

# Universal Mosaicing using Pipe Projection \*

B. Rousso

S. Peleg

I. Finci

A. Rav-Acha

Institute of Computer Science  
The Hebrew University of Jerusalem  
91904 Jerusalem, ISRAEL  
E-Mail: {rousso,peleg}@cs.huji.ac.il

## Abstract

*Video mosaicing is commonly used to increase the visual field of view by pasting together many video frames. Existing mosaicing methods are effective only in very limited cases where the image motion is almost a uniform translation or the camera performs a pure pan. Forward camera motion or camera zoom are very problematic for traditional mosaicing.*

*A mosaicing methodology to allow image mosaicing in the most general cases is presented, where frames in the video sequence are transformed such that the optical flow becomes parallel. This transformation is an oblique projection of the image into a “viewing pipe” whose central axis is the trajectory of the camera.*

*The “pipe projection” enables to define high-quality mosaicing even for the most challenging cases of forward motion and of zoom. In addition, view interpolation, generating dense intermediate views, is used to overcome parallax effects.*

## 1 Introduction

The need to combine pictures into panoramic mosaics existed since the beginning of photography, as the camera’s field of view is always smaller than the human field of view. Also, many times large objects could not be captured in a single picture as is the case in aerial photography. Using a wide field of view (fish-eye) lens can be a partial solution, but the images obtained with such a lens have substantial distortions, and capturing an entire scene with the limited resolution of a video camera compromises image quality. A more common solution is photo-mosaicing: aligning, and pasting, frames in a video sequence, which enables a more complete view. Digital photography enabled new implementations for mosaicing [5, 18], which were

first applied to aerial and satellite images, and later used for scene and object representation.

The simplest mosaics are created from a set of images whose mutual displacements are pure image-plane translations. This is approximately the case with some satellite images. Such translations can either be computed by manually pointing to corresponding points, or by image correlation methods. Other simple mosaics are created by rotating the camera around its optical center using a special device, and creating a panoramic image which represents the projection of the scene onto a cylinder [14, 13]. However, the limitations to motion which is a pure rotation about the optical center limits the applicability of this approach.

In more general camera motions, that may include both camera translations and small camera rotations, more general transformation for image alignment are used [6, 8, 12, 16, 10]. In all cases images are aligned pairwise, using a parametric transformation like an affine transformation or planar-projective transformation. A reference frame is selected, and all images are aligned with this reference frame and combined to create the panoramic mosaic.

Aligning all frames to a single reference frame is reasonable when the camera is far away and its motion is mainly a translation and a rotation around the optical axis. Significant distortions are created when camera motions include rotations which are not about the optical axis.

To overcome most restrictions on mosaicing a new mosaicing methodology is presented, where images in a video sequence are transformed by an oblique projection of the image into a *viewing pipe* whose central axis is defined by the trajectory of the camera. After such a transformation the optical flow between frames becomes parallel, and they can be easily mosaiced. The *pipe mosaic* generated this way includes almost all details observed by the moving camera, where each

---

\*This research was partially funded by DARPA through ARL Contract DAAL01-97-0101 and by the European ACTS project AC074 “Vanguard”. Contact E-Mail: peleg@cs.huji.ac.il

region is taken from that image where it was captured at highest resolution.

A practical implementation of the general pipe mosaicing can be done by a process of collecting strips from image sequences satisfying the following conditions:

- Strips should be perpendicular to the optical flow.
- The collected strips should be warped and pasted into the panoramic image such that after warping their original optical flow - it becomes parallel to the direction in which the panoramic image is constructed.

Using these properties, an example for pipe mosaicing using strips will be given for the case of 2D affine motion. This covers most simple scenarios, and also zoom and forward motion. Generated mosaics have minimal distortions compared to the original images, as no global scaling is performed.

The strip collection process allows the introduction of a mechanism to overcome the effects of parallax by generating dense intermediate views. In some cases mosaics generated in this manner can be considered as linear push-broom cameras [9]. In pipe mosaicing, however, the broom may change its shape from a straight line to a circular or elliptic arc which is perpendicular to the optical flow.

## 2 Determining Strips for Mosaicing

The common method for construction of panoramic mosaics includes the collection of sections (“strips”) from each image and pasting these strips next to each other to become the mosaic. In the simple case of a camera which is moving horizontally, vertical strips are usually taken from each image and pasted side by side (see Fig. 1.a). This process can also be viewed as scanning the scene with a vertical broom [9, 19]. This vertical broom scans the entire sequence, extracts vertical strips along the sequence, and pastes them one next to the other to create the panoramic mosaic. The vertical broom is perpendicular to the horizontal optical flow, and after placing the strips in the panoramic image, the optical flow is pointing exactly to the direction from which the panoramic image is constructed (see Fig. 1.b).

The scanning broom must be perpendicular to the optical flow. A counter example is shown in Fig. 1.c, using a vertical scanning broom with vertical optical flow. When the optical flow is parallel to the scanning broom no mosaic is created as no new information will pass through the broom. The case where the scanning broom is perpendicular to a radial optical flow is

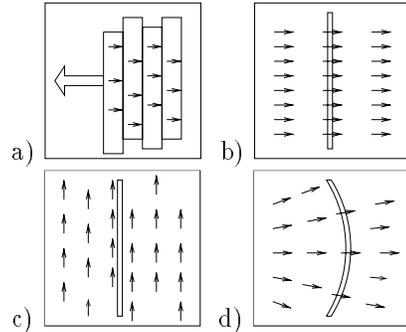


Figure 1: The relation between the mosaicing process using strips and the direction of the optical flow.

(a) The simple case of camera which is moving to the left. The optical flow points to the right, and vertical strips are collected. After pasting, The optical flow is parallel with the direction in which the panoramic image is built. (b) New information is passing through the scanning broom when the Optical Flow is perpendicular to the broom. (c) No new information is passing through the scanning broom when it is parallel to the optical flow. (d) When the scanning broom is set to be perpendicular to a radial optical flow its shape will be a circular arc.

shown in Fig. 1.d. The information from all images in the sequences will pass through the scanning broom, allowing to collect strips for pasting in the mosaic.

## 3 The Pipe Projection

Best mosaicing is based on concatenation of strips which are each perpendicular to the optical flow. However, while straight strips may be best for sideways motion and for panning, this is not the case when forward motion or zoom are involved.

In general camera motion, the optical flow is induced by camera translation and by camera rotation. The rotational part can be recovered and compensated for if needed, as it does not depend on the structure of the scene (see, for example, [15]). The translation (and zoom) induce radial optical flow which emerges from the FOE (Focus Of Expansion), except for the singular case of sideways translation in which the optical flow is parallel. Cases of radial optical flow are much more complicated for mosaicing as the optical flow is not parallel, and it depends on the structure of the scene.

The pipe representation which is described in this section handles camera translation and zoom, and simplifies mosaicing in these cases.

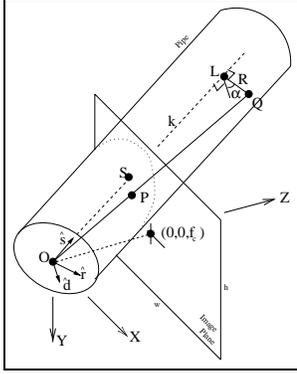


Figure 2: The pipe projection geometry. See text for details.

Given a sequence of images which were taken by a translating camera, we would like to transform the images in a way that the radial optical flow will turn into a parallel optical flow in the transformed representation. In order to do that, we project the 2D planar image onto a 3D cylinder, which we call a *pipe* (see Fig. 2). The axis of the pipe  $\hat{s}$  is chosen to pass through the optical center  $O = (0, 0, 0)$  and through the FOE  $S = (s_x, s_y, f_c)$ , where  $f_c$  is the focal length, and thus  $\hat{s} = S/|S|$ . Each image point  $P = (x, y, f_c)$  is projected onto its corresponding point  $Q$  on the pipe. The point  $Q$  is collinear with  $O$  and  $P$ , and its distance from the pipe's axis  $\hat{s}$  is  $R$ , where  $R$  is the radius of the pipe.

In the pipe representation of the image, the optical flow of each pixel  $Q$  on the pipe is now parallel to the direction of pipe's axis  $\hat{s}$ . This enables a simple mosaicing process on the pipe itself, as subsequent images, after being projected on the pipe, should only be shifted along the pipe in order to become aligned with previous images. This translation along the pipe does not reduce the resolution, as commonly happens in mosaicing methods which are based on alignment to a reference frame.

We use  $k$  and  $\alpha$  to determine the position in the pipe of a point  $Q$ , where  $k$  is the position along the axis  $\hat{s}$ , and  $\alpha$  is the angle from  $\hat{d}$ .  $\hat{d}$  and  $\hat{r}$  are unit vectors chosen to form a Cartesian coordinate system together with  $\hat{s}$ . The 3D position of a point  $(k, \alpha)$  on the pipe is  $Q = (Q_x, Q_y, Q_z) = k\hat{s} + R\cos(\alpha)\hat{d} + R\sin(\alpha)\hat{r}$ , and the corresponding pixel in the image plane for the point  $Q$  is  $P = (x, y, f_c) = (f_c Q_x/Q_z, f_c Q_y/Q_z, f_c)$ .

Pixels in the image whose original distance from the axis  $\hat{s}$  is less than  $R$  become magnified on the pipe, but when projected back to the image they restore their

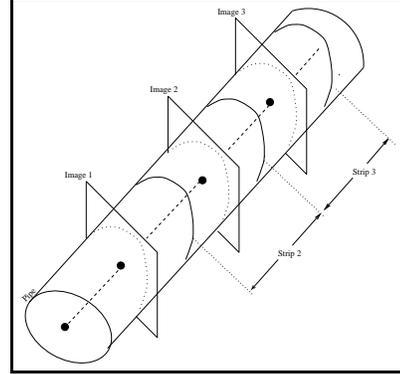


Figure 3: The relation between the pipe projection and the strips used for mosaicing. Each image contributes to the mosaic that strip for which best resolution is obtained from that image. For example, Strip 2 is in best resolution at the pipe projection of Image 2.

resolution. However, pixels with distance greater than  $R$  shrink on the pipe, thus losing their original resolution. Note that selecting  $R = \sqrt{f_c^2 + (\frac{w}{2})^2 + (\frac{h}{2})^2}$  where  $w$  and  $h$  are the width and height of the image, ensures that no pixel will reduce its resolution at the projection, as the intersection of the pipe with the image plane never occurs within the image boundaries. Smaller values of  $R$  will preserve the geometry while possibly reduce the resolution.

In the pipe representation, pipe images are aligned with each other by a simple translation, and the creation of the pipe mosaic involves taking the pixels with the best resolution among all projected images for every point in the pipe. The resolution is best preserved is around the intersection of the pipe with the image plane ( $Q_z = f$ ), and the resolution decreases as  $|Q_z - f_c|$  increases. This definition forms a strip which will be taken from that image having best resolution. An example is in Fig.3.

This pipe representation is a generalization of all mosaicing methods. Methods based on alignment to a reference frame can be simulated by viewing the pipe from the same orientation as the selected reference frame. Methods which are limited to pure sideways translation will give identical results as using a pipe mosaic, where the images are projected on the side of the pipe.

Cases like oblique view, forward motion, and zoom, can be handled well using the pipe projection, and give optimal results, while previous mosaicing methods may fail in these cases. The pipe representation can be generalized for handling complicated trajecto-

ries and rotations by concatenation of pipes along the path of the camera. We leave this for future work.

## 4 Pipe Mosaicing for Affine Motion

This section describes a practical implementation of pipe-mosaicing for the case of affine motion, as such a transformation can give good approximation for image motion in many cases. Numerous methods exist to recover the parameters of an affine transformation [11, 16]. Implementations for other motion models can be done in a similar manner.

### 4.1 Broom Shape in Affine Motion

The affine transformation can be expressed as:

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} x_n - x_{n-1} \\ y_n - y_{n-1} \end{pmatrix} = \begin{pmatrix} a + bx_n + cy_n \\ d + ex_n + fy_n \end{pmatrix} \quad (1)$$

where  $P_n = (x_{n-1}, y_{n-1})$  and  $P_{n-1} = (x_n, y_n)$  are corresponding points in images  $I_{n-1}$  and  $I_n$ , and the parameters of the affine transformation  $\mathcal{A}$  are  $(a, b, c, d, e, f)$ .  $(u, v)$  is the optical flow vector as a function of the position  $(x_n, y_n)$ . The transformation  $\mathcal{A}$  (and the optical flow) vary continuously along the sequence.

The scanning broom that will be used for mosaicing is a curve defined by  $\mathcal{F}(x, y) = 0$  that should be perpendicular to the optical flow. The normal to the curve  $\mathcal{F} = 0$  is in the direction  $(\frac{\partial \mathcal{F}}{\partial x}, \frac{\partial \mathcal{F}}{\partial y})$ , thus it should be in the same direction as  $(u, v)$ . This constraint can be expressed by:

$$\begin{pmatrix} \frac{\partial \mathcal{F}}{\partial x} \\ \frac{\partial \mathcal{F}}{\partial y} \end{pmatrix} = k \begin{pmatrix} u \\ v \end{pmatrix} = k \begin{pmatrix} a + bx + cy \\ d + ex + fy \end{pmatrix} \quad (2)$$

for some value of  $k$ . By integrating, when  $e = c$  we get the equation for the scanning broom:

$$0 = \mathcal{F}(x, y) = ax + dy + \frac{b}{2}x^2 + \frac{c+e}{2}xy + \frac{f}{2}y^2 + M \quad (3)$$

Note that this curve equation exists only when  $e = c$ . In most cases, the difference between the values of  $c$  and  $e$  is due to the rotation around the optical axis (Z axis) by  $\omega_z$  (angles in radians), such that it contributes  $-\omega_z$  to  $c$ , and  $+\omega_z$  to  $e$ . To approximately satisfy the condition  $e \approx c \approx \frac{c+e}{2}$  it is therefore sufficient to rotate the image by  $-\omega_z \approx \frac{e-c}{2}$  after the affine transformation is recovered, and then recompute the affine transformation.

As a result, Equation 3 defines a family of curves that are all perpendicular to the optical flow and can be used as the scanning broom.  $M$  is used to select a specific curve. We suggest that  $M$  will be set to the

value for which the broom contains maximum number of pixels within the image. If many options exit, then we suggest using a broom as close as possible to the center of the image to minimize lens distortions. This selection should ensure that pixels used in the mosaic will be from that image having best resolution at that location. The chosen curve can be considered as the intersection of the pipe with the image plane, assuming the radius of the pipe is chosen accordingly (See Fig. 3). Determining  $M$  can be considered as selecting the radius of the pipe  $R$ .

Eq. 3 can be easily understood for some simple cases.

- In the case of a uniform horizontal optical flow (either a small pan or a sideways translation), the affine transformation  $\mathcal{A}$  takes the form  $\mathcal{A} = (a, 0, 0, 0, 0, 0)$ , thus the selected broom becomes  $0 = \mathcal{F}(x, y) = ax + M$ , which is a vertical straight line (see Fig. 4.a).
- In the case of a uniform vertical optical flow (either a small tilt or a vertical translation), the affine transformation takes the form  $\mathcal{A} = (0, 0, 0, d, 0, 0)$ , thus the selected broom becomes  $0 = \mathcal{F}(x, y) = dy + M$ , which is a horizontal straight line (see Fig. 4.b).
- In the case of zoom or forward motion (towards a planar surface which is parallel to the image plane), the affine transformation takes the form  $\mathcal{A} = (0, b, 0, 0, 0, f)$ , where  $b$  is the scaling factor ( $f = b$ ). As a result, the selected broom becomes  $0 = \mathcal{F}(x, y) = \frac{b}{2}(x^2 + y^2) + M$ , which is a circle around the center of the image (see Fig. 4.c).

In general translations the scanning broom will be a circle around the Focus of Expansion (FOE). In more general cases the scanning broom may be approximated by an elliptic curve.

### 4.2 Determining Strips

The mosaic is constructed by pasting together strips taken from the original images. The shape of the strip, and its width, depend on the image motion. This section describes how to determine these strips in the case of an affine motion to conform with the methodology of the selection of best resolution. Strip selection for other motion models can be done in a similar manner.

We will use the following notation to describe the scanning broom along the sequence: The curve  $\mathcal{F}_n(x_n, y_n) = 0$  is the curve in Image  $I_n$ , in it's coordinate system, which corresponds to the affine transformation  $\mathcal{A}_n = (a_n, b_n, c_n, d_n, e_n, f_n)$ . This affine trans-

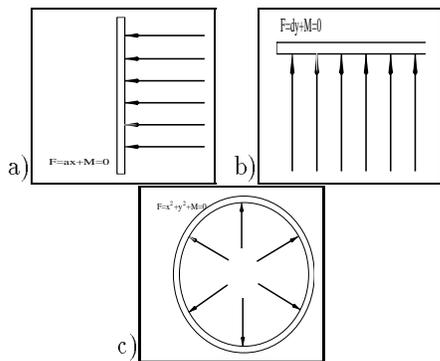


Figure 4: Examples for scanning broom.

(a) A vertical scanning broom is selected for horizontal motion. (b) A horizontal scanning broom is selected for vertical motion. (c) A circular scanning broom is selected for zoom and for forward motion.

formation  $\mathcal{A}_n$  relates points  $p_n$  in Image  $I_n$  to corresponding points  $p_{n-1}$  in Image  $I_{n-1}$  (see Fig. 5).

In order to determine the strip to be taken from Image  $I_n$ , the preceding frame,  $I_{n-1}$ , and the succeeding frame,  $I_{n+1}$ , should be considered. Let  $\mathcal{A}_n$  be the affine transformation relating points  $p_n = (x_n, y_n)$  in Image  $I_n$  to the corresponding points  $p_{n-1} = (x_{n-1}, y_{n-1})$  in Image  $I_{n-1}$ , and let  $\mathcal{A}_{n+1}$  be the affine transformation relating points  $p_{n+1} = (x_{n+1}, y_{n+1})$  in Image  $I_{n+1}$  to the corresponding points  $p_n = (x_n, y_n)$  in Image  $I_n$ .

Given the affine transformations  $\mathcal{A}_n$  and  $\mathcal{A}_{n+1}$ , the curves  $\mathcal{F}_n(x_n, y_n) = 0$  and  $\mathcal{F}_{n+1}(x_{n+1}, y_{n+1}) = 0$  are selected respectively (see Fig. 5.a-c). The curve  $\mathcal{F}_n(x_n, y_n) = 0$  in  $I_n$  corresponds to the curve  $\mathcal{F}'_n(x_{n-1}, y_{n-1}) = 0$  in  $I_{n-1}$  using the affine transformation  $\mathcal{A}_n$ . In the same way, the curve  $\mathcal{F}_{n+1}(x_{n+1}, y_{n+1}) = 0$  in  $I_{n+1}$  corresponds to the curve  $\mathcal{F}'_{n+1}(x_n, y_n) = 0$  in  $I_n$  using the affine transformation  $\mathcal{A}_{n+1}$ .

The strip that is taken from the image  $I_n$  is bounded between the two curves  $\mathcal{F}_n(x_n, y_n) = 0$  and  $\mathcal{F}'_{n+1}(x_n, y_n) = 0$  in  $I_n$  (see Fig. 5.a-c).

Using this selection, the first boundary of the strip will be described by the selected curve  $\mathcal{F}_n$ , thus will be exactly orthogonal to the optical flow with regard to the previous image. The second boundary of the strip is described by the curve  $\mathcal{F}'_{n+1}$  which is the projection of the curve  $\mathcal{F}_{n+1}$  onto the current image  $I_n$ , having the same property in the next image.

This selection of the boundaries of the strip ensures that no information is missed nor duplicated along the strip collection, as the orthogonality to the optical flow

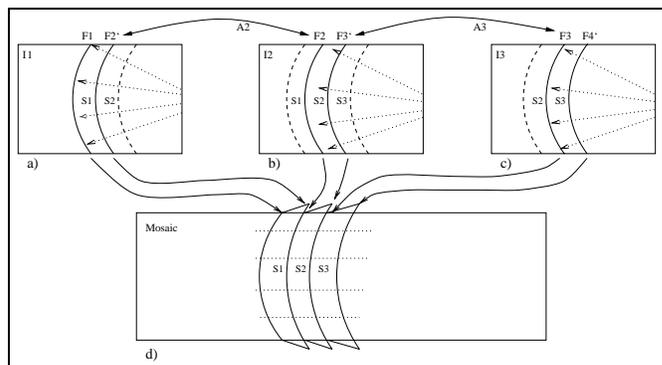


Figure 5: Example of cutting and pasting strips.

(a)-(c) Strips are perpendicular to the optical flow. Curve  $F_2$  is selected in Image  $I_2$  and Curve  $F_3$  is selected in Image  $I_3$ . The mapping of Curve  $F_3$  (in  $I_3$ ) into Image  $I_2$  using the affine transformation is Curve  $F'_3$ . The strip  $S_2$  taken from Image  $I_2$  is bound between curves  $F_2$  and  $F'_3$ . (d) Strips are warped and pasted so that the optical flow becomes parallel, their back is fixed (e.g.  $F_2$  in Strip  $S_2$ ) and their front (e.g.  $F'_3$  in Strip  $S_2$ ) is warped to match the back of the next strip.

is kept.

### 4.3 Pasting Strips

Consider the common approach to mosaicing where one of the frames is used as a reference frame, and all other frames are aligned to the reference frame before pasting. In term of strips, the first strip is put in the panoramic image as is. The second strip is warped in order to match the boundaries of the first strip. The third strip is now warped to match the boundaries of the *already warped* second strip, etc. As a result, the mosaic image is continuous. However, major distortions may be caused by the accumulated warps and distortions. Large rotations can not be handled, and cases such as forward motion or zoom usually cause unreasonable expansion (or shrinking) of the image.

To create continuous mosaic images while avoiding accumulated distortions, the warping of the strips should depend only on the adjacent original frames, independent of the history of previous distortions. This section describes how to warp the strips in the case of affine motion so that it simulates the pipe projection.

In our scheme, the back of each strip is never changed. This is the side of the strip which corresponds to the boundary between Image  $I_{n-1}$  and Image  $I_n$  and defined by  $\mathcal{F}_n$ . The front of the strip is warped to match the back side of the next strip. This is the boundary between Image  $I_n$  and Image  $I_{n+1}$

which is defined by  $\mathcal{F}'_{n+1}$ .

In the example described in Fig. 5.d, we warp the first strip such that its left side does not change, while its right side is warped to match the left side of the original second strip. In the second strip, the left side does not change, while the right side is warped to match the left side of the third strip, etc.

As a result, the constructed image is continuous. Also, if we warp the original optical flow as we did with the strips, the resulting flow will be parallel to the direction in which the panoramic mosaic is constructed. Moreover, no accumulative distortions are encountered, as each strip is warped to match just another original strip, avoiding accumulative warps.

The strips as pasted in the example of Fig. 5 remain curved. To match the definition of the pipe projection, however, the strips should be un-bent into straight edges before pasting into the mosaic. Un-bending the strips may cause distortions in some parts of the mosaic, and before visual presentation the mosaic should be re-bent depending on the desired viewpoint. The bending and un-bending, however, is not very critical when the strips are curved only slightly.

## 5 View Interpolation for Parallax

Taking strips from different images when the width of the strips is more than one pixel would work fine only without parallax. When parallax is involved, no single transformation can be found to represent the optical flow in the entire scene. As a result, a transformation that will align a close object will duplicate far objects, and on the other hand, a transformation that will align a far object will truncate closer objects. Also, rapid changes between aligning close and far objects might result in useless results.

In order to overcome the parallax problems in general scenes, instead of taking a strip with a width of  $N$  pixels, we can synthetically generate intermediate images, and use narrower strips. For example, we can take a collection of  $N$  strips, each with a width of one pixel, from interpolated camera views in between the original camera positions. In order to synthesize new views we can use various methods, such as optical flow interpolation [7, 17], trilinear tensor methods [15], and others. In most cases approximate methods will give good results. The creation of the intermediate views can involve only view interpolation, as in this application view extrapolation is not needed.

The use of intermediate views for strips collection gives the effect of orthographic projection, which avoids parallax discontinuities. This strategy can be combined with the methods that were described in the previous sections as a preliminary stage, such that a

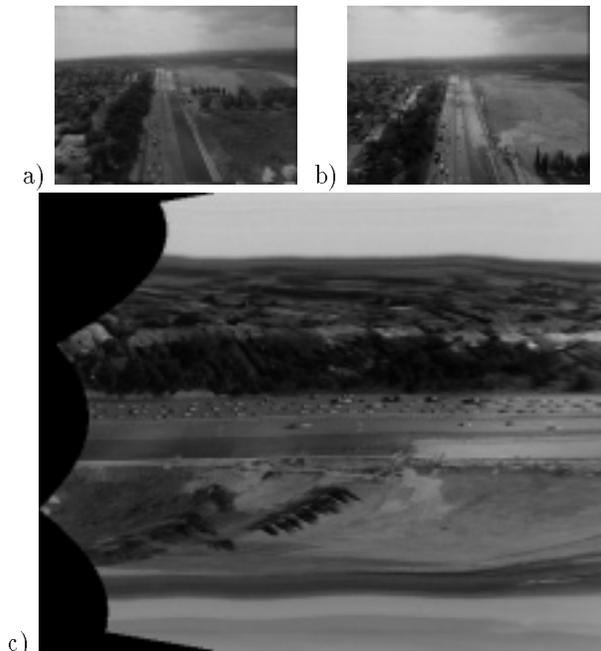


Figure 6: Pipe Mosaicing with a forward moving camera. **Please view this mosaic from all four directions.**

(a) First and (b) middle frames from a video sequence. Camera motion is forward, and the focus of expansion is inside the image.

(c) The pipe mosaic of the sequence. The curved boundary on the left corresponds (top-to-bottom) to the left-bottom-right edges of the first frame. Note that the road is always at full resolution.

complete solution is given for general motion in general scenes.

## 6 Experimental Results

In this section we show several cases which can not be handled with other mosaicing methods. An MPEG video will be deposited to visualize the original sequence and the process of creating the pipe mosaic.

### 6.1 Forward Motion

Forward camera motion used to be the classical case where traditional mosaicing fails. Pipe mosaicing can easily handle forward motion, as shown in Fig. 6. In Fig. 7 views of the pipe mosaic are generated from the left and from the right sides.

### 6.2 Zoom

During zoom, the resolution of the image increases while the field of view becomes smaller, causing the loss of the outside periphery from the next frame. Our process collects these circular peripheral strips, that disappear from one frame to the next, to construct

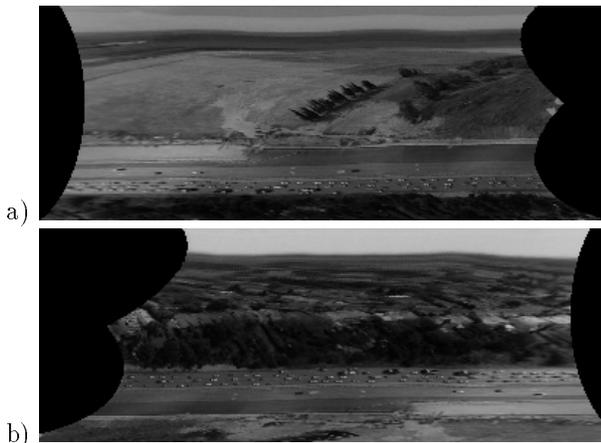


Figure 7: Viewing the pipe mosaic of the previous figure from two directions.

(a) Viewing direction is from left to right (of original sequence).

(b) Viewing direction is from right to left.

the mosaic. In terms of pipe projection, the axis of the pipe is perpendicular to the image plane, and passes through the optical center.

Assume the camera is located at the side of a long wall, with its optical axis parallel to the wall. In this case the closest parts of the wall are seen in high details at the edge of the image, while the distant parts of the wall are seen smaller closer to the center of the image. When zooming in, the further parts are magnified and get closer to the edge of the image, and the mosaic will therefore become a reconstruction of the wall at the highest possible resolution. Under some conditions the wall can even be reconstructed as viewed from the front, in uniform resolution all over. This result is shown in Fig. 8, where circular strips were collected and pasted in the panoramic image.

### 6.3 Complex Motion with Parallax

In Fig. 9 the camera is rotating and moving forward, while generating substantial parallax. Four images were mosaiced using the pipe projection. Without view interpolation, duplications and truncations are seen clearly due to the parallax, while with view interpolation these effects are reduced. The view interpolation was performed by optical flow interpolation.

## References

- [1] *ARPA Image Understanding Workshop*, Monterey, California, November 1994. Morgan Kaufmann.
- [2] *Fifth International Conference on Computer Vision*, Cambridge, MA, June 1995. IEEE-CS.

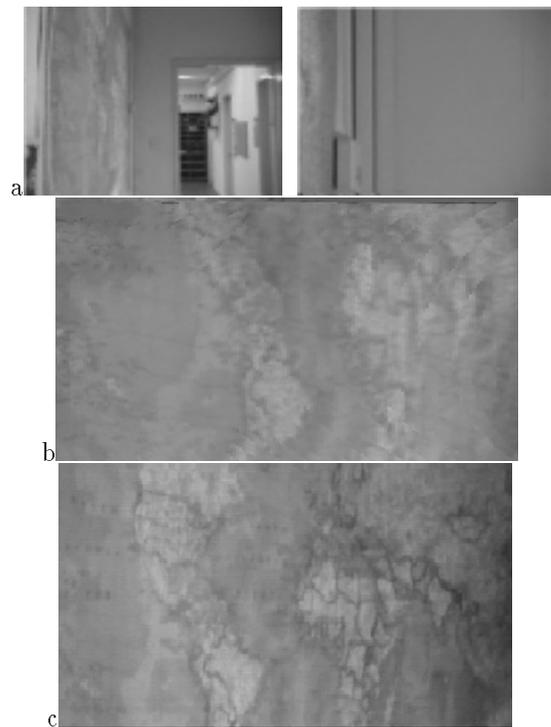


Figure 8: Panoramic mosaic from zoom.

(a) Two original images. A map is seen on a wall parallel to the optical axis. Left image is with smallest zoom, and right image is with largest zoom. (b) Reconstructed panoramic mosaic, which is somewhat similar to a frontal view of the map (c).

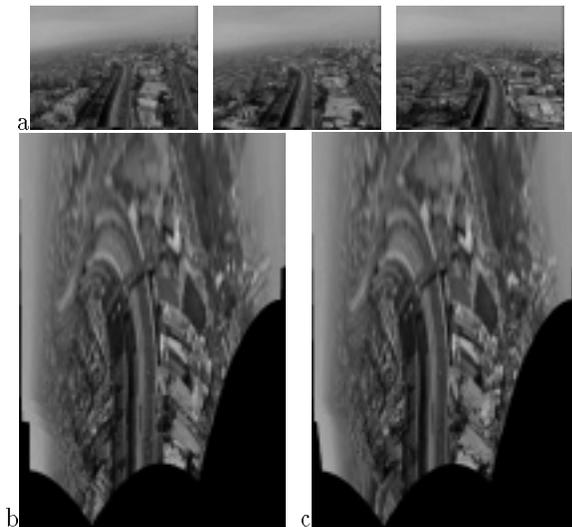


Figure 9: Handling parallax: sideways motion  
 (a) Three original images. The camera was performing composite translation and rotation, inducing substantial parallax. (b) Pipe mosaicing without view interpolation. Distant objects are duplicated, and close objects are truncated. (c) Using view interpolation reduces the distortions.

- [3] *Proc. IEEE Workshop on Representation of Visual Scenes*, Cambridge, MA, June 1995. IEEE-CS.
- [4] *IEEE Conference on Computer Vision and Pattern Recognition*, San Francisco, California, June 1996.
- [5] P.J. Burt and E.H. Adelson. A multiresolution spline with application to image mosaics. *ACM Trans. on Graphics*, 2(4):217–236, October 1983.
- [6] P.J. Burt and P. Anandan. Image stabilization by registration to a reference mosaic. In *ARPA Image Understanding Workshop* [1], pages 457–465.
- [7] S.E. Chen and L. Williams. View interpolation for image synthesis. In *SIGGRAPH*, pages 279–288, Anaheim, California, August 1993. ACM.
- [8] M. Hansen, P. Anandan, K. Dana, G. van der Wal, and P.J. Burt. Real-time scene stabilization and mosaic construction. In *ARPA Image Understanding Workshop* [1], pages 457–465.
- [9] R. Hartley and R. Gupta. Linear pushbroom cameras. In J.O. Eklundh, editor, *Third European Conference on Computer Vision*, pages 555–566, Stockholm, Sweden, May 1994. Springer.
- [10] M. Irani, P. Anandan, and S. Hsu. Mosaic based representations of video sequences and their applications. In *Fifth International Conference on Computer Vision* [2], pages 605–611.
- [11] M. Irani, B. Rousso, and S. Peleg. Detecting and tracking multiple moving objects using temporal integration. In G. Sandini, editor, *Second European Conference on Computer Vision*, pages 282–287, Santa Margherita, Italy, May 1992. Springer.
- [12] P. Jaillon and A. Montanvert. Image mosaicking applied to three-dimensional surfaces. In *12th International Conference on Pattern Recognition*, pages 253–257, Jerusalem, Israel, October 1994. IEEE-CS.
- [13] S. Mann and R. Picard. Virtual bellows: Constructing high quality stills from video. In *First IEEE International Conference on Image Processing*, Austin, Texas, November 1994.
- [14] L. McMillan and G. Bishop. Plenoptic modeling: An image-based rendering system. In *SIGGRAPH*, Los Angeles, California, August 1995. ACM.
- [15] B. Rousso, S. Avidan, A. Shashua, and S. Peleg. Robust recovery of camera rotation from three frames. In *IEEE Conference on Computer Vision and Pattern Recognition* [4], pages 796–802.
- [16] H.S. Sawhney, S. Ayer, and M. Gorkani. Model-based 2D & 3D dominant motion estimation for mosaicing and video representation. In *Fifth International Conference on Computer Vision* [2], pages 583–590.
- [17] S. Seitz and C. Dyer. Physically valid view synthesis by image interpolation. In *Proc. IEEE Workshop on Representation of Visual Scenes* [3].
- [18] R. Szeliski. Video mosaics for virtual environments. *IEEE Computer Graphics and Applications*, pages 22–30, March 1996.
- [19] J.Y. Zheng and S. Tsuji. Panoramic representation for route recognition by a mobile robot. *International Journal of Computer Vision*, 9:55–76, 1992.