T U M

I N S T I T U T F Ü R I N F O R M A T I K

A Note on "Pose Estimation from Corresponding Point Data" by Haralick et al.

Carsten Steger

TUM-I1116
September 2011

T E C H N I S C H E U N I V E R S I T Ä T M Ü N C H E N

TUM-INFO-09-I1116-0/1.-FI

Druck: Institut für Informatik der
Technischen Universität München

A Note on "Pose Estimation from Corresponding Point Data" by Haralick et al.

Carsten Steger
Intelligent Autonomous Systems Group
Department of Computer Science
Technische Universität München

September 19, 2011

Abstract

In 1989, Haralick et al. proposed an algorithm to compute a rigid two-dimensional pose from corresponding point sets. This note points out an error in the algorithm that results in a significant inefficiency and proposes a correction to the algorithm.

1 Introduction

Estimating a rigid two-dimensional (2D) pose from corresponding points is a problem that must be solved in many machine vision applications, for example, when a model must be aligned to an image. The speed of an algorithm is very important in machine vision since often applications must fulfill real-time constraints. In 1989, Haralick et al. [1, Section II] proposed an algorithm to solve this task that minimizes an appropriately defined error term.[1] The derivation of the algorithm, however, contains an error that leads to a significant inefficiency in the implementation. Haralick et al. state that there are two possible solutions of the rotation angle and assert that the correct solution can be found by evaluating the error term for both possible solutions. This evaluation consumes a significant proportion of the runtime of the algorithm. This note shows that the problem actually has a unique solution, i.e., that the check for the correct solution is unnecessary. This leads to a significant increase of the performance of the algorithm by a factor of 2.2–2.5.

2 Algorithm

The problem to be solved can be stated as follows [1]: given two sets of corresponding points x_n and y_n, $n = 1, \ldots, N$, find the rigid 2D pose, given by the rotation matrix

$$R = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \tag{1}$$

[1] Essentially the same derivation of the algorithm is also contained in [2, Chapter 14.7].

and the translation vector t, that minimizes the weighted sum of residual errors:

$$\epsilon^2 = \sum_{n=1}^{N} w_n \|y_n - (Rx_n + t)\|^2 \quad . \tag{2}$$

The weights w_n are used in [1] in an algorithm that is robust to outliers. They must fulfill $w_n \geq 0$ and $\sum_{n=1}^{N} w_n = 1$. Haralick et al. show that the translation t is given by

$$t = \bar{y} - R\bar{x}, \tag{3}$$

where $\bar{x}$ and $\bar{y}$ are the weighted centroids of the two point sets:

$$\bar{x} = \sum_{n=1}^{N} w_n x_n \qquad\qquad \bar{y} = \sum_{n=1}^{N} w_n y_n \quad . \tag{4}$$

With this, the residual error (2) can be transformed to

$$\epsilon^2 = \sum_{n=1}^{N} w_n [(y_n - \bar{y})^T (y_n - \bar{y}) - 2(y_n - \bar{y})^T R (x_n - \bar{x}) + (x_n - \bar{x})^T (x_n - \bar{x})] \quad . \tag{5}$$

To minimize the residual errors, the partial derivative of ϵ^2 with respect to θ must be zero. If we denote the vectors in (5) by their components, i.e., $v = (v_1, v_2)^T$, we obtain

$$\begin{aligned} 0 = -2 \sum_{n=1}^{N} w_n [&- (y_{n1} - \bar{y}_1) \sin\theta (x_{n1} - \bar{x}_1) - (y_{n1} - \bar{y}_1) \cos\theta (x_{n2} - \bar{x}_2) \\ &+ (y_{n2} - \bar{y}_2) \cos\theta (x_{n1} - \bar{x}_1) - (y_{n2} - \bar{y}_2) \sin\theta (x_{n2} - \bar{x}_2)] \quad . \end{aligned} \tag{6}$$

By defining

$$\begin{aligned} A &= \sum_{n=1}^{N} w_n [(y_{n1} - \bar{y}_1)(x_{n1} - \bar{x}_1) + (y_{n2} - \bar{y}_2)(x_{n2} - \bar{x}_2)] \\ B &= \sum_{n=1}^{N} w_n [(y_{n1} - \bar{y}_1)(x_{n2} - \bar{x}_2) - (y_{n2} - \bar{y}_2)(x_{n1} - \bar{x}_1)] \end{aligned}$$

(6) reduces to

$$0 = A \sin\theta + B \cos\theta \quad . \tag{7}$$

Haralick et al. state that (7) has two solutions:

$$\cos\theta = \frac{-A}{\sqrt{A^2 + B^2}} \qquad\qquad \sin\theta = \frac{B}{\sqrt{A^2 + B^2}} \tag{8}$$

and

$$\cos\theta = \frac{A}{\sqrt{A^2 + B^2}} \qquad\qquad \sin\theta = \frac{-B}{\sqrt{A^2 + B^2}} \quad . \tag{9}$$

Furthermore, they state: “The correct value for θ will in general be unique and will be that θ that minimizes ϵ^2. Thus the better of the two choices can always be easily determined by simply substituting each value for θ into the original expression for ϵ^2.” This statement contains two errors. First, the correct value for θ is always unique except for the degenerate case $A = B = 0$, which only occurs if all x_n and all y_n are identical, i.e., if the set of point correspondences collapses to a single point correspondence. In this case, all rotation angles $\theta \in [0, 2\pi)$ are valid solutions, and we can set $\theta = 0$ to enforce a unique solution. Second, and more importantly, since θ is always unique, the correct solution need not be determined by substituting each value for θ into ϵ^2. Indeed, the analysis in [1] does not take into account that one of the two solutions (8) and (9) corresponds to a minimum of ϵ^2, while the other one corresponds to a maximum.

To see which solution is correct, we can examine the second partial derivative of ϵ^2 with respect to θ, which must be greater than zero for ϵ^2 to have a local minimum. Hence, we require

$$0 < A \cos\theta - B \sin\theta \ . \tag{10}$$

By substituting (8) and (9) into the right hand side of (10), we obtain

$$\frac{-A^2 - B^2}{\sqrt{A^2 + B^2}} < 0 \tag{11}$$

and

$$\frac{A^2 + B^2}{\sqrt{A^2 + B^2}} > 0 \ . \tag{12}$$

Therefore, the only correct solution is given by (9) and it is unnecessary to perform the check proposed by Haralick et al.

To determine the increase in performance that can be obtained with the proposed correction of the algorithm, the original algorithm of Haralick et al. (using (2) to determine the solution with the minimum error, which corresponds to (1) in [1]) and the improved algorithm were implemented in C and compiled on different processors (an Intel Core i5 750 and a Sun Ultra-SPARC IIIi) using different compilers (Intel Composer XE 2011 and Sun Studio 11). Experiments using different numbers of point correspondences (2–10000) showed that the original algorithm is 2.2–2.5 times slower than the improved algorithm.

3 Conclusion

This note has pointed out an error in the two-dimensional pose estimation algorithm proposed by Haralick et al. in [1], which leads to a significant inefficiency, and has proposed a correction to the algorithm.

References

[1] Robert M. Haralick, Hyonam Joo, Chung-Nan Lee, Xinhua Zhuang, Vinay G. Vaidya, and Man Bae Kim. Pose estimation from corresponding point data. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(6):1426–1446, November/December 1989.

[2] Robert M. Haralick and Linda G. Shapiro. *Computer and Robot Vision*, volume II. Addison-Wesley Publishing Company, 1993.