Mosaicing with Generalized Strips *

Benny Rousso Shmuel Peleg Ilan Finci
Inst. of Computer Science
The Hebrew University of Jerusalem
91904 Jerusalem, ISRAEL

Abstract

Video mosaicing is commonly used to increase the effective visual field of view. Existing mosaicing methods are based on image alignment, and are effective only in very limited cases.

To overcome most restrictions, mosaicing is presented in this paper as a process of collecting strips. Strips which are perpendicular to the optical flow are cut out of the images, and are warped so that within each strip the optical flow will be parallel. These strips are then pasted into the mosaic. This approach enables to define mosaicing even for cases of forward motion and for zoom. View interpolation, generating dense intermediate views, is used to overcome parallax effects.

1 Introduction

An introduction and a survey of mosaicing methods can be found in [6]. We will only give a brief introduction focusing on the aspects relevant to this work.

Early mosaicing methods were used for aerial and satellite images. In both cases the objects in the scene are distant from the camera, and camera motion could be modeled as a translation parallel to the image plane with no parallax effects. Other methods use a camera which is rotating around the y axis passing through it's optical center without any translation. The resulting mosaic corresponds to a projection onto a cylinder [5].

Most methods select one frame to be a reference frame, towards which all other frames are warped [8, 3]. This approach uses 2D analysis to find motion based on an affine model or a general planar surface model, and allows somewhat more general camera motion. However, this approach can not handle parallax, and is restricted to small rotations (around the x or the y axis) with regard to the reference frame. Large rotations cause distortions when trying to perform the reprojection onto the reference frame. In addition, existing methods are not well defined for forward motion or for zoom.

To overcome most restrictions, mosaicing is defined here as 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 when warping their original optical flow it becomes parallel to the direction in which the panoramic image is constructed.

Using these properties, we define mosaicing methods for the case of 2D affine motion. This covers most simple cases, 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 at linear pushbroom cameras [2].

2 Mosaicing Using Strips

Construction of panoramic mosaics includes the collection of sections from each image and pasting these sections next to each other to become the mosaic. In the simple case of a camera which is moving horizontally, vertical sections are usually taken from each image and pasted side by side (see Fig. 1.a). In this case the process can also be viewed as scanning the scene with a vertical line. This vertical line scans the entire sequence, extracts vertical strips along the sequence, and pastes them one next to the other to create the panoramic mosaic. In this case the vertical line is perpendicular to the horizontal optical flow, and after placing the strips in the panoramic image, the optical

*This research was partially funded by DARPA through the U.S. Office of Naval Research under grant N00014-93-1-1202 and by the European ACTS project AC074 "Vanguard". Contact E-Mail: peleg@cs.huji.ac.il

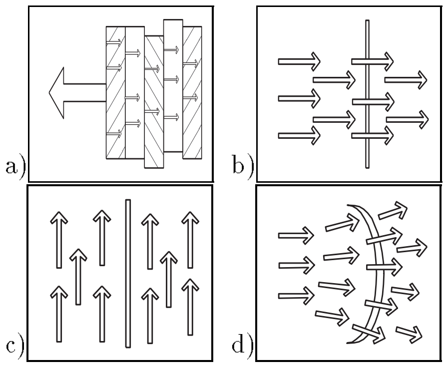

Figure 1: The relation between the mosaicing process and the direction of the Optical Flow.
(a) The simple case of camera which is moving to the left. The Optical Flow points to the right, and vertical strips are collected. After pasting, The Optical Flow is parallel with the direction in which the panoramic image is built. (b) New information is passing through a given line when the Optical Flow is perpendicular to the line. (c) No new information is passing through a given line when it is not perpendicular to the Optical Flow. (d) In the general case the line is set to be perpendicular to the Optical Flow.

flow is pointing exactly to the direction from which the panoramic image is constructed (see Fig. 1.b).

Using such a vertical scanning line with vertical camera motion, when the optical flow is parallel to this scanning line, (see Fig. 1.c), will not create any mosaic, as no new information will pass through the selected line.

In general, optimal results would be achieved by selecting a scanning line which is perpendicular to the optical flow (see Fig. 1.d). The information from all images in the sequences will pass through the scanning line, allowing to collect strips for pasting in the mosaic.

The requirement that the scanning line be perpendicular to the optical flow can be described for a pair of subsequent images I_{n-1} and I_n. If a point $p_n = (x_n, y_n)$ in Image I_n is on the scanning line, and corresponds to Point $p_{n-1} = (x_{n-1}, y_{n-1}) = (x_n - u, y_n - v)$ in Image I_{n-1}, then new information arrives to Point p_n from direction $(-u, -v)$, and for optimal results the direction of scanning line at point p_n should be perpendicular to $(-u, -v)$.

2.1 Affine Motion

In many cases the motion between two images is approximated by an affine transformation. Many methods exist to recover the parameters of an affine transformation [4].

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 (x_{n-1}, y_{n-1}) and (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.

We are looking for a line $\mathcal{F}(x, y) = 0$ such that it will be perpendicular to the optical flow. The normal to the line $\mathcal{F} = 0$ 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 of the scanning line:

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

This is a family of lines that are all perpendicular to the optical flow. M is used to select a specific line. We suggest that M will be set to the value for which the line contains maximum number of pixels within the image. If many options exit, then we suggest using a line as close as possible to the center of the image to minimize lens distortions.

Note that this line 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 ω_z, such that it contributes $-\omega_z$ to c, and $+\omega_z$ to e. To satisfy the condition $e = c$ it is therefore sufficient to rotate the image by $\omega_z \approx \frac{e-c}{2}$ after the affine transformation is recovered, and then recomute the affine transformation.

We will use the following notation to describe the scanning line along the sequence: The line $\mathcal{F}_n(x_n, y_n) = 0$ is the line in Image I_n, in it's coordinate system (x_n, y_n), which corresponds to the affine transformation $\mathcal{A}_n = (a_n, b_n, c_n, d_n, e_n, f_n)$. This affine transformation $\mathcal{A}_n$ relates points p_n in Image I_n to corresponding points p_{n-1} in Image I_{n-1} (see Fig. 3).

2.2 Special Cases

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

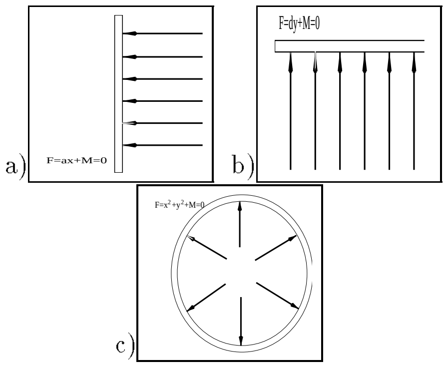

Figure 2: Examples for scanning line.
(a) A vertical scanning line is selected for horizontal motion. (b) A horizontal scanning line is selected for vertical motion. (c) A circular scanning line is selected for zoom and for forward motion.

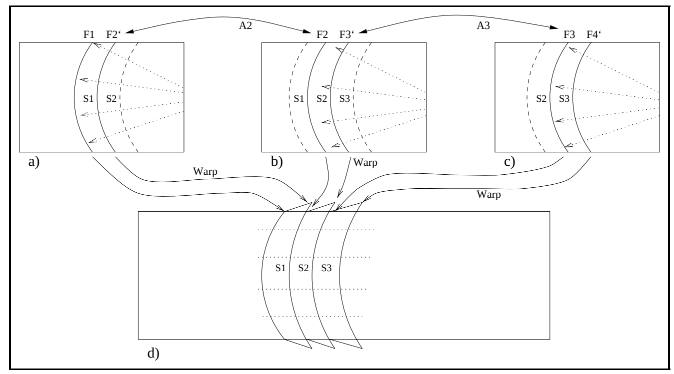

Figure 3: Cutting and pasting strips.
(a)-(c) Strips are perpendicular to the optical flow. (d) Strips are warped and pasted so that their back is fixed and their front is warped to match the back of the next strip.

- In the case of sideway motion (either small sideway rotation or sideway translation), the affine transformation $\mathcal{A}$ takes the form $\mathcal{A} = (a, 0, 0, 0, 0, 0)$, thus the selected line becomes $0 = \mathcal{F}(x, y) = ax + M$, which is a vertical line (see Fig. 2.a).
- In the case of upwards motion (either small rotation or translation), the affine transformation takes the form $\mathcal{A} = (0, 0, 0, d, 0, 0)$, thus the selected line becomes $0 = \mathcal{F}(x, y) = dy + M$, which is a horizontal line (see Fig. 2.b).
- In the case of zooming or forward motion (towards a planar surface which is parallel to the image plane), the affine transformation takes the form $\mathcal{A} = (0, s, 0, 0, 0, s)$, where s is the scaling factor. As a result, the selected line will become $0 = \mathcal{F}(x, y) = \frac{s}{2}(x^2 + y^2) + M$, which is a circle around the center of the image (see Fig. 2.c).

In the more general translation case, the result will be a circle around the Focus Of Expansion (FOE), assuming that the scene is planar and parallel to the image plane. More complex cases exist, in which the result will be generalized elliptic curve.

3 Cutting and Pasting of 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 select and to paste these strips.

3.1 Cutting Strips

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 lines $\mathcal{F}_n(x_n, y_n) = 0$ and $\mathcal{F}_{n+1}(x_{n+1}, y_{n+1}) = 0$ are selected respectively (see Fig. 3.a-c). The line $\mathcal{F}_n(x_n, y_n) = 0$ in I_n corresponds to the line $\mathcal{F}'_n(x_{n-1}, y_{n-1}) = 0$ in I_{n-1} using the affine transformation $\mathcal{A}_n$. In the same way, the line $\mathcal{F}_{n+1}(x_{n+1}, y_{n+1}) = 0$ in I_{n+1} corresponds to the line $\mathcal{F}'_{n+1}(x_n, y_n) = 0$ 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 lines $\mathcal{F}_n(x_n, y_n) = 0$ and $\mathcal{F}'_{n+1}(x_n, y_n) = 0$ in I_n (see Fig. 3.a-c).

Using this selection, the first boundary of the strip will be described by the selected line $\mathcal{F}_n$, 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 line $\mathcal{F}'_{n+1}$ which is the projection of the line $\mathcal{F}_{n+1}$ onto the current image I_n, having the same properties 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 is kept.

3.2 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 as 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 an adjacent original frame, independent of the history of previous warpings. 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. 3.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, were we to warp the original optical flow as we did with the strips, the resulting flow is continuous as well, and is 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 strips, avoiding accumulative warps.

4 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 L pixels, we can synthetically generate intermediate images, and use narrower strips. For example, we can take a collection of L 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 [1, 9], trilinear tensor methods [7], and others. In most cases approximate methods will give good results. The creation of the intermediate views can involve only view interpolation, as in most of the applications 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 complete solution is given for general motion in general scenes.

5 Experimental Results

In this section we show two cases which can not be done with other mosaicing methods. These results are still preliminary, but indicate the potential of this approach. Simple cases can be seen in [6].

5.1 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 the mosaic.

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. 4, where circular strips were collected and pasted in the panoramic image.

5.2 Sideway Motion with Parallax

In Fig. 5 the camera is moving sideways, generating substantial parallax. Vertical strips were collected according to the affine transformation that was recovered along the sequence, and the strips were pasted in the panoramic image. Without view interpolation, duplications and truncations are seen clearly, while with view interpolation these effects are reduced. The view interpolation was performed by optical flow interpolation.

Bibliography

[1] S.E. Chen and L. Williams. View interpolation for image synthesis. In *SIGGRAPH*, pages 279–288,

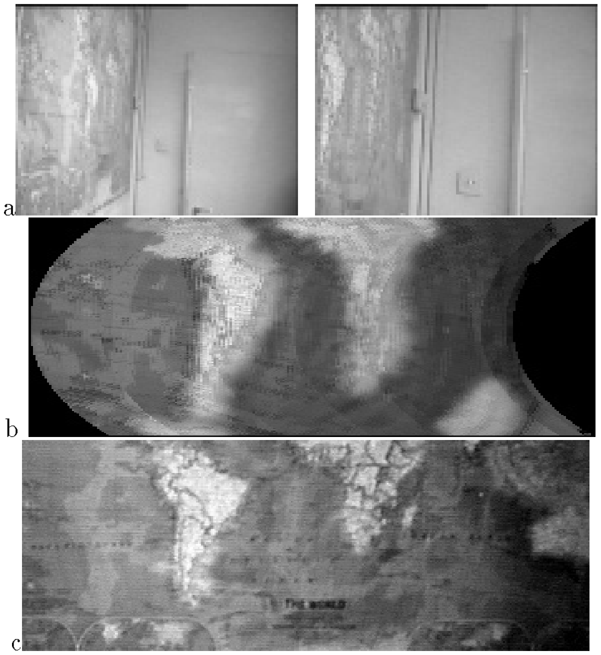

Figure 4: Panoramic mosaic for Zoom.
(a) Two original images. A map is seen on a wall parallel to the optical axis. (b) Reconstructed panoramic mosaic, which is similar to a real frontal view of the map (c).

Anahiem, California, August 1993. ACM.

[2] 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.

[3] M. Irani, P. Anandan, and S. Hsu. Mosaic based representations of video sequences and their applications. In *Fifth International Conference on Computer Vision*, pages 605–611, Cambridge, MA, June 1995. IEEE-CS.

[4] 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.

[5] Leonard McMillan and Gary Bishop. Plenoptic modeling: An image-based rendering system. In *SIGGRAPH*, Los Angeles, California, August 1995. ACM.

[6] S. Peleg and J. Herman. Panoramic mosaics with VideoBrush. In *IUW-97*, New Orleans, Louisiana, May 1997. Morgan Kaufmann.

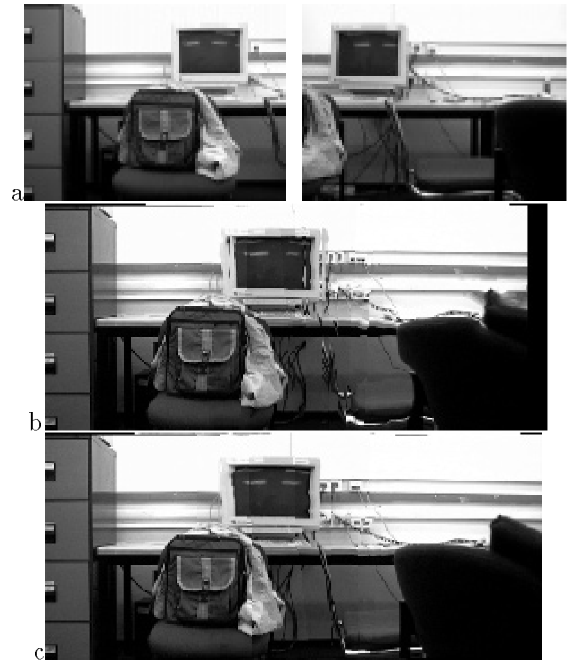

Figure 5: Handling parallax: sideways motion
(a) Two original images. (b) Mosaicing without view interpolation. Distant objects are duplicated, and close objects are truncated. (c) Using view interpolation reduces the distortions.

[7] 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*, pages 796–802, San Fransisco, California, June 1996.

[8] 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*, pages 583–590, Cambridge, MA, June 1995. IEEE-CS.

[9] S. Seitz and C. Dyer. Physically valid view synthesis by image interpolation. In *Proc. IEEE Workshop on Representation of Visual Scenes*, Cambridge, MA, June 1995. IEEE-CS.