V. Cappellini and A. G. Constantinides, Eds.   Amsterdam, The Netherlands: Elsevier, pp. 770–775.

[3] J. O'Rourke, "Motion detection using Hough techniques," in *Proc. Conf. Pattern Recognition and Image Processing*, Dallas, TX, 1981, p. 737.

[4] T. M. Silberberg, L. Davis, and D. Harwood, "An iterative Hough procedure for three-dimensional object recognition," *Pattern Recognition*, vol. 17, no. 6, pp. 621–629, 1984.

[5] H. Li, M. A. Lavin, and R. J. LeMaster, "Fast Hough Transform," in *Proc. 3rd Workshop Computer Vision: Representation and Control*, Bellair, MI, 1985, pp. 75–83.

[6] H. Li and M. A. Lavin, "Fast Hough Transform based on Bintree data structure," in *Proc. Conf. Computer Vision and Pattern Recognition*, Miami Beach, FL, 1986, pp. 640–642.

[7] R. Lumia, L. Shapiro, and O. Zuniga, "A new connected components algorithm for virtual memory computers," *Comput. Graphics Image Processing*, vol. 22, pp. 287–300, 1983.

[8] A. Bowyer and J. Woodwark, *A programmers geometry*.   London: Butterworth, 1983.

[9] M. Cohen and G. T. Toussaint, "On the detection of structures in noisy pictures," *Pattern Recognition*, vol. 9, pp. 95–98, 1977.

[10] T. M. Van Veen and F. C. A. Groen, "Discretization errors in the Hough Transform," *Pattern Recognition*, vol. 14, pp. 137–145, 1981.

[11] R. Lumia, "A new three-dimensional connected components algorithm," *Comput. Vision, Graphics, Image Processing*, vol. 23, pp. 207–217, 1983.

Least-Squares Fitting of Two 3-D Point Sets

K. S. ARUN, T. S. HUANG, AND S. D. BLOSTEIN

Abstract—Two point sets $\{ p_i \}$ and $\{ p_i' \}$; $i = 1, 2, \cdots, N$ are related by $p_i' = Rp_i + T + N_i$, where R is a rotation matrix, T a translation vector, and N_i a noise vector. Given $\{ p_i \}$ and $\{ p_i' \}$, we present an algorithm for finding the least-squares solution of R and T, which is based on the singular value decomposition (SVD) of a 3×3 matrix. This new algorithm is compared to two earlier algorithms with respect to computer time requirements.

Index Terms—Computer vision, least-squares, motion estimation, quaternion, singular value decomposition.

I. Introduction

In many computer vision applications, notably the estimation of motion parameters of a rigid object using 3-D point correspondences [1] and the determination of the relative attitude of a rigid object with respect to a reference [2], we encounter the following mathematical problem. We are given two 3-D point sets $\{ p_i \}$; $i = 1, 2, \cdots, N$ (here, p_i and p_i' are considered as 3×1 column matrices)

$$p_i' = Rp_i + T + N_i \tag{1}$$

where R is a 3×3 rotation matrix, T is a translation vector (3×1 column matrix), and N_i a noise vector. (We assume that the rotation is around an axis passing through the origin). We want to find R and T to minimize

$$\Sigma^2 = \sum_{i=1}^{N} \| p_i' - (Rp_i + T) \|^2. \tag{2}$$

An iterative algorithm for finding the solution was described in Huang, Blostein, and Margerum [3]; a noniterative algorithm based on quaternions in Faugeras and Hebert [4]. In this correspondence, we describe a new noniterative algorithm which involves the singular value decomposition (SVD) of a 3×3 matrix. The computer time requirements of the three algorithms are compared.

After the submission of our correspondence, it was brought to our attention that an algorithm similar to ours had been developed independently by Professor B. K. P. Horn, M.I.T., but not published.

II. Decoupling Translation and Rotation

It was shown in [3] that: If the least-squares solution to (1) is $\hat{R}$ and $\hat{T}$, then $\{ p_i' \}$ and $\{ p_i'' \triangleq \hat{R}p_i = \hat{T} \}$ have the same centroid, i.e.,

$$p' = p'' \tag{3}$$

where

$$p' \triangleq \frac{1}{N} \sum_{i=1}^{N} p_i' \tag{4}$$

$$p'' \triangleq \frac{1}{N} \sum_{i=1}^{N} p_i'' = \hat{R}p + \hat{T} \tag{5}$$

$$p \triangleq \frac{1}{N} \sum_{i=1}^{N} p_i. \tag{6}$$

Let

$$q_i \triangleq p_i - p \tag{7}$$

$$q_i' \triangleq p_i' - p'. \tag{8}$$

We have

$$\Sigma^2 = \sum_{i=1}^{N} \| q_i' - Rq_i \|^2. \tag{9}$$

Therefore, the original least-squares problems is reduced to two parts:

(i) Find $\hat{R}$ to minimize Σ^2 in (9).
(ii) Then, the translation is found by

$$\hat{T} = p' - \hat{R}p. \tag{10}$$

In the next section, we describe an algorithm for (i) which involves the SVD of a 3×3 matrix.

III. An SVD Algorithm for Finding $\hat{R}$

A. Algorithm

Step 1: From $\{ p_i \}$, $\{ p_i' \}$ calculate p, p'; and then $\{ q_i \}$, $\{ q_i' \}$.

Step 2: Calculate the 3×3 matrix

$$H \triangleq \sum_{i=1}^{N} q_i q_i'^t \tag{11}$$

where the superscript t denotes matrix transposition.

Step 3: Find the SVD of H,

$$H = U\Lambda V^t. \tag{12}$$

Step 4: Calculate

$$X = VU^t. \tag{13}$$

Step 5: Calculate, det (x), the determinant of X.
If det $(x) = +1$, then $\hat{R} = X$.
If det $(x) = -1$, the algorithm fails. (This case usually does not occur. See Sections IV and V.)

B. Derivation

Expanding the right-hand side of (9),

$$\Sigma^2 = \sum_{i=1}^{N} (q_i' - Rq_i)^t (q_i' - Rq_i)$$

$$= \sum_{i=1}^{N} (q_i'^t q_i' + q_i^t R^t R q_i - q_i'^t R q_i - q_i^t R^t q_i')$$

$$= \sum_{i=1}^{N} (q_i'^t q_i' + q_i^t q_i - 2q_i'^t R q_i).$$

Therefore, minimizing Σ^2 is equivalent to maximizing

$$F = \sum_{i=1}^{N} q_i'^t R q_i$$

$$= \text{Trace} \left(\sum_{i=1}^{N} R q_i q_i'^t \right) = \text{Trace} (RH) \qquad (14)$$

where

$$H \triangleq \sum_{i=1}^{N} q_i q_i'^t. \qquad (11)$$

Lemma: For any positive definite matrix AA^t, and any orthonormal matrix B,

$$\text{Trace} (AA^t) \geq \text{Trace} (BAA^t).$$

Proof of Lemma: Let a_i be the ith column of A. Then

$$\text{Trace} (BAA^t) = \text{Trace} (A^t BA)$$

$$= \sum_i a_i^t (Ba_i).$$

But, by the Schwarz inequality,

$$a_i^t (Ba_i) \leq \sqrt{(a_i^t a_i)(a_i^t B^t Ba_i)} = a_i^t a_i.$$

Hence, Trace $(BAA^t) \leq \Sigma_i a_i^t a_i = $ Trace (AA^t).   Q.E.D.
Let the SVD of H be:

$$H = U\Lambda V^t \qquad (12)$$

where U and V are 3×3 orthonormal matrices, and Λ is a 3×3 diagonal matrix with nonnegative elements. Now let

$$X = VU^t \quad \text{(which is orthonormal)}. \qquad (13)$$

We have

$$XH = VU^t U\Lambda V^t$$

$$= V\Lambda V^t \qquad (15)$$

which is symmetrical and positive definite. Therefore, from Lemma, for any 3×3 orthonormal matrix B,

$$\text{Trace} (XH) \geq \text{Trace} (BXH) \qquad (16)$$

Thus, among all 3×3 orthonormal matrices, X maximizes F of (14). And if det $(X) = +1$, X is a rotation, which is what we want.

However, if det $(X) = -1$, X is a reflection, which is not what we want. Fortunately, this degenerate case usually does not occur. We shall discuss the situation in some detail in the next two sections.

IV. DEGENERACY: NOISELESS CASE

Assume $N_i = 0$ in (1) for all i. Then, obviously there is a solution $\hat{R}$ (which is a rotation, i.e., det $(\hat{R}) = +1$) for which $\{q_i'\}$ and $\{\hat{R}q_i\}$ are congruent and hence $\Sigma^2 = 0$. From geometrical considerations, it is easy to see that there are three possibilities.

1) $\{q_i\}$ *are not coplanar*—Then, the rotation solution is unique. Furthermore, there is no reflection X which can make $\Sigma^2 = 0$. Therefore, the SVD algorithm will give the desired solution.

2) $\{q_i\}$ *are coplanar but not colinear*—There is a unique rotation as well as a unique reflection which will make $\Sigma^2 = 0$.

Therefore, the SVD algorithm may give either. We shall see presently that this situation can be easily resolved.

3) $\{q_i\}$ *are colinear*—There are infinitely many rotations and reflections which will make $\Sigma^2 = 0$.

Now we come back to the coplanar case. From examining the elements of the 3×3 matrix H, it can readily be shown that the points $\{q_i\}$ are coplanar, if and only if one of the three singular values of H is zero. Let the three singular values be $\lambda_1 > \lambda_2 > \lambda_3 = 0$. Then

$$H = \lambda_1 u_1 v_1' + \lambda_2 u_2 v_2' + 0 \cdot u_3 v_3' \qquad (17)$$

where u_i and v_i are columns of U and V, respectively. Note that changing the sign of u_3 or v_3 will not change H. Therefore, if $X = VU^t$ minimizes Σ^2, so does $X' = V'U^t$ where

$$V' = [v_1, v_2, -v_3]. \qquad (18)$$

If X is a reflection, then X' is a rotation, and vice versa. Thus, if the SVD algorithm gives a solution X with det $(X) = -1$, we form $X' = V'U^t$ which is the desired rotation.

We mention, in passing, that the points $\{q_i\}$ are colinear, if and only if, two of the three singular values of H are equal.

V. DEGENERACY: NOISY CASE

If either $\{q_i\}$ or $\{q_i'\}$ are coplanar, then it can readily be shown that the discussion on the coplanar case in Section IV is still valid, except of course now the minimum of Σ^2 is no longer zero. Hence, if the SVD algorithm gives a reflection $X = VU^t$, we can form the desired rotation $X' = V'U^t$. A special case of interest is when $N = 3$. Then both $\{q_i\}$ and $\{q_i'\}$ are coplanar point sets.

The situation we cannot handle is when the SVD algorithm gives a solution X with det $(X) = -1$, and none of the singular values of H is zero. This means that neither $\{q_i\}$ nor $\{q_i'\}$ are coplanar; yet there is no rotation which yields a smaller Σ^2 then the reflection x. This can happen only when the noise N_i are very large. In that case, the least-squares solution is probably useless anyway. A better approach would be to use a RANSAC-like technique (using 3 points at a time) to combat against outliers [5].

VI. SUMMARY OF ALGORITHM

Using the procedure of Section III-A, we obtain

$$X = VU^t.$$

1) If det $(X) = +1$, then X is a rotation which is the desired solution.
2) If det $(X) = -1$, then X is a reflection.
    a) one of the singular values (λ_3, say) of H is zero. Then, the desired rotation is found by forming

$$X' = V'U^t$$

where V' is obtained from V by changing the sign of the 3rd column.
    b) None of the singular values of H is zero. Then, conventional least-squares solution is probably not appropriate. We go to a RANSAC-like technique.

VII. COMPUTER TIME REQUIREMENTS

Computer simulations have been carried out on a VAX 11/780 to compare the three algorithms (SVD, quaternion, iterative) with respect to time requirements. In each simulation, a set of 3-D points $\{p_i\}$ were generated. They are randomly distributed in a cube of size $6 \times 6 \times 6$ with center at $(0, 0, 0)$. Then $\{p_i'\}$ were calculated by rotating $\{p_i\}$ by an angle of $75°$ around an axis through the origin with direction cosines $(0.6, 0.7, 0.39)$ followed by a translation of $(80, 60, 70)$, and finally by adding to each coordinate of the resulting points Gaussian random noise with mean zero and standard deviation 0.5. Then the algorithms were used to estimate $\hat{R}$ and $\hat{T}$. The CPU times used are listed in Table I. For the iterative algorithm, the numbers of iterations are given in parentheses. The programs were written in C. The IMSL subroutine package was

TABLE I
VAX 11/780 CPU TIME PER RUN IN ms

Number of Point Correspondences	Method Used		
	SVD	Quaternion	Iterative
3	54.6	26.6	126.8 (25)
7	41.6	32.4	108.2 (12)
11	37.0	41.0	105.2 (8)
16	39.4	45.6	94.2 (5)
20	40.4	45.2	135.0 (6)
30	44.2	48.3	111.0 (6)

used in finding the SVD (subroutine LSVDF) and in doing the eigen analysis (subroutine EIGRS) for the quaternion method. For the iterative method, the initial guess solution was zero in all cases.

We observe that the computer time requirements of the SVD and the quaternion algorithms are comparable, while the time for the iterative method is much longer. However, in the iterative method, the solutions were calculated to 7-digit accuracy. If we can accept 10 percent accuracy, then the number of iterations are reduced by a factor of 2 to 3. Furthermore, the rate of convergence can be increased by overrelaxation.

REFERENCES

[1] S. D. Blostein and T. S. Huang, "Estimating 3-D motion from range data," in *Proc. 1st Conf. Artificial Intelligence Applications*, Denver, CO, Dec. 1984, pp. 246–250.
[2] D. Cyganski and J. A. Orr, "Applications of tensor theory to object recognition and orientation determination," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-7, pp. 663–673, Nov. 1985.
[3] T. S. Huang, S. D. Blostein, and E. A. Margerum, "Least-squares estimation of motion parameters from 3-D point correspondences," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Miami Beach, FL, June 24–26, 1986.
[4] O. D. Faugeras and M. Hebert, "A 3-D recognition and positioning algorithm using geometrical matching between primitive surfaces," in *Proc. Int. Joint Conf. Artificial Intelligence*, Karlshrue, West Germany, Aug. 1983, pp. 996–1002.
[5] M. Fischler and Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, June 1981.

Registration of Translated and Rotated Images Using Finite Fourier Transforms

E. DE CASTRO AND C. MORANDI

Abstract—A well-known method for image registration is based on a conventional correlation between phase-only, or whitened, versions of the two images to be realigned. The method, covering rigid translational movements, is characterized by an outstanding robustness against correlated noise and disturbances, such as those encountered with nonuniform, time varying illumination. This correspondence discusses an extension of the method to cover both translational and rotational movements.

Index Terms—Digital image processing, fast Fourier transform, image registration, image sequence analysis, motion estimation.

I. INTRODUCTION

Let us consider a plane image performing rigid movements of translation and rotation within a rectangular domain C representing the observed field. Let the image be defined by a density function vanishing outside a region A_t whose position varies with time t, but is always fully contained inside C. The limitations which arise when parts of the image leave the observed field will be mentioned later on.

Let $s_0(x, y)$ represent the image at a reference time, $t = 0$, and $s_t(x, y)$ be the present image, which is but a replica of $s_0(x, y)$ translated by (x_0, y_0) and rotated by θ_0:

$$s_t(x, y) = s_0(u, v), \qquad (1)$$

where

$$x - x_0 = u \cos \theta_0 - v \sin \theta_0$$

$$y - y_0 = u \sin \theta_0 + v \cos \theta_0. \qquad (2)$$

In order to realign images s_0 and s_t, it is first necessary to determine the translation vector (x_0, y_0) and the rotation θ_0 from the information provided by $s_0(x, y)$ and $s_t(x, y)$. Several image registration algorithms are known: for excellent review papers see [1]–[6].

The present correspondence is meant as a contribution in the context of the phase correlation technique [7], [8], which was developed with reference to purely translatory displacements. According to [7], [8], let $S_0(\xi, \eta)$ and $S_t(\xi, \eta)$ be the Fourier transforms of $s_0(x, y)$ and $s_t(x, y)$. Since in the case of pure translations $s_t(x, y) = s_0(x - x_0, y - y_0)$, it follows that

$$S_t(\xi, \eta) = e^{-j2\pi(\xi x_0 + \eta y_0)} S_0(\xi, \eta). \qquad (3)$$

Therefore, by inverse transforming the ratio of the cross-power spectrum of s_t and s_0 to its magnitude, $S_t S_0^* / |S_t S_0^*| = \exp(-j2\pi(\xi x_0 + \eta y_0))$, a Dirac δ-distribution centered on (x_0, y_0) is obtained. In practice continuous transforms are replaced by finite ones, and by inverse transformation a unity pulse centered on (x_0, y_0) is obtained, so that the translation is immediately determined.

In this correspondence the principles of a generalization of the phase correlation method for the registration of rotated and translated images [9] are briefly recalled and the corresponding numerical algorithm is presented. The effectiveness of the procedure is then illustrated by means of simple experiments. Finally, the edge effects which arise when the image completely fills the observed field C are pointed out.

This correspondence is a part of research aiming at the implementation of an image stabilization system [10] for the observation of the human retina, in which the image viewed by the TV camera may be observed on a monitor free of the unavoidable spontaneous movements, which do not allow any automatic analysis of dynamic effects, such as the pulsations frequently observed in blood vessels. The program is carried out in cooperation with the Ophthalmological Clinic of the University of Bologna and the IBM Research Center in Rome.

II. THEORY

If $s_t(x, y)$ is a translated and rotated replica of $s_0(x, y)$, [see (1)], according to the Fourier Shift Theorem and the Fourier Rotation Theorem [11] their transforms are related by

$$S_t(\xi, \eta) = e^{-j2\pi(\xi x_0 + \eta y_0)} S_0(\xi \cos \theta_0 + \eta \sin \theta_0,$$
$$-\xi \sin \theta_0 + \eta \cos \theta_0) \qquad (4)$$