Improved Video Mosaic Construction by Accumulated Alignment Error Distribution

Manuel Guillén González, Phil Holifield and Martin Varley
Department of Engineering and Product Design
University of Central Lancashire
Preston PR1 2HE, UK
m.guillen-gonzalez@uclan.ac.uk

Abstract

Mosaic techniques have been used to obtain images with a large field of view from video sequences by assembling individual overlapping images. In existing methods of mosaic construction only consecutive frames are aligned. Accumulation of small alignment errors occur, and in the case of the image path returning to a previous position in the mosaic (looping path), a significant mismatch between non-consecutive frames will result. A new method for ensuring the consistency of the positions of all images in a mosaic is proposed. From the resulting improvement in mosaic quality, the new method enables construction of mosaics with a very large field of view.

1 Introduction

In recent years, computers have experienced a huge expansion of transmission, storage and processing capabilities, at the same time they have become commonplace in our homes. Video capture technology is available at low prices, but often does not give good resolution or field of view. Video mosaicing is a convenient way to capture images without such limitations.

Since the beginning of photography, mosaics have been used to obtain images with a larger field of view by assembling two or more individual overlapping images [1]. Today's applications include the scanning of large realistic images from the real world to generate virtual environments [2].

The construction of mosaics from video begins with the alignment of successive images. Using their relative positions, the images can be integrated in a single large picture. Projective transformation and lens distortion have been successfully modelled [3], so image alignment is not a major problem in mosaic construction. The aim of this paper is to show that a new step must be introduced in the mosaicing process to account for problems that occur when the camera follows a loop (looping path). This is the case, for example, when a camera pans in one direction then pans back to the starting position. In almost all existing methods of image registration, consecutive frames of a video sequence are aligned. Accumulation of small alignment errors occur, and in the case of the image path returning to a previous position in the mosaic, a significant mismatch between non-consecutive frames will result.

BMVC 1998 doi:10.5244/C.12.38

It has been shown [4, 5] that instead of aligning successive images, the alignment can be done between an image and the actual mosaic as it is being composited. This may be an improvement with respect to the frame-to-frame alignment method, but loops involving large numbers of images result in distortions in the mosaic.

A new method for ensuring the consistency of the positions of all images in the mosaic is proposed, resulting in general improvement of mosaic quality and making it possible to create mosaics with a very large field of view, including spherical mosaics. The spherical mosaic is a two-dimensional mosaic mapped onto a sphere, and has applications in virtual reality environment maps [6].

2 Mosaic Construction

The basic processing steps involved in video mosaic construction are well known [7], and can be summarised as follows:

Image alignment: Determines the transformation that aligns two successive images, or one image with the current mosaic. In some cases simple translation and rotation operations correctly describe the transformation, in others, a projective transformation is needed.

Image integration: Consists of the selection of non-overlapping areas in the images that will contribute to the final mosaic or the combination of pixel intensities from overlapping images. Further blending of neighbour images is necessary to reduce the visibility of seams due to differences in intensities.

2.1 Image Alignment with Progressive Complexity

To determine the transformation that aligns two images, the different existing techniques can be divided in two types. The first identifies and matches common features in a pair of images such as lines [8], corners, text [9], etc. and uses them as references to align the two images. This method imposes limitations on the content of the images being aligned, for they must contain such features. The second type finds the transformation that minimizes the sum of the squared intensity errors for all overlapping pixels as shown in (1), therefore relying on the pixel intensities as features. For this method to work properly, there must be intensity variations in the images.

$$E = \sum_{n}^{N} \left[\mathrm{I}_i(x_n, y_n) - \mathrm{I}_j(x'_n, y'_n) \right]^2 \tag{1}$$

where I_i is image i
I_j is image j after the transformation is applied
N is the number of overlapping pixels
x_n, y_n and x'_n, y'_n are related by the transformation matrix

A rigid transformation would only involve translation and rotation of the images, while a projective transformation requires more parameters to be considered. The distortion

introduced in the acquisition process by the camera lens must also be modelled and corrected if accurate alignments are to be obtained.

The rigid transformation that aligns image I_i with image I_j is given using homogeneous coordinates in matrix form as follows.

$$\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & d_x \\ \sin\theta & \cos\theta & d_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ w \end{bmatrix} \tag{2}$$

where d_x, d_y are the translation in pixels and θ is the angle of rotation.

It has been shown that in image alignment, the use of transformation models with progressive complexity reduces the computation cost [4, 10]. Although for the rigid transformation there are only three parameters to be computed, two for the translation and one for the angle of rotation, a simpler model involving translation solely can be used initially for the alignment. Then, using this translation component, the angle of rotation can be computed.

```
DO {
        shift = CALC_TRANSLATION(image i, image j)
        TRANSLATE_IMAGE(image j, shift)

        angle = CALC_ROTATION(image i, image j)
        ROTATE_IMAGE(image j, angle)

} UNTIL better accuracy cannot be achieved
```

Figure 1: Steps for calculating the translation and rotation alignment between two overlapping images using transformation models with progressive complexity.

This particular implementation starts by finding the translation (`CALC_TRANSLATION`) that best aligns the pair of images, that is, parameters d_x and d_y in (2). This offset does not account for the rotation between the images, but since the angle of rotation is small between successive images it may be neglected at this stage. Then, once the translation has been calculated, the angle θ is worked out (`CALC_ROTATION`). This process of adjusting translation and then rotation is repeated in several passes until better accuracy cannot be achieved. This normally requires no more than 2 or 3 passes. The steps are summarised in Figure 1.

Although the accuracy that can be achieved in image alignment is excellent when assessed by the human eye, it is not error-free. A small error is always present and will manifest itself after a number of successive image alignments.

2.2 Translation

Laplacian pyramids have been used for the computation of translations. Smaller images are created from the original images by averaging blocks of pixels. A translation is computed for these smaller images which is then used as an initial

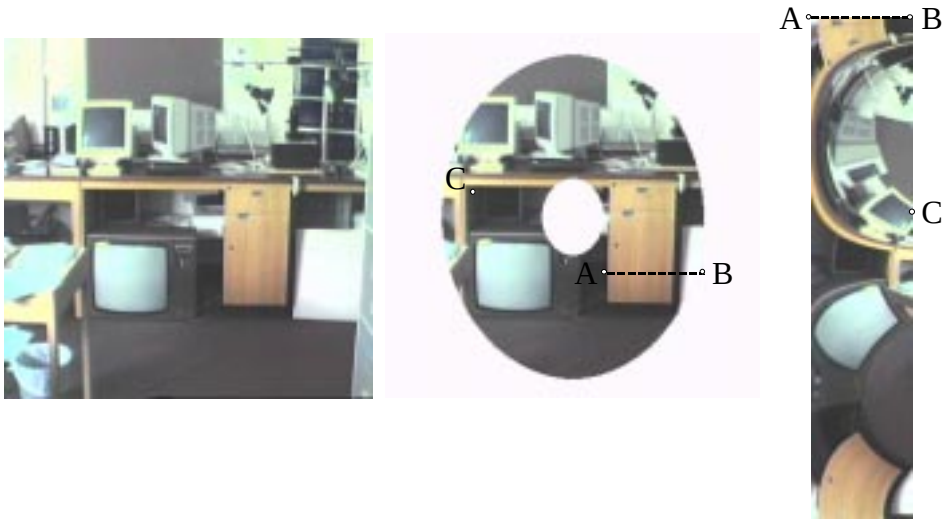
position to compute the translation of the original images. A detailed description can be found in [10].

It has been found that it is important to follow all local minima to the next pyramid level, especially for document images where false matches at the lower resolution levels may occur due to the repetitive nature of text lines.

2.3 Rotation

Rotating an image is a time consuming operation, in particular when subpixel accuracy by means of interpolation is necessary. The process that has been used for the computation of the angle of rotation between two images involves the mapping of a circle onto a rectangle as shown in Figure 2. This warping takes a radial line from the source image (e.g. Figure 2a) and maps it into a row (e.g. Figure 2c), corresponding directly to a mapping from the polar coordinate system to the Cartesian coordinate system, i.e. $(r, \theta) \rightarrow (x, y)$.

A vertical translation in the warped image (Figure 2c) is equivalent to a rotation in the original image (Figure 2a). Thus the problem of minimising the error function for different angles is reduced to a vertical translation matching, which is computationally less expensive and uses the same algorithms developed for translation alignment.



(a) (b) (c)

Figure 2: By warping from the polar to the Cartesian coordinate system, finding the alignment angle between two images is reduced to a vertical matching.
(a) Original image. (b) Area to be warped onto a rectangle. (c) Effect of warping a circle onto a rectangle.

2.4 Image Integration

Once the position of the images are known they can be integrated in the mosaic. Each pixel in the mosaic is taken from the image whose centre is the nearest among all image centres. This scheme corresponds to Voronoi tessellation [11], which, given the

position of the centres of the images, defines a polygonal area for each image that will be pasted on the mosaic.

3 Looping Path Problem

In constructing mosaics from video sequences, almost all existing methods have used parameters computed by successive image alignment. Cumulative alignment errors occur when the position of images in the mosaic is based on successive image alignment only.

Although good alignment is achieved between successive images, cumulative errors cause poor alignment when the image path follows a loop, i.e. when the same area of the scene is covered by images which are distant in the sequence. This problem has only been identified in literature [4, 5], and has not been satisfactorily solved.

In Figure 3, assuming a perfect loop has been followed by the camera, images 1 and 60 should overlap perfectly, but misalignment error occurs due to accumulation of small errors in each successive image alignment. The effects of the *looping path problem* are dramatic when large numbers of images are involved in the loop. In addition, the misalignment of neighbour images is unavoidable even when the frame-to-frame displacement has been computed very accurately.

The alignment between images 1 and 60 in Figure 3 is inconsistent with the position of the rest of the images. Previous attempts at solving this inconsistency align the images with the mosaic as it is being composited [12]. Using this approach will result in a poor quality mosaic when a large number of images are involved in a loop, which is the case for large field of view mosaics. There will be cases where the next image to be aligned with the current mosaic will need to fit two or more different transformations and distortion will be inevitable (e.g. image 60 in Figure 3 will be distorted when aligned with image 59 and with image 1).

A new step must be introduced in the mosaicing process to account for the looping path problem. The proposed solution seeks to distribute the accumulated error of the positions of all images in the mosaic.

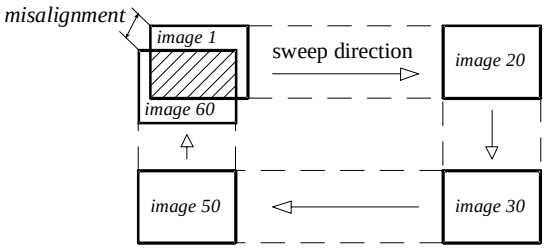


Figure 3: Misalignment error between image 1 and image 60 due to accumulation of small errors in successive image alignment.

3.1 Solution

Neighbour images are those which share a boundary in the mosaic. Each pair of neighbour images are related by a relative position $\mathbf{t}_{ij}$ computed using the alignment

method explained in section 3. For the rigid model, a translation (d_x, d_y) and a rotation angle d_θ of image j with respect to image i, correctly describes their relative position.

$$\mathbf{t}_{ij} = \begin{pmatrix} d_x \\ d_y \\ d_\theta \end{pmatrix} \tag{3}$$

A premise is introduced here: the relative position of a neighbour pair of images can be modified slightly without introducing a visible loss in quality. Such a change from its computed position must not exceed a fraction of a pixel if the seam is to remain unnoticeable.

$\mathbf{T}_{ij}$ represents the correct relative position that aligns images i and j consistently along with all other images in the mosaic. The cost for this consistency is a slight modification $\mathbf{\Delta}_{ij}$ of the computed relative positions of the images.

$$\mathbf{T}_{ij} = \mathbf{t}_{ij} + \mathbf{\Delta}_{ij} \qquad\qquad \mathbf{\Delta}_{ij} = \begin{pmatrix} \delta_x \\ \delta_y \\ \delta_\theta \end{pmatrix} \tag{4}$$

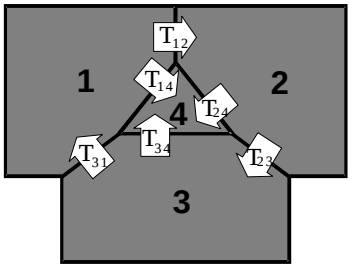


Figure 4: Example of mosaic composed from a sequence of 4 images.

In the sequence of 4 images shown in Figure 4, the pairs (1,2), (2,3), (3,4), (3,1), (1,4), (2,4) are neighbour images. The transformations that align them are $\mathbf{t}_{12}$, $\mathbf{t}_{23}$, $\mathbf{t}_{34}$, $\mathbf{t}_{31}$, $\mathbf{t}_{14}$ and $\mathbf{t}_{24}$. An equation can be established for each possible route connecting the images:

$$\begin{array}{lll}
\mathbf{T}_{12} \oplus \mathbf{T}_{23} = \mathbf{T}_{13} & \therefore & (\mathbf{t}_{12} + \mathbf{\Delta}_{12}) \oplus (\mathbf{t}_{23} + \mathbf{\Delta}_{23}) = (\mathbf{t}_{13} + \mathbf{\Delta}_{13}) \\
\mathbf{T}_{12} \oplus \mathbf{T}_{24} = \mathbf{T}_{14} & \therefore & (\mathbf{t}_{12} + \mathbf{\Delta}_{12}) \oplus (\mathbf{t}_{24} + \mathbf{\Delta}_{24}) = (\mathbf{t}_{14} + \mathbf{\Delta}_{14}) \\
\mathbf{T}_{23} \oplus \mathbf{T}_{34} = \mathbf{T}_{24} & \therefore & (\mathbf{t}_{23} + \mathbf{\Delta}_{23}) \oplus (\mathbf{t}_{34} + \mathbf{\Delta}_{34}) = (\mathbf{t}_{24} + \mathbf{\Delta}_{24}) \\
\mathbf{T}_{31} \oplus \mathbf{T}_{14} = \mathbf{T}_{34} & \therefore & (\mathbf{t}_{31} + \mathbf{\Delta}_{31}) \oplus (\mathbf{t}_{14} + \mathbf{\Delta}_{14}) = (\mathbf{t}_{34} + \mathbf{\Delta}_{34})
\end{array} \tag{5}$$

where $\oplus$ means composition of transformations.

The minimum values of $\mathbf{\Delta}_{ij}$ that satisfy equations 5 give the set of relative positions $\mathbf{T}_{ij}$ that consistently align all images in the mosaic.

Although a solution can be found that minimises $\mathbf{\Delta}_{ij}$, its implementation becomes impractical for a large number of images, which is the case in reality. A different approach to the problem is therefore necessary.

3.2 Proposed algorithm

The proposed method for consistently aligning all images in the mosaic is explained in this section.

For a sequence of N images, the relative positions $\{ \mathbf{t}_{01}, \mathbf{t}_{12}, \mathbf{t}_{23}, \ldots \mathbf{t}_{N-2\ N-1}\}$ that align successive images are computed. Then, the initial positions of the images in the mosaic $\{ \mathbf{P}_0, \mathbf{P}_1, \mathbf{P}_2, \ldots \mathbf{P}_{N-1}\}$ can be calculated, as shown in equation 6, by composition of the transformations that align successive images in the sequence.

$$\mathbf{P}_0 = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \qquad \mathbf{P}_i = \mathbf{P}_{i-1} \oplus \mathbf{t}_{i-1\,i} \qquad 1 \le i \le N-1 \tag{6}$$

where the meaning of $\oplus$ is given in (7).

$$\mathbf{P}_i \oplus \mathbf{t}_{ij} = \begin{pmatrix} Px_i \\ Py_i \\ P\theta_i \end{pmatrix} + \begin{pmatrix} t'x_{ij} \\ t'y_{ij} \\ t\theta_{ij} \end{pmatrix} \quad \text{where} \begin{pmatrix} t'x_{ij} \\ t'y_{ij} \end{pmatrix} = \begin{pmatrix} \cos(P\theta_i) & -\sin(P\theta_i) \\ \sin(P\theta_i) & \cos(P\theta_i) \end{pmatrix} \begin{pmatrix} tx_{ij} \\ ty_{ij} \end{pmatrix} \tag{7}$$

So far this corresponds to successive image alignment.

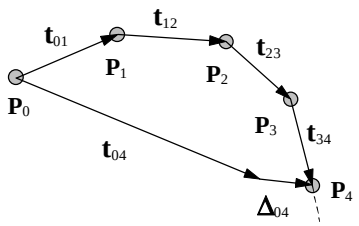


Figure 5: Relation between the positions of the images in the mosaic $\mathbf{P}_0$, $\mathbf{P}_1$, $\mathbf{P}_2$, $\mathbf{P}_3$, $\mathbf{P}_4$ and the relative positions $\mathbf{t}_{01}$, $\mathbf{t}_{12}$, $\mathbf{t}_{23}$, $\mathbf{t}_{34}$ that align successive images. $\boldsymbol{\Delta}_{04}$ is the error between the computed relative position of images 0 and 4 ($\mathbf{t}_{04}$) and their actual relative position in the mosaic ($\mathbf{P}_4 - \mathbf{P}_0$). The circles represent the centres of the images.

Let $\boldsymbol{\Delta}_{ij}$ be the difference between the constant relative position of images i and j ($\mathbf{t}_{ij}$) and their relative position in the mosaic ($\mathbf{P}_j - \mathbf{P}_i$), which will be modified. $\boldsymbol{\Delta}_{ij}$ represents the error between the computed relative position of images i and j and their actual relative position in the mosaic.

$$\boldsymbol{\Delta}_{ij} = \mathbf{t}_{ij} - (\mathbf{P}_j - \mathbf{P}_i) \tag{8}$$

Initially, for successive images (i.e. $j = i+1$), $\boldsymbol{\Delta}_{ij} = 0$. For the rest of the neighbour images $\boldsymbol{\Delta}_{ij} \neq 0$ due to the accumulated error in the successive alignments between the images i and j, that is, the error to be reduced. An analogy with a physical model is introduced, consisting of a network of connected nodes representing the centres of the images on which forces are exerted in order to change their position. The links

between nodes are defined by the transformations that align neighbour images (see Figure 5). In this analogy $f(\mathbf{\Delta}_{ij})$ represents the force pushing image i towards the right position with respect to image j, where the function f will be defined later in this section.

Let $\{n_1, n_2, n_3, \ldots n_m\}$ be the m neighbour images of image i. Each of its neighbour images will exert a force upon image i, $\mathbf{\Delta}_i$ is the resultant summation of these forces.

$$\mathbf{\Delta}_i = \sum_{k=1}^{m} \mathbf{\Delta}_{i\,n_k} \tag{9}$$

The refinement process that leads to a consistent set of positions is accomplished in an iterative fashion. For each iteration the forces acting on all images are calculated, then their positions are modified accordingly. The loop ends when equilibrium is achieved, i.e. $\mathbf{\Delta}_i \approx 0, \quad 0 \le i \le N-1$.

The positions of the images ($\mathbf{P}_i$) are modified by small increments. These increments are a function of $\mathbf{\Delta}_i$.

$$\mathbf{P}_i \rightarrow \mathbf{P}_i + f(\mathbf{\Delta}_i) \tag{10}$$

The performance of the function $f(\mathbf{\Delta}_i)$ is assessed by inspection of the overall distortion $\mathbf{E}$ and the error for the worst case $\mathbf{E}_{\max}$ once the equilibrium is achieved.

$$\mathbf{E} = \sum \mathrm{abs}(\Delta_{ij}) \qquad \mathbf{E}_{\max} = \max\{\mathrm{abs}(\mathbf{\Delta}_{ij}), \quad \forall\, i,j \text{ neighbour images}\} \tag{11}$$

Different approaches have been tried to model $f(\mathbf{\Delta}_i)$. The function that gives the minimum error and the fastest convergence was found to be proportional to the square of $\mathbf{\Delta}_i$ (shown in equation 12). The constant k is a small number required to maintain the stability of the system, since large increments lead recursively to even larger increments.

$$f(\mathbf{\Delta}_i) = k \times \begin{pmatrix} sign(\delta_x) \times \delta_x^{\,2} \\ sign(\delta_y) \times \delta_y^{\,2} \\ sign(\delta_\theta) \times \delta_\theta^{\,2} \end{pmatrix} \tag{12}$$

After the positions $\{\mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3, \ldots \mathbf{P}_N\}$ have been readjusted, new pairs of images may have become neighbours and some may no longer have common boundaries. In the case of new neighbour images appearing, their alignment transformation will be computed and the readjustment process performed again until no new neighbours arise.

At the end of the process the cumulative error is spread across all images, and therefore no single pair of images show a marked misalignment.

4 Experimental Results

The technique for solving the looping path problem has been tested with various sets of images resulting in excellent overall improvement of the mosaic. See Figures 6 and 7 for illustrative results.

The errors in the positions of the images ($\mathbf{E}$ and $\mathbf{E}_{\max}$ in equation 11) are shown in Tables 1,2 and 3, for three different mosaics. 'Text 1' is a mosaic of a text document. Mosaic 'Text 2' is the same text with a superimposed grid used to assess visually the

quality of the seams. The images for the mosaic labelled 'Lab' were obtained with a hand held video camera from a fixed location, and despite the parallax and the distortion from mapping of a spherical view on a flat image, the results are promising (Figure 8).

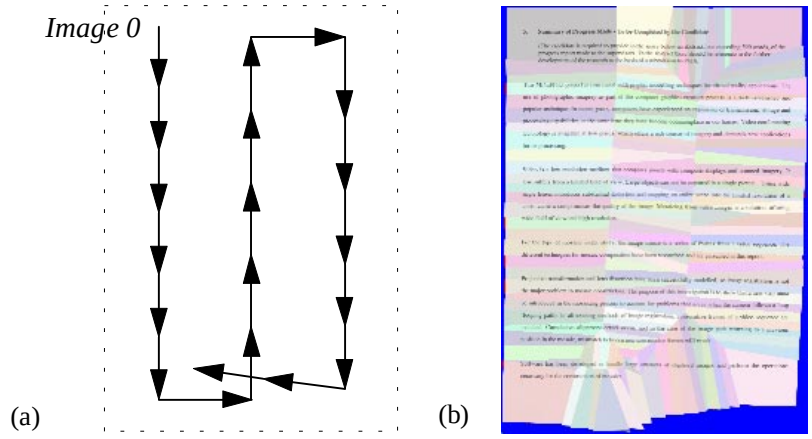


Figure 6: (a) Path followed by the camera. (b) Mosaic 'Text 1' (1663x2320 pixels), composed of 141 images (736x560 pixels). The grey levels represent the areas in the mosaic used from each particular image.

Figure 7: Details of mosaic 'Text 1'. Left, successive image alignment only, showing looping path problem. Right, corrected positions of images.

'Text 1'	$\mathbf{E}_{max}$	Between Images	Sum Error **E**	Average Error per seam
x (pixels)	0.6823	128 and 118	63.11	0.1783
y (pixels)	0.5386	105 and 57	75.63	0.2136
θ (rads)	0.002597	7 and 6	0.3728	0.001053

Table 1: Mosaic 'Text 1', 141 images, 354 pairs of neighbour images. The table shows the errors present after readjustment. In the worst case the images have been displaced by about half a pixel from their computed position.

'Text 2'	$\mathbf{E}_{max}$	Between Images	Sum Error **E**	Average Error per seam
x (pixels)	0.5627	71 and 65	28.56	0.1527
y (pixels)	0.5162	16 and 15	46.35	0.2479
θ (rads)	0.003333	71 and 65	0.1636	0.0008749

Table 2: Mosaic 'Text 2', size 1481×2139 pixels, 80 images, 187 pairs of neighbour images. Final errors.

Figure 8: Mosaic 'Lab'. Left, successive image alignment only, showing looping path problem. Right, corrected positions of images.

'Lab'	$\mathbf{E}_{max}$	Between Images	Sum Error **E**	Average Error per seam
x (pixels)	0.9934	123 and 62	71.64	0.2372
y (pixels)	1.0576	82 and 81	77.65	0.2571
θ (rads)	0.01610	123 and 65	1.186	0.003927

Table 3: Mosaic 'Lab', 130 images, 302 pairs of neighbour images. The errors are higher than in the other mosaics due to errors in the computation of image alignment caused by parallax. In addition, the field of view is about 90°, so further distortions are introduced by the mapping onto a plane.

5 Conclusions

It has been shown that a new step must be introduced in video mosaic construction to account for the looping path problem. Cumulative errors occur in successive image alignment, and in the case of the image path returning to a previous position in the mosaic, a significant mismatch between non-successive images will result. The proposed solution makes use of the alignments between all neighbour images to consistently position the images on the mosaic. Starting with the successive image alignment positions, these are modified by small increments to reduce the overall misalignment error.

Since the field of view is not a limitation when using this approach, 360° mosaics can now be produced. Current research aims at full spherical mosaics for applications in Virtual Reality.

Considering the projective transformation matrix as a representation of the position and orientation of a camera in space, an analogous method using forces can be used for the consistent alignment of all images in the mosaic using the projective model.

References

[1] Paul R. Wolf, “Elements of photogrammetry (with air photo interpretation and remote sensing)”. - New York; London : McGraw-Hill, 1974.

[2] R. Szeliski, “Video Mosaics for Virtual Environments”, IEEE Workshop on Applications of Computer Vision, pp. 44-53, 1994.

[3] S. Mann and R. W. Picard, “Virtual Bellows: constructing high quality stills from video”, IEEE International Conference on Image Proceesing, pp. 363-367, 1994.

[4] H. S. Sawhney, R. Kumar, “True Multi-Image Alignment And Its Application To Mosaicing And Lens Distortion Correction”, Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 450-456, 1997.

[5] P. J. Burt, M. Hansen, P. Anandan, “Video Mosaic Displays”, Proceedings of SPIE - International Society for Optical Engineering, Vol. 2736, pp.119-127, 1996.

[6] R. Szeliski, “Image Mosaicing for Tele-Reality Applications”, Cambridge Research Laboratory, Technical Report Series, May 1994.

[7] S. Peleg and J. Herman, “Panoramic mosaics by manifold projection”, IEEE Conference on Computer Vision and Pattern Recognition, pp. 338-343, San Juan, Puerto Rico, June 1997.

[8] Rik D. T. Janssen and A. M. Vossepoel, “Compilation Of Mosaics From Separately Scanned Line Drawings”, IEEE Workshop on Applications of Computer Vision - Proceedings, pp. 36-43, 1994.

[9] A. Zappala, A. Gee, M. Taylor, “Document Mosaicing”, Proceedings of the British Machine Vision Conference, Vol. 2, pp. 600-609, 1997.

[10] M. Hansen, P. Anandan, K. Dana, G. van der Wal, P. Burt, “Real-time Scene Stabilization and Mosaic Construction”, IEEE Workshop on Applications of Computer Vision - Proceedings, pp.54-62, 1994.

[11] F. Aurenhammer, “Voronoi diagrams: A survey of a fundamental geometric data structure”, ACM Comp. Surv., 23:345-405, 1991.

[12] M. Irani, P. Anandan, S. Hsu, “Mosaic Based Representations of Video Sequences and Their Applications”, IEEE International Conference on Computer Vision, pp. 605-611, 1995.