VIDEOPHOTOGRAMMETRIC TECHNIQUES FOR THE REPRESENTATION OF PLANAR TEXTURED SURFACES

G.M. Cortelazzo
Professor at D.E.I.
Università degli Studi di Padova - Italy -
e-mail: corte@dei.unipd.it

C. Doignon
Guest at D.E.I.
Università degli Studi di Padova - Italy -

A. Vettore
Professor at Istituto di Topografia
Università degli Studi di Padova - Italy -
e-mail: vettoan@ux1.unipd.it

Commission VI, Working group 3

ABSTRACT

This short note presents an automatic data acquisition procedure for features extraction and it describes a technique for high-resolution representation of planar textured surfaces.

1. Introduction

Telepresence and virtual visits of objects and interiors are new multidisciplinary applications where photogrammetry methods can be profitably used for various purposes. The qualifying parameter of typical photogrammetric procedures is precision, but in this text the main target is the photorealistisc rendition of inspected objects and the use of as automatic procedures as possible, hence simple and inexpensive.
This work describes a technique for high resolution representation of textured planar surfaces such as frescoed walls and large size paintings. Projective transformations are used for registering and correcting the subimages before forming a large size picture, as this approach waves from the use of a calibrated camera. Metric information can be indirectly recovered from images' particulars of known metric characteristics. Features extraction is accomplished by a semiautomatic procedure and the computation of the projective transformation coefficients is performed by a numerically efficient method.
The technique achieves subpixel accuracy and it is practically very effective since it offers very satisfactory results. The next section describes the details of the feature extraction methods. Section 3 consider the construction of the large size image from the subimages. Section 4 draws conclusion and future work directions.

2. Features extraction and matching

Automatic procedures are used in order to improve the precision of first approximation matches which are manually performed. The algorithm works over two windows W_1 and W_2 located around the two points to be matched. The automatic procedure identifies within W_2 the subarea of size W_1 with texture, light and shape characteristics closer to those of W_2 with respect to the minimum mean square error. It returns then point P_2' which is at the center of this sub-area, as a proposed match for P_1.
Other similarity criteria were tried in place of the mean square of the differences, such as the local analysis of image's activity, quantified by spatial moments of the various type, but final results does not substantially differ form the one obtained by mean square error.
An important fact is the color nature of images, for which each one them is coded as a triplet of its *R*, *G* and *B* components.
The semiautomatic matching procedure is performed separately on each of the three component images returning instance points P_2^R with minimum mean square error ε_R for the red component, P_2^G with minimum square error ε_G for the green component and P_2^B with the minimum square error ε_B for the blue component.
The match of point P_1 is then evaluated as

$$P_2' = \lambda_R \cdot P_2^R + \lambda_G \cdot P_2^G + \lambda_B \cdot P_2^B \tag{1}$$

$$\lambda_{n=R,G,B} = \frac{1/\varepsilon_{n=R,G,B}}{\sum_{j=R,G,B} \frac{1}{\varepsilon_j}} \tag{2}$$

Point P_2' is the convex combination of the points P_2^R P_2^G P_2^B weighted according to weights which are inversely proportional to errors ε_R ε_G ε_B. Clearly expression *(2)* does not hold if some of the errors is exactly zero and the numerical consistency of *(1)* and *(2)* is checked before returning the final result.
The main assumption in this part of our work is that we consider only a small change of intensity of colors between the two images I and I' due to a small motion and no brightness changes.

3. Mosaicing procedure

An exact projective transformation Θ between two sets of coplanar points, for which the homogeneous coordinates are $Q = \begin{pmatrix} sx \\ sy \\ s \end{pmatrix}$ and $Q' = \begin{pmatrix} wx' \\ wy' \\ w \end{pmatrix}$, can be represented by matrix Θ and the following relationships:

$$Q = \Theta \cdot Q' \tag{3}$$

$$\Theta = \begin{pmatrix} \theta_1 & \theta_2 & \theta_3 \\ \theta_4 & \theta_5 & \theta_6 \\ \theta_7 & \theta_8 & \theta_9 \end{pmatrix} \tag{4}$$

or:

$$x' = \frac{\theta_1 x + \theta_2 y + \theta_3}{\theta_7 x + \theta_8 y + \theta_9} \tag{5}$$

$$y' = \frac{\theta_4 x + \theta_{52} y + \theta_6}{\theta_7 x + \theta_8 y + \theta_9} \qquad (6)$$

It is important to note that Θ is defined up to a scale factor so its ninth component can be set to one. Rewriting previous equations we can provide linear ones with respect to θ_i's:

$$x' \cdot x \cdot \theta_7 + x' \cdot y \cdot \theta_8 + x' - \theta_1 \cdot x - \theta_2 \cdot y - \theta_3 = 0 \qquad (7)$$

$$y' \cdot x \cdot \theta_7 + y' \cdot y \cdot \theta_8 + y' - \theta_4 \cdot x - \theta_5 \cdot y - \theta_6 = 0 \qquad (8)$$

Then, the *LMS* solution of this linear system yields estimate of projective transformation parameters:

$$\underbrace{\begin{pmatrix} -x & -y & -1 & 0 & 0 & 0 & x \cdot x' & y \cdot x' \\ 0 & 0 & 0 & -x & -y & -1 & x \cdot y' & y \cdot y' \end{pmatrix}}_{\underline{h}} \cdot \begin{pmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \\ \theta_5 \\ \theta_6 \\ \theta_7 \\ \theta_8 \end{pmatrix} = \underbrace{\begin{pmatrix} -x' \\ -y' \end{pmatrix}}_{\underline{k}} \qquad (9)$$

With at least four pairs of corresponding points, one obtains:

$$\underline{\theta} = \begin{pmatrix} \theta_1 \\ \vdots \\ \theta_9 \end{pmatrix} \left(H^T H\right)^{-1} \cdot H^T \cdot \underline{k} + \underline{v} \qquad (10)$$

with:

$$H = \left(\underline{h}_1^T \cdots \underline{h}_{n \geq 4}^T\right)^T \quad \text{and} \quad \underline{k} = \left(\underline{k}_1^T \cdots \underline{k}_{n \geq 4}^T\right)^T \qquad (11)$$

and $\underline{v}$ an arbitrary vector, which could contain a second criteria to minimize. The main advantage of using projective transformation instead of rigid one is that the inner structure of the camera , which requires a model for calibration, is not necessary and by this way could not propagate some inaccurate intrinsic values due to the difficult problem of camera calibration.

The calculation of Θ can be performed efficiently by Heckbert's method. Coefficients are determined by solving a linear system. This yields a solution to the general Quadrilateral-to-Quadrilateral problem. Speed up is possible when considering several special cases: Square-to-Quadrilateral, Quadrilateral-to-Square and then Quadrilateral-to-Quadrilateral using the results of the last two cases. We now consider each case individually:

Case 1: Square-to-Quadrilateral

Consider the mapping of a unit square onto an arbitrary quadrilateral. The following four-point correspondences are established from the xy-plane to the $x'y'$-plane:

$$\begin{aligned} (0,0) &\rightarrow (x'_0, y'_0) \\ (1,0) &\rightarrow (x'_1, y'_1) \\ (1,1) &\rightarrow (x'_2, y'_2) \\ (0,1) &\rightarrow (x'_3, y'_3) \end{aligned} \qquad (12)$$

In this case the eight equations *(6)* and *(7)* become:

$$\begin{cases} \theta_3 = x'_0 \\ \theta_1 + \theta_3 - \theta_7 \cdot x'_1 = x'_1 \\ \theta_1 + \theta_2 + \theta_3 - \theta_7 \cdot x'_2 - \theta_8 \cdot x'_2 = x'_2 \\ \theta_2 + \theta_3 - \theta_8 \cdot x'_3 = x'_3 \\ \theta_6 = x'_4 \\ \theta_4 + \theta_6 - \theta_7 \cdot y'_1 = y'_1 \\ \theta_4 + \theta_5 + \theta_6 - \theta_7 \cdot y'_2 - \theta_8 \cdot y'_2 = y'_2 \\ \theta_5 + \theta_6 - \theta_8 \cdot y'_3 = y'_3 \end{cases} \qquad (13)$$

The solution can take another form if we define the following terms for our calculations:

$$\begin{cases} \Delta x_1 = x'_1 - x'_2 \\ \Delta x_2 = x'_2 - x'_3 \\ \Delta x_3 = x'_0 - x'_1 + x'_2 - x'_3 \\ \Delta y_1 = y'_1 - y'_2 \\ \Delta y_2 = y'_2 - y'_3 \\ \Delta y_3 = y'_0 - y'_1 + y'_2 - y'_3 \end{cases} \qquad (14)$$

Coefficients of the projective transformation are:

$$\begin{cases} \theta_7 = \begin{vmatrix} \Delta x_3 & \Delta x_2 \\ \Delta y_3 & \Delta y_2 \end{vmatrix} \Big/ \begin{vmatrix} \Delta x_1 & \Delta x_2 \\ \Delta y_1 & \Delta y_2 \end{vmatrix} \\ \theta_8 = \begin{vmatrix} \Delta x_1 & \Delta x_3 \\ \Delta y_1 & \Delta y_3 \end{vmatrix} \Big/ \begin{vmatrix} \Delta x_1 & \Delta x_2 \\ \Delta y_1 & \Delta y_2 \end{vmatrix} \\ \theta_1 = (x'_1 - x'_0) + \theta_7 \cdot x'_1 \\ \theta_2 = (x'_3 - x'_0) + \theta_8 \cdot x'_3 \\ \theta_3 = x'_0 \\ \theta_4 = (y'_1 - y'_0) + \theta_7 \cdot y'_1 \\ \theta_5 = (y'_3 - y'_0) + \theta_8 \cdot y'_3 \\ \theta_6 = y'_0 \end{cases} \qquad (15)$$

This proves to be faster than the general method. The computation may be generalized to map arbitrary rectangles onto quadrilateral by pre-multiplying by a scale and adding a translation vector.

Case 2: Quadrilateral-to-Square

The inverse of a projective mapping can be easily computed in terms of the adjoint of the transformation matrix Θ. Thus,

$\Theta^{-1} = adj(\Theta)/det(\Theta)$ where $adj(\Theta)$ is the adjoint of Θ and $det(\Theta)$ is its determinant. Since two matrices which are nonzero scalar multiples one of each other are equivalent in the homogeneous coordinate system, there is no need to divide by the determinant if it is not zero.
Consequently, the adjoint matrix can be used in place of the inverse. This proves to be a very useful result because the adjoint is always a well-behaved matrix even if the determinant is very small, e. g. when the matrix is nearly singular. Due to these results the inverse mapping is expressed below in terms of the elements of Θ :

$$adj(\Theta)=\begin{pmatrix} \theta_5\cdot\theta_9-\theta_6\cdot\theta_8 & \theta_3\cdot\theta_8-\theta_2 & \theta_2\cdot\theta_6-\theta_3\cdot\theta_5 \\ \theta_6\cdot\theta_7-\theta_4 & \theta_1-\theta_3\cdot\theta_7 & \theta_3\cdot\theta_4-\theta_1\cdot\theta_6 \\ \theta_4\cdot\theta_8-\theta_5\cdot\theta_7 & \theta_2\cdot\theta_7-\theta_1\cdot\theta_8 & \theta_1\cdot\theta_5-\theta_2\cdot\theta_4 \end{pmatrix} \tag{16}$$

Case 3: Quadrilateral-to-Quadrilateral

The results of the last two cases may be cascaded to yield a fast solution to the general Quadrilateral-to-Quadrilateral mapping problem. The general Quadrilateral-to-Quadrilateral problem is known as four corner mapping. Projective transformations offer a planar solution to this problem. We note that the order of points' labels of the quadrilateral is important. One quadrilateral must have point labels ordered and only cyclic permutations are available and this because convexity is conserved by projective transformations; we can use it like a constraint in order to reduce the number of distributions.
Once the projective transformation Θ is computed, we calculate the projections of the borders of I' by using Θ, and then by mapping $\Theta(I')$ on the plane of I' we form the borders of the resulting image.

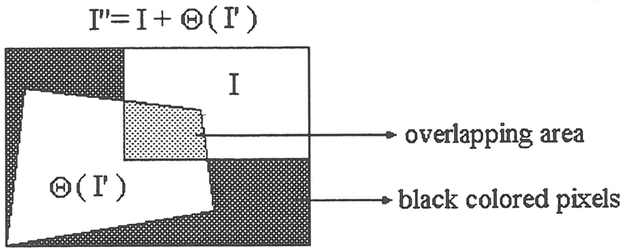

Figure 1: *Image mosaicing via projective transformation. By default each pixels in the resulting image is assumed to be black before the mosaicing step is performed.*

3.1 Pixels coloring

By considering each rectangular pixel Q'' in the result image I'', one wants to color it with an appropriate value $C(Q'')$. The method consists to back-project Q'', but only those belonging to $\Theta(I')$, onto I'. The output of the mapping is a quadrilateral pixel. The color interpolation C_I is given by the approximated formula:

$$C(Q'') \approx C_I\big(\underbrace{\Theta^{-1}(Q'')}_{Q'}\big) \tag{17}$$

The computation of C_I is obtained by considering all fractions of area A_i of output cell with color c_i crossed by the quadrilateral, so this simple calculation of $C(Q'')$ is given by:

$$C(Q'') = \frac{\sum A_i \cdot C_i}{\sum A_i} \quad \text{for each crossed cell } i \tag{18}$$

<u>Notes</u>

- *In our method each color is a 24 bits RGB value $256^0 R + 256^1 G + 256^2 B$*
- *If $\Theta^{-1}(Q) \notin I'$ the color is going to be set to black.*

4. Conclusions

This work describes a procedure for high resolution representation of planar textured images which is based on projective transformations and which does not require calibrated cameras.
W e are currently working in order to make a fully automatic matching procedure between subimages.

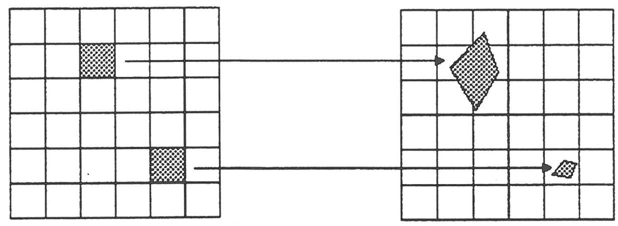

Figure 2: *Output pixel mapped via a projective transformation: left array is the output while the right one is the input.*

5. Acknowledgments

This work was partially supported by the MURST (40%) funds for the year 1997.

REFERENCES

[1] C. Doignon, G. Abba and E. Ostertag, "Recognition and localization of solid objects by a monocular vision system for robotic tasks", *Proceedings of IEEE International Conference on Intelligent Robots and System*, vol. 3, Munich, Germany, September 12 - 16, 1994.

[2] O. Faugeras, "What can be seen in three dimensions with an uncalibrated stereo rig ?", *Proceedings of the 2nd European Conference on Computer Vision*, pp. 563-578, Springer Verlag, Santa Margherita Ligure, Italy, May 1992.

[3] H. Herman and T. Kanade, "The 3D mosaic scene understanding system: incremental reconstruction of 3D scenes from complex images", *Edited by Fischler and Firscheim Reading in Computer Vision: Issues, Problems Principles and Paradigms*, pp. 471-482, 1987 (reprint).

[4] H.C. Longuet - Higgins, "A computer vision algorithm for reconstructing a scene from two projections", *Edited by*

Fischler and Firscheim Reading in Computer Vision: Issues, Problems Principles and Paradigms, pp. 61-61, 1987 (reprint).

[5] M. Richetin, M. Dhome, J.T. Laprestè and G. Rives, " Inverse perspective transform using zero - curvature contour points: application to the localization of some generalized cylinders from a single view", *IEEE Transaction on PAMI*, vol. 13, n° 2, pp. 185-192, February 1991.