more image pairs than the total number of images in the sequence, the system of equations will be over constrained. Solving this linear system of equations in a least squares sense will produce a set of global registration matrices that minimally deviate from the set of calculated pairwise projections. Since projection matrices contain one unnecessary degree of freedom it is useful to normalize the scale of $A_{ij}$ before solving the above system.

Attempting to register all image pairs results in $O(N^2)$ local image registrations. The number of registration attempts can be dramatically reduced by estimating the global image position of $I_i$ using $A_{i,i-1}$. Now $A_{ij}$ $(j<i)$ is only computed for those pairs of images which occupy overlapping regions of space. In practice, even fewer image pairs are necessary. Relating each image in the sequence to one other image drawn from much earlier in the sequence in addition to its immediate neighbors produces results with no visible alignment errors.

The camera focal depth is needed to determine pairwise projection in section 2.2. Rather than assume this is known, we can be obtained it by treating the focal depth $f$ as another variable and solving the resulting set of nonlinear equations. In our implementation, we iterate over values of $f$, solve the linear system of equations above, and then choose $f$ to minimize the residual. Although the coefficients of $A_{ij}$ must be computed for each value of $f$, the Mellin transform is only evaluated once.

The necessity of global registration is illustrated by a sequence of 72 images digitized at a resolution of 200x200. The sequence was captured with a hand–held camcorder panning repeatedly across a building facade. Figure 5 shows a mosaic created without using global registration. Parts of the mosaic are visibly out of alignment. In contrast, Figure 6 shows a 830x987 mosaic created with global registration. The quality of the final mosaic is significantly improved.

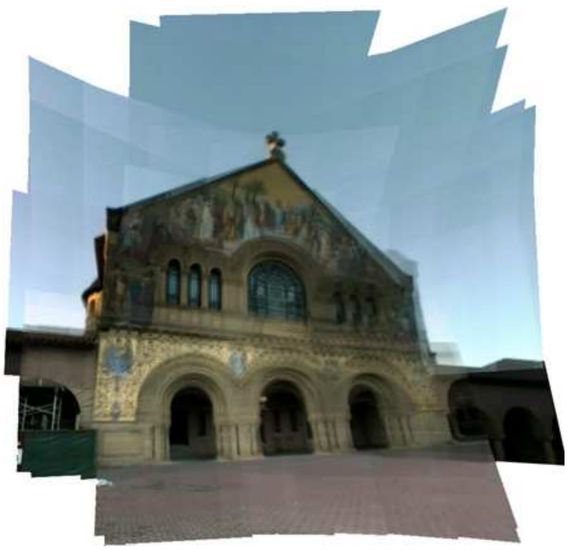


**Figure 5: Mosaic created without global registration. Errors accumulate causing poor alignment of images near the end of the sequence.**

**Figure 6: Mosaic created with global registration. Images in all parts of the sequence are mutually aligned.**

## 4 Compositing

After registering a sequence of images, the images are composited into a complete mosaic. Typical systems for building mosaics find all source images that map into a given destination pixel and compute some weighted blend of these source images. A weighting function that decreases near the boundary of a source image will prevent visible discontinuities due to gain adjustment between frames. Although more sophisticated blending functions exist [Burt83], any function sampling information from all available images produces blurred results in the case of moving objects. For some applications, blurred regions may be desirable; however many applications (such as virtual environments) prefer a single focused image.

When a focused image is desired, samples containing contradictory information should not be blended. An algorithm determining which source image contains the 'correct' imagery should be employed. The final mosaic will be comprised of a set of regions; the pixels in each region are sampled from a single 'correct' source image. If a moving object falls on the boundary between regions, it can be visibly truncated. Determining regions that do not exhibit object truncation is desirable. Previous methods for creating mosaics with moving objects have used a single frame to define a 'correct' region; areas outside this frame are blended with a temporal median filter. [Sawhney96]. A method for picking 'correct' regions of more general shape is presented below.

Consider the overlapping section in a pair of registered images. A method for dividing this section into two regions so that no discontinuities occur along the boundary is desirable. The relative difference between two images, $abs(I_i-I_j)/max(I_i-I_j)$, provides a measure of similarity between the source pair on a per pixel basis. Identical samples have a difference of zero, whereas pixels with samples drawn from different objects tend to have a larger intensity difference. A dividing boundary falling along a path of low intensity in the difference image will produce minimum discontinuity between regions in the final mosaic. The best path dividing the overlapping section can be found by using Dijkstra's algorithm [Cormen90] with weights drawn from the difference image. This path avoids areas where the source pair is inconsistent, including regions where objects are in motion. Inexact registration due to lens distortion or unintentional parallax also causes image discrepancy; segmenting mosaics is useful for this class of errors as well. Furthermore, paths through broad regions of image similarity are preferred because a low intensity path may accidentally be found through some moving object. Low pass filtering the difference image before applying Dijkstra's algorithm encourages desirable behavior.

Compositing a sequence of images using the above method is straightforward. Each source image is compared to the mosaic created thus far. A difference image of the overlapping section is created, and the shortest low intensity path traversing this section is found. On one side of this path we preserve pixels from the mosaic; on the other, we discard previous information in favor of samples from the current source image. Iterating over all source images produces a final mosaic without blurred regions. Figure 7 shows the final segmentation of the construction scene in Figure 9. Note that the moving truck falls entirely within one region since by design boundaries are not drawn across areas with moving objects. This algorithm relies upon the existence of at least one image containing a complete view of the moving object. If a view containing the whole object is unavailable the algorithm is forced to choose a region boundary bisecting the object.

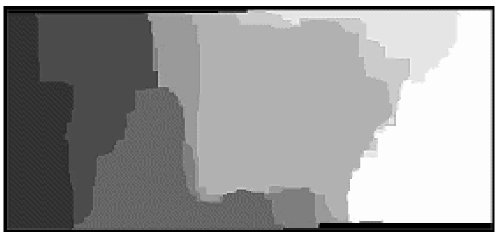


**Figure 7: Final segmentation of the construction scene mosaic in Figure 9 into disjoint regions. Note that the moving truck falls entirely within one of these regions**

To illustrate the advantages of the compositing method presented, a sequence of nine images was captured at a construction site. Figure 8 shows a mosaic composited by blending all images equally. The moving truck is almost unrecognizable due to blurriness. In contrast, Figure 9 shows that the truck is clearly visible when the compositing algorithm in this paper is used.

## 5 Discussion

A complete method for creating image mosaics in the presence of moving objects has been presented. In order to register images without bias the Mellin transform is used. Although the Mellin transform can not directly register projective transformations, a geometric derivation is shown that obtains the projective matrix. In the case of extremely short focal depth the approximations involved will not be valid and future work should explore the range over which acceptable results can be obtained. In addition, the Mellin transform can recover an image scaling term and inclusion of this parameter is an obvious avenue for growth.

A method for locally registering images is not sufficient. Accumulated registration errors can substantially degrade the quality of a final mosaic; therefore a new method of globally registering all images is developed. By decoupling local image registration from global alignment, mosaics with hundreds of images can be efficiently registered. Solving a linear system of equations which incorporates the local registration solutions produces a set of global alignment estimates which minimally deviate from the calculated pairwise registrations.

Conventional blending methods produce a final mosaic with blurred regions when moving objects are present. Superior results can be obtained by segmenting the mosaic into disjoint regions. Filling each region with pixels sampled from a single source image ensures that a mosaic without blurred regions is created. The segmentation operator presented here attempts to minimize visible discontinuities in the final mosaic. Although this work has concentrated on moving objects, many other factors can cause discontinuity. Future work might explore this. For instance, images with significant motion parallax might be combined into a single visually continuous mosaic.


## Acknowledgements

I owe Deborah Harber and Homan Igehy many thanks for hours spent helping prepare this paper. David Heeger and Pat Hanrahan have given invaluable advice on many topics including this one. Clay Kunz helped implement an early version of the software. This research was partially supported by DARPA contract DABT63-95-C-0085-P00006.


## References


Brown, L.G., “A Survey of Image Registration Techniques”, ACM Computing Surveys, 24:4, 1992.

Burt, P.J., Adelson, E.H. “A multiresolution spline with applications to image mosaics”, ACM Transactions on Graphics, 2(4):217-236, October 1983.

Casasent, D., Psaltis, D., “Position oriented and scale invariant optical correlation”, Applied Optics, 15 p1793-1799, 1976.

Chen, S.E., “Quicktime VR - an image based approach to virtual environment navigation”, Computer Graphics (Siggraph95), p29-38, August 1995.

Cormen, C., Leiserson, C., Rivest, R., Introduction to Algorithms, MIT Press, 1990.

Kuglin, C.D., Hines D.C., “The phase correlation image alignment method”, IEEE 1975 Conference on Cybernetics and Society, p163-165, 1975.

Mann, S., Picard, R.W., “Virtual bellows: Constructing high-quality images from video”, International Conference on Image Processing (ICIP94), p363-367, November 1994.

McMillan, L., Bishop, G., “Plenoptic Modeling: An image-based rendering system”, Computer Graphics (Siggraph95), p39-46, August 1995.

Reddy, B.S., Chatterji, B.N., “A FFT-Based Technique for Translation, Rotation, and Scale-Invariant Image Registration”, IEEE Trans. on Image Processing, 5:8, August 1996.

Rousso, B., Peleg, S., Finci, I., Rav-Acha, A., “Universal Mosaicing Using Pipe Projection”, Accepted to International Conference on Computer Vision (ICCV98), 1998.

Sawhney, H.S., Ayer, S., “Compact representations of videos through dominant multiple motion estimation”, IEEE Trans. on Pattern Analysis and Machine Intelligence, 18(8):814-830, August 1996.

Sawhney, H.S., Kumar, R., “True Multi-Image Alignment and its Applications to Mosaicing and Lens Distortion Correction”, IEEE Comp. Soc. Conf. on Computer Vision and Pattern Recognition (CVPR97), 1997.

Shum H., Szeliski, R., “Panoramic Image Mosaics”, Microsoft Research MSR-TR-97-23, 1997.

Szeliski, R., “Video Mosaics for Virtual Environments”, IEEE Computer Graphics and Applications, 16:22-30, 1996.

Wood, D.N., Finkelstein, A., Hughes, J., Thayer, C.E., Salesin, D.H., “Multiperspective Panoramas for Cel Animation”, Computer Graphics (Siggraph97), August 1997.

Ziegler, M., “Hierarchical motion estimation using the phase correlation method in 140 Mbits/s HDTV coding”, in Signal Processing of HDTV II(L. Chiariglione, ed.), p131-137, 1990.

**Figure 8: Mosaic created by compositing images with a blending function. Note the blurred region around moving objects.**

**Figure 9: Mosaic created using our compositing algorithm. Each region of the mosaic has samples drawn from a single source image.**