MVA '92 IAPR Workshop on Machine Vision Applications Dec. 7-9,1992, Tokyo

Target Tracking from Binocular Image Sequence Using the Autoregressive Moving Average Model

Zhen Hong and Narendra Ahuja
Beckman Institute and Dept. of ECE, Univ. of Illinois
405 N. Mathews Avenue, Urbana, IL 61801, USA
Email: hongzh, ahuja@vision.csl.uiuc.edu

ABSTRACT

A point feature based algorithm is presented in this paper which solves the problem of target tracking given a binocular image sequence. As the object motion is in general subject to a certain but unrevealed differential equation, we develope an autoregressive moving average (ARMA) model to fit the motion profile while taking into account the observation error. The prediction procedure and the matching procecures are carried out based on the ARMA model. Some experiments on real images with different target motion are provided to demostrate the efficiency and robustness of this model based approach.

1 Introduction

The objective of visual tracking is to keep the target object at the image center all the time in the complex visiual environment. Visual tracking has found its applications in areas such as aircraft and missile tracking, robot manipulation of objects, navagation, traffic monitoring, cell motion and tracking of moving parts of body in biomedicine. More recently, with the increasing interest in active vision systems, visual tracking becomes an indispensable part in tasks such as attention and gaze control, which select the processing region restrictively according to its location, motion, or depth so as to utilize the limited computational resources [1]

Most of the previous methods are composed of three main steps: target region selection and feature extraction, feature corresponcence, and depth structure estimation. The feature correspondence problem, although discussed extensively in many early works, is inherently difficult and tends to be computationally intensive. As a new trend, some researchers try to utilize various stochastic models as predictors and alternatively achieve the correspondence via prediction and verification procedure. A simplified model called "stochastic approximation" is proposed in [2] to assist in the matching as a fast predictor. A Kalman filtering based appoach is exploited in [3] for token correspondence. However, to make these models effective, the target motion is under vigorous constraints, that is, the target velocity should remain relatively unchanged throughout the samples sufficient to determine the stochastics.

This research was supported by the Defense Advanced Research Projects Agency and the National Science Foundation under grant IRI-8902728.

We present in this paper an algorithm for tracking target object in the complex visual environment from a binocular image sequence. A corner feature detector proposed in [4] is applied to each image to locate those characteristic points which yield high curvature in their local edge profiles. As the preprocessing step, we use the correlation based method to match the feature points both in space and in time, thus produce the partial 3D trajectories without prior knowledge of the target motion. As the object motion is subject to some unrevealed differential equation, we introduce the ARMA model to fit the motion trajectory and to reflect the drift in stochastics. The ARMA model is initialized by the partial 3D trajectory obtained from the preprocessing, and the model is used to generate the predicted point, and correspondence is established by comparing the projections of the predicted point and the actual image feature points. The spacial locations of the feature points are computed from their projections, the trajectories are extended and the ARMA model parameters are updated. We implemented the algorithm on the University of Illinois Active Vision System with two motorized CCD cameras coupled with the camera positioning units, the imaging parameters (tilt, pan, translation, and independent vergence) and the intrinsic parameters of the lenses (focus, aperture, and zoom) are controled by the high level routines that could be called in the visual processing programs.

We sketch the overall algorithm in a brief manner in Section 2, and describe the preprocessing procedure in Section 3. The ARMA model prediction and correspondence are explained in detail in Section 4. As the last section before the conclusion, we present the implementation of the algorithm and demonstrates some experimental results on real images.

2 The Target Tracking Algorithm

In this research, we work on a binocular image sequence taken by the dynamic camera system. We define the goal of the visual tracking as to fixate the cameras on the centroid of the visible object surface represented by its characteristic points.

The block diagram of the tracking algorithm is shown in Figure 1. The image sensors capture the left image and the right image under the controlled system configuration. The corner feature detector extracts the edge profile at

first, and then fits the local edge segments with arcs, the centers of those arcs whose curvature values exceed the threshold will become the feature points [4]. The ARMA model is applied to produce the predictions in 3D space, and the partial trajectories used to initialize the model are provided by the preprocessing step via the correlation based method. The feature point correspondences are determined by comparing the projections of the predicted 3D points and the features on the image planes obtained from the corner feature detection. The 3D locations of the feature points are constructed from their projections, and the centroid of these points will be the fixation point for the cameras. In the meantime, the 3D trajectories are extended to include the result from the current step and the ARMA model is updated based on the extended trajectory. The camera control module calculate the optimal camera movement necessary to fixate the centroid, and send the new camera configuation to the motor actuators.

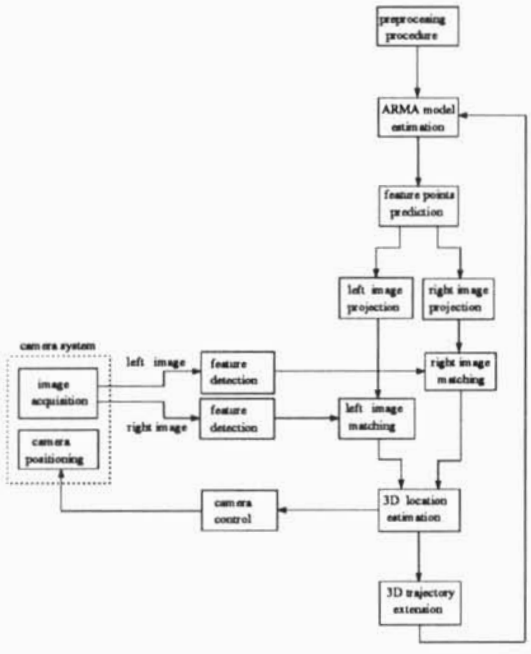


Fig. 1 Block Diagram of the Target Tracking Algorithm

Some common assumptions are implied in this work. How to find out the target object and derive the target region is a difficult but task-oriented problem, which belongs to the category of pattern recognition. We hypothize that the image region that contains the target object are given initially, the left image and the right image are registered at the beginning. Once the target is locked on, its projected regions could be updated automatically in the subsequent image frames. The target object is moving at a speed which causes only several pixels displacement during the preprocessing phase, as the correlation based method searches the matching points in the vicinity of the expected locations, although the target could be accelerated later on. Fundamentally, the target should be textured so as to provide sufficient feature points that could be extracted by the corner feature detector.

3 Preprocessing

Assume we are given the left image sequence L_1, L_2, ..., L_n, ... and the right image sequence R_1, R_2, ..., R_n, ..., we apply the corner feature detector to the first frame of the left image sequence L_1 to produce sufficient number of feature points. Let us denote the $i-th$ feature point on the left image plane at time instant k as $p^i_{L_k}$, and the $i-th$ feature point on the right image plane at time instant k as $p^i_{R_k}$. Let us consider an arbitrary trajectory that starts from $p^i_{L_1}$, for each subsequent image $L_k, k = 2, 3, \cdots, N$, in the left sequence, we extract a template centered at the feature point $p^i_{L_{k-1}}$ in the previous image frame L_{k-1}, denote as $A^i_{L_{k-1}}(s,t)$, where $s,t = 1, 2, \cdots, M$. Since we assume that the matching point for $p^i_{L_{k-1}}$ in image L_k lies in the vicinity, we try each candidate point, calculate the normalized mean correlation function of the template centred at the candidate point and the original telplate $A^i_{L_{k-1}}(s,t)$. The point which yields the maximum correlation value is chosen as the matching point of $p^i_{L_{k-1}}$, i.e. $p^i_{L_k}$. The normalized mean correlation function ρ of two templates A and B is given as

$A = A(s,t) \qquad s = 1, 2, \cdots, M;\ t = 1, 2, \cdots, M;$
$B = B(s,t) \qquad s = 1, 2, \cdots, M;\ t = 1, 2, \cdots, M;$

$$\rho = \frac{\sum_{s=1}^{M}\sum_{t=1}^{M}(A_{s,t}-\mu_A)\cdot(B_{s,t}-\mu_B)}{\sqrt{\sum_{s=1}^{M}\sum_{t=1}^{M}(A_{s,t}-\mu_A)^2}\sqrt{\sum_{s=1}^{M}\sum_{t=1}^{M}(B_{s,t}-\mu_B)^2}} \tag{3.1}$$

where μ_A and μ_B are the means of templates A and B respectively.

The correspondences of the left image features and the right image features are performed in the similar way by calculating the normalized mean correlation function, except that in the right image, the candidate points are located along the epipolar lines. The 3D locations of the feature points could be easily computed after we establish the point correspondences, and the partial 3D trajectories are obtained from the interframe correspondences. These trajectories are then used as the observations to initialize the stochastic ARMA models.

4 the ARMA Model for Feature Prediction and Correspondence

Several approaches have been developed to determine the feature point correspondences based on the assumption

that the image trajectory of a certain feature point substends some "smoothness". For instance, Sethi and Jain suggested in [5] the Greedy Exchange Algorithm to maximize the path coherence function, which tends to preserve the magnitude and the direction of the target velocity. But as pointed out in [2], this is a brute forth search method, with vigorous assumption that the target is taking approximately rectilinear motion.

We propose an autoregressive moving average model for 3D feature prediction and correspondence in this paper, and it has several advantages comparing to the existing methods. From the theory of time sequence analysis, we know that there always exists a high order ARMA model to fit an arbitrary data set with required precision. Based on this fact, the arbitrary object motion could also be approximated by an appropriate ARMA model, which reflects the motion dynamics and the observation error.

Generally, for a stochastic process $X_t, t = \cdots$,-n,$\cdots$,-1,0,1,$\cdots$, n,$\cdots$, the ARMA$(q, q-1)$ model expresses the dependence of X_t on the previous states and the previous fitting errors $a_{q-1}, a_{q-2}, \cdots, a_{t-q+1}$, which may be written in the form

$$X_t - \phi_1 X_{t-1} - \phi_2 X_{t-2} - \cdots - \phi_q X_{t-q} = a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2} - \cdots - \theta_{q-1} a_{t-q+1} \tag{4.1}$$

$$and\ a_t \sim NID(0, \sigma_a^2) \tag{4.2}$$

where $\phi_1, \phi_2, \cdots, \phi_q$ are autoregressive coefficients, $\theta_1, \theta_2, \cdots, \theta_{q-1}$ are moving average coefficients, and the fitting error at is subject to the indepedent normal distribution with variance σ_a.

Suppose $P_k(x_k, y_k, z_k)$ is the 3D point at time sample k on an arbitrary trajectory, where $k = 1, 2, \cdots, n$, we try to fit x, y, and z coordinates with three independent ARMA models, that is, to find model parameters $\phi_{x,1}$, $\phi_{x,2}$, $\cdots$, $\phi_{x,q}$, $\theta_{x,1}$, $\theta_{x,2}$, $\cdots$, $\theta_{x,q-1}$; $\phi_{y,1}$, $\phi_{y,2}$, $\cdots$, $\phi_{y,q}$,$\theta_{y,1}$, $\theta_{y,2}$, $\cdots$, $\theta_{y,q-1}$; $\phi_{z,1}$, $\phi_{z,2}$, $\cdots$, $\phi_{z,q}$, $\theta_{z,1}$, $\theta_{z,2}$, $\cdots$, $\theta_{z,q-1}$ from three observation sequences x_k,y_k, andz_k, $k = 1, 2, \cdots, n$. There are standard methods to estimate the ARMA model parameters from finite observations in the literature of time series anylysis, and we use "mothod of moments" proposed in [6]. The autocovariance function is estimated from the samples and the Yuke-Walker equations are solved. Since the stochastics are changing, we calculated the autocovariance function from the latest N samples.

For a given ARMA$(q, q-1)$ model in the form

$$X_t - \phi_1 X_{t-1} - \phi_2 X_{t-2} - \cdots - \phi_q X_{t-q} = a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2} - \cdots - \theta_{q-1} a_{t-q+1} \tag{4.3}$$

where $a_t \sim NID(0, \sigma_a^2)$, the forward prediction has the form

$$\hat{X}_{t+1} = \phi_1 X_t + \phi_2 X_{t-1} + \cdots + \phi_q X_{t-q+1} - \theta_1 a_t - \theta_2 a_{t-1} - \cdots - \theta_{q-1} a_{t-q+2} \tag{4.4}$$

and the prediction error $e_t = \hat{X}_{t+1} - X_{t+1}$ is subject to $NID(0, \sigma_a^2)$.

As a summary, the ARMA model based prediction and feature points matching take the following steps:

1.determine the model parameters for x, y, and z coordinates of each point that belongs to the same parital trajectory using the method of moments.

2. detect the feature points in the current left image and right image.

3. make a forward prediction based on the ARMA model derived in step 1 or in step 6.

4. project each predicted point onto the left image and choose the nearest feature point as its correspondence, same procedure for the right image.

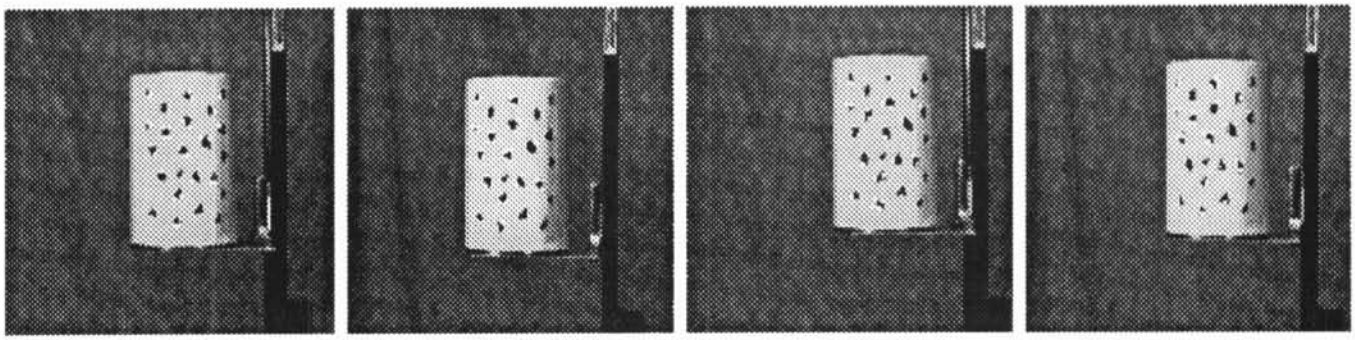
5. derive the 3D location of the feature point since we have obtained its projections, and extende the 3D trajectory related to this point.

6. repeat step 2 with the updated trajectory.

5 Experimental Results

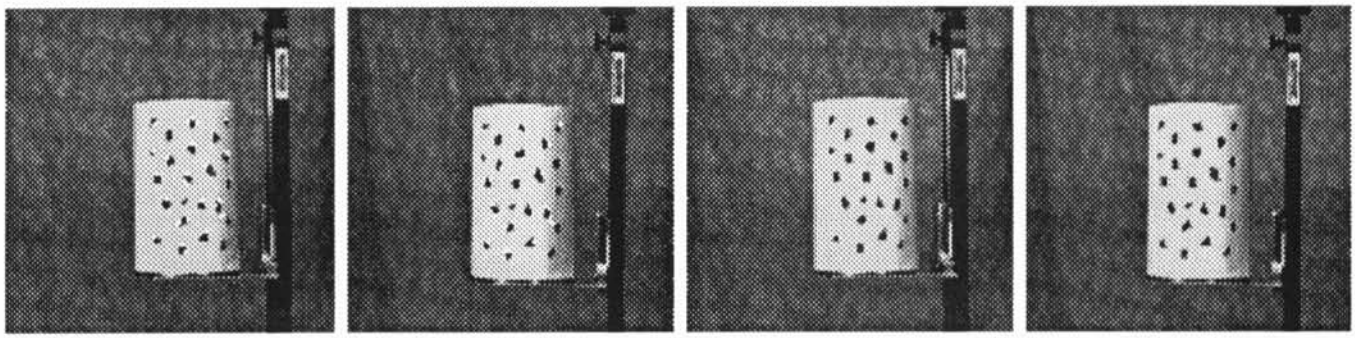
The tracking algorithm detailed in the previous sections was implemented on the University of Illinois Vision system [7]. The dynamic camera system which consists of a pair of high resolution monochrome CCD cameras, positioners, and lens controllers is capable of changing its tilt, pan, vergence, translation settings and controlling its lens parameters such as aperture, focus, zoom from the host workstation.

Initially, the cameras were made to fixate an cylindrical object resting on a sliding rail and against a textured background. The object was moving at an approximately constant speed while the stereo image sequence was being taken. Figure 2 shows four images of the sequence while the algorithm was at the model construction phase, and no camera movement was made at that stage.. Figure 3 shows the image of the sequence while the algorithm is in prediction and tracking phase, and the object is being centered after each cycle.



(a). The first left image. (b). The first right image. (c). The seventh left image. (d). The seventh right image.

Fig. 2 Figures (a) and (b) show the first image pair and the correspondences of feature points; Figures (c) and (d) show the seventh image pair and the correspondences of feature points.



(a). The tenth left image. (b). The tenth right image. (c). The fifteenth left image. (d). The fifteenth right image.

Fig. 3 Figures (a) and (b) show the tenth image pair and the correspondences of feature points; Figures (c) and (d) show the fifteenth image pair.

6 Summary

In this paper, we describe a ARMA model based tracking algorithm which adaptively utilizes the stochastic model for 3D feature points prediction, and the results are combined with the actual observation to obtain the updated estimates.

REFERENCES

[1] M. Swain and M. Stricker, "Promising directions in active vision," technical report cs 91–27, University of Chicago, Nov. 1991.

[2] K. W. M. J. Fletcher and R. J. Mitchell, "The application of a hybrid tracking algorithm to motion analysis," *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 84–89, Jan. 1991.

[3] R. Deriche and O. Faugeras, "Tracking line segments," *Proc. of Second ECCV*, pp. 259–268, 1990.

[4] N. A. J. Weng and T. S. Huang, "Two-view matching," *Proc. of Second Intl. Conf. on Computer Vision*, pp. 64–73, 1988.

[5] I. K. Sethi and R. Jain, "Finding trajectories of feature points in a monocular image sequence," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 9, pp. 56–73, Jan. 1992.

[6] G. E. P. Box and G. M. Jenkins, *Time Series Analysis: Forecasting and Control*. Holden-Day, Oakland, California, 1976.

[7] A. L. Abbott, *Dynamic Integration of Depth Cues for Surface Reconstruction from Stereo Images*. Ph.d. thesis, Dept. of Electrical and Computer Engineering, University of Illinois, Urbana, IL, 1990.

[8] A. L. Abbott and N. Ahuja, "Surface reconstruction by dynamic integration of focus, camera vergence, and stereo," *Proc. of Second Intl. Conf. on Computer Vision*, pp. 532–543, Dec. 1988.

[9] Y. Aloimonos and D. P. Tsakiris, "Tracking in a complex visual environment," *Proc. of Second ECCV*, pp. 249–258, 1990.