Asian Journal of Information Technology 6 (7): 839-841, 2007
ISSN: 1682-3915
© Medwell Journals, 2007

S3D Motion Estimation Based on 2D Motion Field Using Neural Networks

Yunhua Zhang, Yaming Wang, Sun Li and Wenqing Huang
College of Informatics and Electronics, ZheJiang Sci-Tech University,
Xia-Sha, Hangzhou, 310018, P. R. China

Abstract: This study uses neural networks to estimate three-Dimensional (3D) rigid motion parameters based on two-Dimensional (2D) motion fields. The motion fields are computed from image sequences. The neural networks update their weights by Newton-Raphson procedure for minimizing the error measures. Experimental results are presented for validating the proposed approach.

Key words: Computer vision, motion estimation, 2D motion fields, neural networks

INTRODUCTION

Motion estimation is a most important part in computer vision research area. In general, the problem of three-Dimensional (3D) rigid motion analysis can be divided into 2 sub-problems (Thompson, 1989). The one concerns the image changes which are always two-Dimensional (2D) distortion function (optical flow field) or the successive positions of distinctive features. The other is how to use optical flow or point correspondence to infer the 3D object structure and motion between object and camera. This study proposes neural networks to cope with the second sub-problem.

3D rigid motion estimation is a ill-posed problem and can be solved by adding constraints. Existing methods can be divided into two categories, i.e., based feature correspondence and based optical flow. Huang estimated motion parameters by eight feature points or more. Longuet-Higgins estimated motion parameters by computing fundamental matrix (Longue-Higgins, 1981). However, these methods have to match points firstly and decompose matrix which consumes large calculation. In order to avoid points matching, motion estimation methods based on optical flow were put forward. Albus and Hong expatiate on motion analysis and calculation confines based optical flow (Albus and Hong, 1990). Zhuang et al estimated rigid motion by optical flow and depth (Zhuang *et al*., 1988).

In recent years, neural networks have been applied to motion estimation problem. Chen *et al*. (1995) used neural networks to estimate 3D rigid motion parameters based 3D feature points. Tzovars *et al*. (2000) used neural networks with adjusting weights by Newton-Raphson procedure to estimate 3D rigid motion parameters with initial 2D motion data. Chen's method used only one point to train network, which has problems of convergence in some case. Tzovars used CAHV camera model whose calibration is harder than the normal pinhole camera model.

RIGID MOTION MODEL AND CAMERA MODEL

3D rigid motion: Let us assume a 3D point P of object is $P = (X, Y, Z)^T$ before motion. After motion, the coordinates is $P' = (X', Y', Z')^T$. The motion model is the following:

$$P' = RP + T \tag{1}$$

Where R is rotation matrix and T is translation vector or equivalently by homogeneous coordinates:

$$P'^H = MP^H \tag{2}$$

Where P^H is homogeneous coordinates $(X, Y, Z)^T$ of point P before motion, P^H is homogeneous coordinates $(X', Y', Z')^T$ of point P' after motion, M is homogeneous matrix:

$$M = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} W_1 \\ W_2 \\ W_3 \\ W_4 \end{bmatrix} = \begin{bmatrix} w_{11} & w_{12} & w_{13} & w_{14} \\ w_{21} & w_{22} & w_{23} & w_{24} \\ w_{31} & w_{32} & w_{33} & w_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{3}$$

The basic problem of 3D rigid motion estimation is to solve motion matrix M.

Camera model: Assuming intrinsic parameters of camera is:

Corresponding Author: Yunhua Zhang, College of Informatics and Electronics, ZheJiang Sci-Tech University, Xia-Sha, Hangzhou, 310018, P. R. China

$$\begin{pmatrix} f_x & 0 & u_x \\ 0 & f_y & u_y \\ 0 & 0 & 1 \end{pmatrix} \tag{4}$$

Where we ingore the distortion of camera. Let us assume the projection of a 3D point P (X, Y, Z) onto the image plane is p(x, y) while the center of camera is the origin of world coordinate system. We have:

$$s\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & u_x \\ 0 & f_y & u_y \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{5}$$

or equally

$$\begin{cases} x = f_x X / Z + u_x \\ y = f_y Y / Z + u_y. \end{cases} \tag{6}$$

RIGID MOTION ESTIMATION BASED 2D MOTION FIELD

Let us assume the points' coordinates before and after motion are $P = (X, Y, Z)^T$ and $P' = (X', Y', Z')^T$, respectively. While the projection coordinates of them onto image are $P = (x, y)^T$ and $P' = (X', Y', Z')^T$, respectively. According (1), (3) and (6), we have,

$$\begin{cases} x' = f_x \cdot \dfrac{w_{11}X + w_{12}Y + w_{13}Z + w_{14}}{w_{31}X + w_{32}Y + w_{33}Z + w_{34} + u_x} \\ y' = f_y \cdot \dfrac{w_{21}X + w_{22}Y + w_{23}Z + w_{24}}{w_{31}X + w_{32}Y + w_{33}Z + w_{34} + u_y} \end{cases} \tag{7}$$

The characteristic of rigid motion is that the relationships between points are constant. Let us assume the estimated 2D motion fields are available, which are used for the estimation of rigid 3D motion parameters by minimize the following error measures:

$$\begin{aligned} d_x(P) &= p'_x - p_x \\ d_y(P) &= p'_y - p_y \end{aligned} \tag{8}$$

Six error measures can be obtained from three selected randomly points of object by Eq. 8. According the characteristic of rigid motion, other three error measure can be

$$\begin{aligned} d_7 &= (|P_1 - P_2|^2 - |M(P_1 - P_2)|^2)^2 = (A_{12}^2 - \hat{A}_{12}^2)^2 \\ d_8 &= (|P_1 - P_3|^2 - |M(P_1 - P_3)|^2)^2 = (A_{13}^2 - \hat{A}_{13}^2)^2 \\ d_9 &= (|P_2 - P_3|^2 - |M(P_2 - P_3)|^2)^2 = (A_{23}^2 - \hat{A}_{23}^2)^2 \end{aligned} \tag{9}$$

Assuming C is the center of object, other three error measures can be

$$\begin{aligned} d_{10} &= (|P_1 - C|^2 - |M(P_1 - C)|^2)^2 = (A_{1c}^2 - \hat{A}_{1c}^2)^2 \\ d_{11} &= (|P_2 - C|^2 - |M(P_2 - C)|^2)^2 = (A_{2c}^2 - \hat{A}_{2c}^2)^2 \\ d_{12} &= (|P_3 - C|^2 - |M(P_3 - C)|^2)^2 = (A_{3c}^2 - \hat{A}_{3c}^2)^2 \end{aligned} \tag{10}$$

Then the error vector is

$$E = [d_x^1, d_y^1, d_x^2, d_y^2, d_x^3, d_y^3, d_7, d_8, d_9, d_{10}, d_{11}, d_{12}]^T \tag{11}$$

The motion parameter vector is

$$\begin{aligned} W = [&w_{11}, w_{12}, w_{13}, w_{14}, w_{21}, w_{22}, \\ &w_{23}, w_{24}, w_{31}, w_{32}, w_{33}, w_{34}]^T \end{aligned} \tag{12}$$

weights are updating by Newton-Raphson procedure

$$W^{(n+1)} = W^{(n)} - \mu J^{-1} E \tag{13}$$

As the Fig. 1 shown, Eq. 13 is calculated by neural network. This neural network is composed of three layers and minimizes the error measures by adjusting weights. The inputs of the first layer are three random points' coordinates. Motion parameter vector W is the weight between the first and the second layer. The outputs of the first layer are the inputs of the second layer. The outputs of the second layer are the estimated coordinates of points after motion. Weights between the first and the second layer are constant. In ideal case, the outputs of the third layer are the error vector E.

SIMULATION EXPERIMENT

In the simulation experiment, points are randomly generated. The range of coordinate are between (0.200) and the intrinsic parameters of camera is assumed as

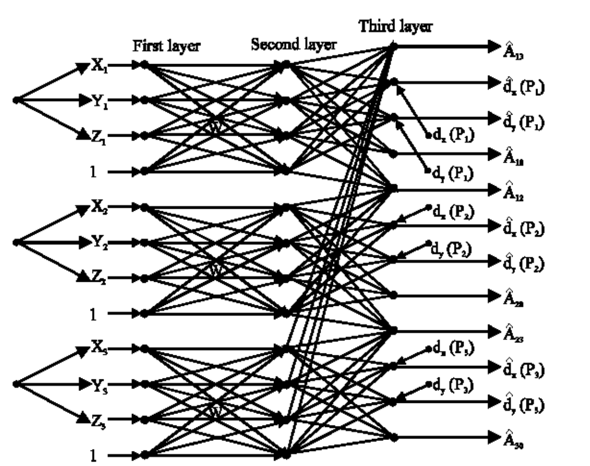

Fig. 1: Neural network for 3D motion estimation

Table 1: Results of rigid motion estimation by NN based on 2D motion field

No.	μ		Actual motion	Estimated result	Error
1	0.005	R	$\begin{pmatrix} 0.69353509 & -0.33259091 & 0.63905579 \\ 0.63905579 & 0.69353509 & -0.33259091 \\ -0.33259091 & 0.63905579 & 0.69353509 \end{pmatrix}$	$\begin{pmatrix} 0.69656301 & -0.32695553 & 0.63868570 \\ 0.63602126 & 0.69345683 & -0.33859468 \\ -0.33206156 & 0.64208692 & 0.69100356 \end{pmatrix}$	0.005836640460
		T	$(13\ \ -5\ \ 3)^T$	$(11.79440212\ \ -4.29482746\ \ 2.69716239)^T$	0.100306062904
2	0.01	R	$\begin{pmatrix} 0.69353509 & -0.33259091 & 0.63905579 \\ 0.63905579 & 0.69353509 & -0.33259091 \\ -0.33259091 & 0.63905579 & 0.69353509 \end{pmatrix}$	$\begin{pmatrix} 0.69485998 & -0.32995775 & 0.63895863 \\ 0.63769352 & 0.69346100 & -0.33532450 \\ -0.33248588 & 0.64045310 & 0.69234425 \end{pmatrix}$	0.002671681838
		T	$(13\ \ -5\ \ 3)^T$	$(12.43967533\ \ -4.70299673\ \ 2.85540509)^T$	0.045652487068
3	0.01	R	$\begin{pmatrix} 0.41492507 & -0.77503961 & 0.47660345 \\ 0.48553500 & 0.63161546 & 0.60441518 \\ -0.76947576 & -0.01937935 & 0.63838190 \end{pmatrix}$	$\begin{pmatrix} 0.41519690 & -0.77474540 & 0.47683793 \\ 0.48534754 & 0.63196665 & 0.60419792 \\ -0.76944870 & -0.01943367 & 0.63841033 \end{pmatrix}$	0.000376651551
		T	$(13\ \ -5\ \ 3)^T$	$(12.90625477\ \ -4.97099590\ \ 2.99808908)^T$	0.006888649095
4	0.01	R	$\begin{pmatrix} 0.69353509 & -0.33259091 & 0.63905579 \\ 0.63905579 & 0.69353509 & -0.33259091 \\ -0.33259091 & 0.63905579 & 0.69353509 \end{pmatrix}$	$\begin{pmatrix} 0.69371420 & -0.33132705 & 0.63952029 \\ 0.63842809 & 0.69391125 & -0.33297619 \\ -0.33348405 & 0.63926381 & 0.69292045 \end{pmatrix}$	0.001117719390
		T	$(9\ \ 1\ \ -7)^T$	$(8.83267784\ \ -1.10474038\ \ -6.82978344)^T$	.022773516445

$$\begin{pmatrix} 10 & 0 & 160 \\ 0 & 10 & 120 \\ 0 & 0 & 1 \end{pmatrix}$$

The rotation vector is generated and converted to rotation matrix using equation as follow

$$\begin{aligned} r &= r\,|r| \\ R &= \cos(|r|)\cdot I + (1-\cos(|r|))rr^T + \sin(|r|)[r]_x \end{aligned} \quad (14)$$

Where $|r|$ is the norm of rotation vector

$$r \text{ and } [r]_x = \begin{bmatrix} 0 & -r_z & r_y \\ r_z & 0 & -r_x \\ -r_y & r_x & 0 \end{bmatrix}$$

The results of 3D motion estimation based on 2D motion field are shown in the Table 1. Meanwhile the estimated results are compared with the actual ones and the error is calculated by following equation:

$$\text{error} = \frac{\|\text{estimate} - \text{true}\|}{\|\text{true}\|} \quad (15)$$

Where estimate is the estimated rotation matrix or translation vector and true is the actual parameters.

CONCLUSION

This study estimates 3D rigid motion by neural network based on 2D motion field. The neural network can be implemented by hardware. And the back propagate neural network with regularization has power calculation capability. Using Newton-Raphson procedure to adjust weights can ensure convergence. From the simulation experiment, the estimated translation vector is almost right. This experiment has demonstrated the proposed approach.

ACKNOWLEDGEMENT

This project is supported by the National Science Foundation of China under grants 60103016 and 60473038.

REFERENCES

Albus, W.B. and T.H. Hong, 1990. Motion, depth, and image flow, Proc. IEEE. Int. Conf. Robotics and Automation, pp: 1161-1170.

Chen, T., W.C. Lin and C.T. Chen, 1995. Artificial neural networks for 3-D motion analysis I: Rigid motion. IEEE. Trans. Neural Networks, 6: 1386-1393.

Huang, T.S., 1986. Handbook of Pattern Recognition and Image Processing Chapter 14: Determining three-Dimensional motion and structure from two perspective views, Academic Press.

Longue-Higgins, H.C., 1981. A computer algorithm for reconstructing a scene from two projectios, Nature, 293: 133-135.

Thompson, W.B., 1989. Introduction to the special issue on visual motion, IEEE. Trans. Pattern Anal. Machine Intell., 5: 449-450.

Tzovars, D., N. Ploskas and M. G. Strintzis, 2000. Rigid 3-D motion estimation using neural networks and initially estimated 2-D motion data, IEEE. Trans. Circuits and Sys. Video Tech., 1: 158-162.

Zhuang, X., R.M. Haralick and Y. Zhao, 1988. From depth and optical flow to rigid body motion, CVPR: Computer Soc. Conf. CVPR., pp: 393-397.