A Multi-Target Tracking System for a Binocular Image Sequence

Sheng-Fuu Lin and Yu-Min Chen
Department of Electrical and Control Engineering
National Chiao Tung University, 1001 Ta Hsueh Rd.
Hsinchu, Taiwan 30010, R.O.C.

Abstract —**In many occasions where safety and security are critical, such as military affairs and scene monitoring, target-tracking plays a vital role. First, in this paper the position of the moving target is measured, and then in the subject of tracking multi-target, the trajectory of every target in an image sequence is calculated immediately by using path coherence function. Because of the occlusion or the other factors, the targets may not be detected in the tracking process. In this paper, the next position of every target can be predicted by employing the α-β filter combined with the fuzzy theory and the tracking system will be processed. Finally, the proposed method based on the fuzzy theory can be applied to describe its motion trajectory. Moreover, the simulate results show that the tracking system has a good performance.**

I. Introduction

In the recent years, tracking moving objects using an image sequence has been very popular. It can be used for capturing and recognizing moving targets as well as for analyzing object motions, so as to be applied to various applications such as weapon systems, transportation systems, security systems and factory automation, etc. Three main processes should be introduced in order to implement target tracking system. The first part is a preprocessing which is used to extract the feature points of the target [3, 4, 6] and estimates the 3-D coordinates information of feature points [2, 3, 6, 15]. The second part is to track the trajectory of every target simultaneously [5, 8, 9, 10, 11, 12, 13, 14]. The third part is to predict the next position information of each target [1].

Most researchers interested in the topic of tracking multi-target consider a motion sequence of multiple objects by using a static camera, and the problem is restricted to 2-D projection of real 3-D motion or the motion state of the object on a certain plane. So, the motion trajectories are limited to 2-D plane. When the targets move in 3-D space, the number of the intersections of the trajectories which is projected from 3-D to 2-D space may become large in some case. Therefore, the tracking error will be increased. In this paper, the dual-CCD cameras will be applied to obtain more information of 3-D space to improve the drawback described previously. Besides, if some feature points are lost, it will result in tracking failure because of the insufficient feature points. In order to deal with this problem, a fuzzy predictor will include the tracking system.

On the other hand, in the process of capturing images, some errors between the measured and the real trajectories will be produced because of the hardware and noise disturbance. In order to improve the situation, a fuzzy estimator will be introduced in this paper to decrease the error such that the estimated trajectories will be more similar to the real ones.

II. Multi-Target Tracking system

In this section, there are two major parts in the system architecture. In the part of describing the motion trajectory, it is essentially composed of three portions; image preprocessing, 3-D coordinates estimation and a fuzzy estimator. In the part of tracking multi-target system, it is essentially composed of two portions; the tracking multi-target method and a fuzzy predictor.

A. Preprocessing

In order to obtain the target from the background, the global image thresholding [3] method is applied. The single thresholding T is derived to partition the image histogram. After using the image thresholding method, some isolated noise still remained in the image. To cancel these noise in the frame, the median filter [3] is adopted. The median filter not only can erase these noise, but also it can smooth the edge of the target. After using image thresholding and median filtering, the target can be extracted from the background.

In order to derive the feature point of the target, the region growing technique [3] is used to obtain the center of gravity of the target. First, the pixel of the top-left boundary of the target regarded as the seed point. Then, the pixel aggregation approach starts with the seed point and, from this grows regions, append to the seed point so that neighboring pixels with same gray level can be employed. These coordinates of region growing regions will be recorded. Next, the center of gravity of the target is estimated by

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{1}$$

$$\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i \tag{2}$$

B. 3-D Coordinates Estimation

For any camera model, the perspective projection matrix can be established by the perspective projection transformation. The coordinate system *(x,y,z)* for the 3-D space and *(u,v)* for the 2-D space in the image can be written linearly as

$$\begin{bmatrix} \hat{u} \\ \hat{v} \\ s \end{bmatrix} = \begin{bmatrix} q_{11} & q_{12} & q_{13} & q_{14} \\ q_{21} & q_{22} & q_{23} & q_{24} \\ q_{31} & q_{32} & q_{33} & q_{34} \end{bmatrix}_{3x4} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = P_{3x4} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (3)$$

where $u = \hat{u}/s, v = \hat{v}/s$. A self-calibration technique is used to decide every element of the matrix *P*. A calibrated pattern is used, as shown in Fig. 1. The coordinates of the corners on the calibrated pattern are known precisely with respect to a real 3-D coordinates system and the coordinates of the corners projected in the image plane are also derived by using image processing techniques. Therefore, there are 32 pairs of mapping points: (u_i, v_i) and (x_i, y_i, z_i) , *(i=1, 2, ...,32,)*. Substituting the mapping points into (3) gives

$$\hat{u} - s_i u_i = 0 \quad i = 1,2,..., 32, \quad (4)$$

$$\hat{v} - s_i v_i = 0 \quad i = 1,2,..., 32, \quad (5)$$

The 64 equations are replace with the matrix formation below

$$A_{64x12} q_{12x1} = 0, \quad (6)$$

where

$q = [q_{11}, q_{12}, q_{13}, q_{14}, q_{21}, q_{22}, q_{23}, q_{24}, q_{31}, q_{32}, q_{33}, q_{34,}]$. The solution of q is the unit eigenvector of matrix $\mathbf{A}^T\mathbf{A}$ associated to the *smallest* eigenvalue. By using the above method, the projection matrices $[P]_{3x4}$ and $[P']_{3x4}$ of binocular CCD cameras can be computed respectively. The 3-D coordinates of the feature points can be derived by using the projection matrices. Given a pair of feature point in correspondence: *(u,v)* and *(u',v')* in the image planes of binocular CCD cameras. Let $X = [\hat{x}, \hat{y}, \hat{z}, t]^T$ be the corresponding 3-D projective point. Following the camera model, the two equations can be derived.

$$s = [u, v, 1]^T = P[\hat{x}, \hat{y}, \hat{z}, t]^T, \quad (7)$$

$$s' = [u', v', 1]^T = P[\hat{x}, \hat{y}, \hat{z}, t]^T, \quad (8)$$

Eliminating *s* and *s*' from (11) and (12) yield the following equation:

$$A_{4x4} X = 0 \text{ with } A = \begin{bmatrix} p_1 - up_3 \\ p_2 - up_3 \\ p'_1 - u' p'_3 \\ p'_1 - u' p'_3 \end{bmatrix} \quad (9)$$

Denote p_i be the vector corresponding to the *i*th row of *P*, and p'_i be the vector corresponding to the *i*th row of *P'*. The solution to (14) is well known to be the eigenvector of matrix $A^T A$ that has the smallest eigenvalue. Then, the real 3-D coordinates *(x,y,z)* can be written as

$$x = \hat{x}/t, y = \hat{y}/t, z = \hat{z}/t. \quad (10)$$

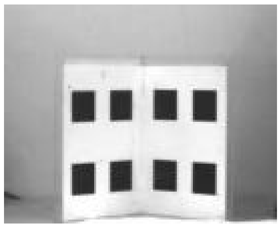


Fig. 1: Calibrated pattern

C. A Fuzzy Estimator

Hardware limitation and noise disturbance result in the error between previous 3-D coordinates estimation and the real ones. To describe the trajectory which is near real one, a fuzzy estimator is proposed. For the case of spherical and rectangular coordinates these transformations are

$$x = R\cos\phi\sin\theta, \quad y = R\sin\phi, \quad z = R\cos\phi\cos\theta, \quad (11)$$

for spherical-to-rectangular coordiantes and

$$R = (x^2 + y^2 + z^2)^{1/2}, \quad \theta = \tan^{-1}(x/z), \quad \phi = \tan^{-1}\left[y \big/ (x^2 + z^2)^{1/2} \right], \quad (12)$$

for rectangular-to-spherical coordinates. Then, the measured coordinate of every target has to make above transformation from the rectangular coordinates *(x,y,z)* to the polar coordinates (R, θ, ϕ), as shown in Fig. 2. Based on common sense knowledge about moving targets with reasonable changes in velocity, it is assumed that moving targets can not change dramatically in any directions, i.e., the difference $\theta_i(k)$ of the angles $\theta_{i1}(k)$ and, $\theta_{i1}(k-1)$, and the difference $\phi_i(k)$ of the angles $\phi_{i1}(k)$ and $\phi_{i1}(k-1)$, *i=1, 2, ..., m.*

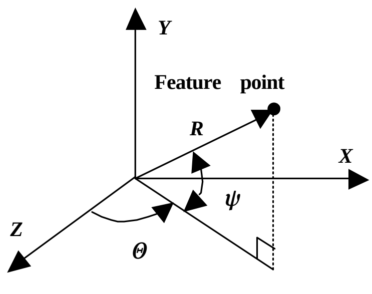


Fig. 2: Spherical coordinates typically used for three-dimensional measurements.

The smoothness assumption described above formulates the fuzzy If-then rules for the fuzzy controller in the fuzzy estimator, for example: If the angle difference, $\theta_i(k)$ is positive small, then the angle adjustment, $\theta_{ia}(k)$, is negative small. By using a fuzzy estimator, the motion trajectory is described smoother and the measurement errors can also be reduced.

D. Tracking Trajectories of Feature Points

By using the three adjacent frames, the value of the cost function can be calculated. Then, the tracking system will be executed. The following function is used:

$$d_i^k = \psi(X_{ik-1}, X_{ik}, X_{ik+1}) = w_1\left(1 - \frac{\overrightarrow{X_{ik-1}X_{ik}} \cdot \overrightarrow{X_{ik}X_{ik+1}}}{\left\|\overrightarrow{X_{ik-1}X_{ik}}\right\| \cdot \left\|\overrightarrow{X_{ik}X_{ik+1}}\right\|}\right) + w_2\left(1 - \frac{2\left[\left\|\overrightarrow{X_{ik-1}X_{ik}}\right\| \cdot \left\|\overrightarrow{X_{ik}X_{ik+1}}\right\|\right]^{1/2}}{\left\|\overrightarrow{X_{ik-1}X_{ik}}\right\| + \left\|\overrightarrow{X_{ik}X_{ik+1}}\right\|}\right), \qquad (13)$$

where w_1 and w_2 are weights. This algorithm will extend trajectories up to the *(k+1)*st frame, assuming the trajectories up to the *k*th frame have already been obtained. The points of the *(k+1)*st frame are assigned to be the established trajectories using the nearest neighbors. These tentative trajectories are then iteratively refined using the following method. The value of the criterion *D* for the tentative trajectories for the *(k+1)* st frame is

$$D = \sum_{p=1}^{m} D_p = \sum_{p=1}^{m} \sum_{q=2}^{k} d_p^q, \qquad (14)$$

Now let us consider two trajectories, the *i*th and the *j*th, and rewrite the above equation as

$$\begin{aligned} D &= \sum_{p=1;\, p\neq i,j}^{m} D_p + \sum_{q=2}^{k} d_i^q + \sum_{q=2}^{k} d_j^q \\ &= \sum_{p=1;\, p\neq i,j}^{m} D_p + \sum_{q=2}^{k-1} d_i^q + \sum_{q=2}^{k-1} d_j^q + d_i^k + d_j^k, \end{aligned} \qquad (15)$$

Suppose now the points from the *(k+1)*st frame on the *i*th and *j*th trajectories are exchanged. Clearly, such an exchange would not affect the first three terms in the above equation. Now let $\boldsymbol{d_i^k}$ and $\boldsymbol{d_j^k}$ be the new path coherence measures for the *i*th and *j*th trajectories after the exchange. Then, for the above exchange to be profitable, let us have

$$g_{ij}^k = d_i^k + d_j^k - (\boldsymbol{d_i^k} + \boldsymbol{d_j^k}), \qquad (16)$$

should be positive.

If the feature points is tracked in the previous *k* frames, the next step is to determine the feature points in the *(k+1)*st frame belongs to which trajectories. The detailed method is as following:

Step 1: Using the nearest neighbors method, initialize trajectories, such that the sum of distance of the feature points in the *k*th frame and in the *(k+1)*th frame are the shortest. The decision is a point in any frame is assigned to only one trajectory.

Step 2: The gain g_{ij}^k for *i=1* to *m-1* and *j=i+1* to *m* is computed in the *(k-1)*th, *k*th,and *(k+1)*th frames by using the path coherence function. The weight values w_1, w_2 are experimentally selected to be 0.1 and 0.9. Next, the *i-j* pair providing the maximum gain is picked. Then the feature points of the T_i and T_j trajectories in the *(k+1)*th frame are exchanged.

Step 3: Finally, the second step is repeated until the below condition is satisfied. Then the final trajectories derived are the smoothest in the theorem.

E. A Fuzzy Predictor

After tracking feature points is accomplished, the next task is to predict the next positions of the targets. Here, the **α - β** filter is adopted. The **α - β** filter is probably the most extensively applied fixed-coefficient filter. This filter can be defined by the following equations [1]:

$$x_s(k) = x_p(k) + \alpha\left[x_o(k) - x_p(k)\right], \qquad (17)$$

$$v_{sx}(k) = v_{sx}(k-1) + \beta / qT\left[x_o(k) - x_p(k)\right], \qquad (18)$$

$$x_p(k+1) = x_s(k) + Tv_{sx}(k), \qquad (19)$$

The fuzzy rules are expressed in terms of two input variables and two output variables. The input variables, defined in terms of the prediction error *e(k)* and change of error **Δ** *e(k)*, are given as

$$e(k) = \frac{\sqrt{e_x^2(k) + e_y^2(k) + e_z^2(k)}}{\sqrt{3}} \qquad (20)$$

$$\Delta e(k) = \frac{\sqrt{\Delta e_x^2(k) + \Delta e_y^2(k) + \Delta e_z^2(k)}}{\sqrt{3}} \qquad (21)$$

The input membership function including two input variables *e(k)* and **Δ** *e(k)* is seen in Fig. 3, and the output membership function including two output variables are shown in Fig. 4. The set of fuzzy rules is shown in TABLE 1.

By employing a fuzzy predictor, the next position of every target can be predicted. Even though the feature point may be lost in the next frame, a fuzzy predictor can still estimate the possible position of the feature point.

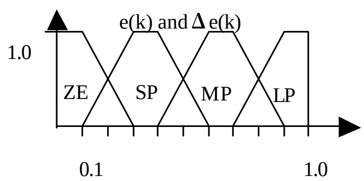


Fig. 3: Member function for input

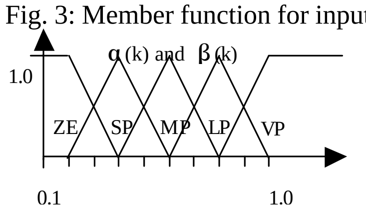


Fig. 4: Member function for output

e(k)

	α(k)	ZE	SP	MP	LP
Δ e(k)	ZE	VP	SP	VP	VP
	SP	LP	LP	VP	VP

MP	VP	VP	MP	MP
LP	VP	ZE	MP	VP

		e(k)			
	β(k)	ZE	SP	MP	LP
Δe(k)	ZE	VP	SP	VP	VP
	SP	LP	LP	VP	VP
	MP	VP	VP	MP	MP
	LP	VP	ZE	MP	VP

TABLE 1. Fuzzy rule.

III. Simulations and Results

The performance of multi-target tracking system is evaluated. The simulation is divided into two parts. First, the fuzzy estimator is tested and verified. Second, the multi-targets tracking system is simulated and tracked. The simulate results and analyses are discussed.

A. The Results of Estimating the Motion Trajectory

The simulate environment consists of two CCD cameras and the mechanism, as shown in Fig. 5. By using the mechanism instrument, a trajectory of a moving target in the 3-D space can be generated. When the target moves in the 3-D space, the real 3-D coordinates of the target are recorded immediately. Therefore, the real motion trajectory can be plots. In the mean while, the measured coordinates of the target are computed by using the binocular vision combined with self-calibration technique. Then the measurement motion trajectory can be described. Moreover, the fuzzy estimator can estimate the position coordinates by using the measured coordinates simultaneously. Finally, the whole estimated trajectory can be derived.

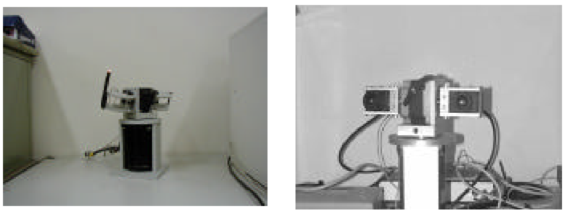


Fig. 5: Illustration of simulation setup

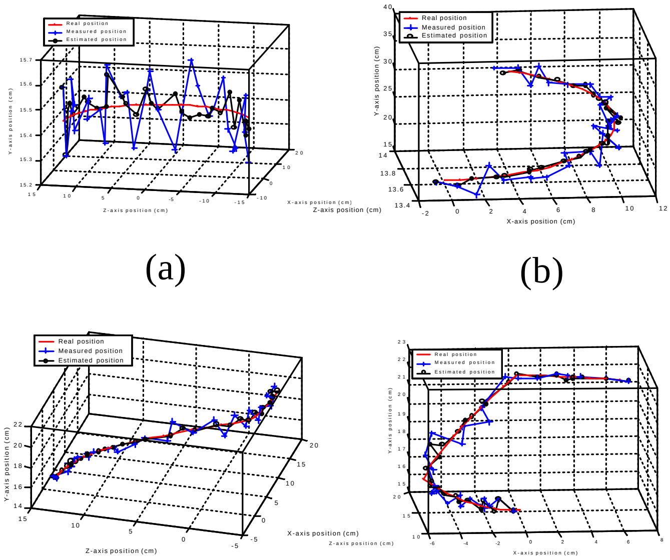
In order to evaluate the performance of the estimation in 3-D space and the fuzzy estimator, four trajectories will be designed and every the same trajectory is tested 100 times. Moreover, it is assumed that these trajectories are smooth. Simulate results about four different trajectories are shown in Fig. 6. The average error $\hat{E}_{ave}$ is defined as follows:

$$\hat{E}_{ave} = \frac{1}{100}\sum_{j=1}^{100}\left(\frac{1}{n}\sum_{i=1}^{n}\left((x_{ij}-x_{rij})^2+(y_{ij}-y_{rij})^2 + (z_{ij}-z_{rij})^2\right)^{1/2}\right),$$

where n is the number of estimated points, (x_{ij}, y_{ij}, z_{ij}) is the estimated coordinates or measured ones, and $(x_{rij}, y_{rij}, z_{rij})$ is the real coordinates. The statistical data of the different paths are shown in TABLE 2. According to simulate results, most measured error can be decreased effectively and the estimated trajectories become smoother than the measured ones. When the target moves abruptly, the estimated result is not perfect because the fuzzy rules are designed by the smoothness assumption. If the sampling rate is increased, the trajectory of the moving target will become smoother and the estimated result will be improved.

B. Results of Tracking Multi-Target System

To evaluate the performance and efficacy of multi-target tracking system, the simulations are established. First, a virtual 3-D space is created and



(a) (b)

(c) (d)

motion trajectories of the targets are plotted in the virtual 3-D

Fig. 6: Four designed trajectories. (a) First. (b) Second. (c) Third. (d) Fourth.

Path	Measured Error	Estimated Error
First	0.9137	0.6932
Second	0.9035	0.7088
Third	0.9732	0.7523
Fourth	1.0116	0.7642

TABLE 2. The Average Error.

space. The feature points are selected in every motion path and arranged in accordance with early or late sequence. In order to increase reliability about simulation, the chosen coordinates of feature points will include some random noise and the range of random noise is ± 1 unit. These coordinates are regareded as measured ones.

In the simulations, the motion states of the targets will be defined as follows:

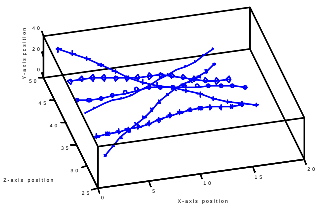
intersection : occurred if the targets move in 3-D space, the distance between two feature points on different trajectories at a certain point of time is smaller than a prespecified limit of distance.
complexity : complex occurred if the number of intersections are larger than half of the number of the targets in the same frame, otherwise, it is called non-complex.

Then, the simulations of *complex* and *non-complex* will be executed. Several group of targets will be used to move at different trajectories. The number of targets in each group are 3, 4, 5, 6, 7, 8 and 9, respectively. They are tested for 250 times in the cases of *complex* and *non-complex*. In the process of tracking feature points, if any feature point has a wrong correspondence, the case will be defined as **tracking failure.** Otherwise, the result is **tracking success**.

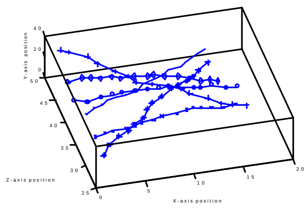
In the case of *non-complex*, The tested result is listed in TABLE 3. The part simulation result is

Number	Tested Times	Tracking Success	Tracking Failure	Success Rate(%)
3	250	247	3	98.8
4	250	242	8	96.8
5	250	236	14	94.4
6	250	229	21	91.6
7	250	224	26	89.6
8	250	221	29	88.4
9	250	213	37	85.2

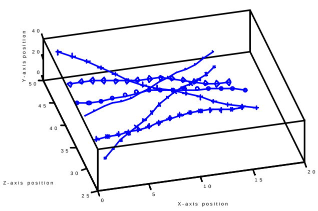
TABLE 3. The result of simulation for non-complex



(a)



(b)



(c)

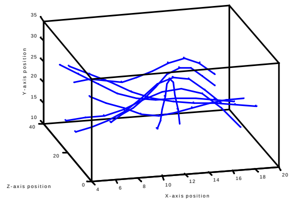
Fig. 7: The non-complex example of tracking six targets. (a) The real trajectories. (b) The measured trajectories. (c) The tracked trajectories.

Number	Tested Times	Tracking Success	Tracking Failure	Success Rate(%)
3	250	244	6	97.6
4	250	235	15	94
5	250	221	29	88.4
6	250	203	47	81.2
7	250	186	64	74.4
8	250	168	82	67.2
9	250	153	97	61.2

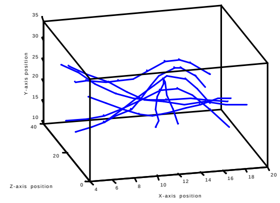
TABLE 4. The result of simulation for complex

presented in Fig. 7. .In the case of *complex*, the tested result is listed in TABLE 4, and the part simulation result is illustrated in Fig. 8. However, when there are a lot of feature points which intersect densely at the same time, it is difficult to avoid misclassified trajectories. Because the trajectory-tracking method is an local optimization problem, stepwise optimization procedures guarantee only local optimization.

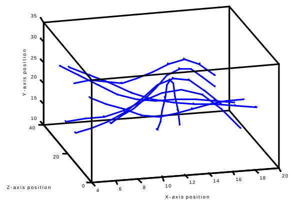
In order to evaluate the performance of the fuzzy predictor, some feature points will be hidden. When some feature points are lost, it will result in tracking failure because of the insufficient feature points. Therefore, the fuzzy predictor will predict the available positions associated to the hidden feature points such that the tracking algorithm will be continued. The result is shown in Fig. 9.



(a)

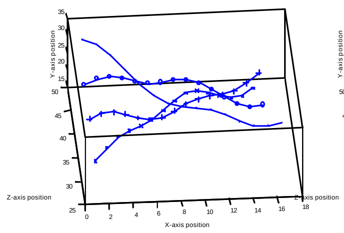


(b)

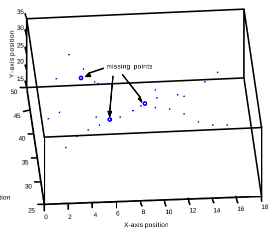


(c)

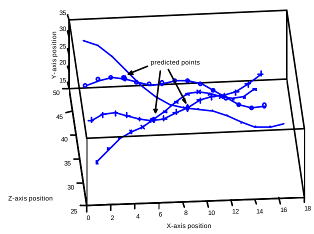
Fig. 8: The complex example of tracking six targets. (a) The real trajectories. (b) The measured trajectories. (c) The tracked trajectories.



(a)



(b)



(c)

Fig, 9: The example of tracking four targets. (missing feature points) (a) The correct trajectories. (b) All feature points of every trajectory. (c) The tracked trajectories.

IV. Conclusions

In this paper, an approach for multi-target tracking system is proposed. To evaluate the system, first, image processing techniques are employed. The method of binocular vision combined with self-calibration technique is further applied to compute 3-D coordinates of the target. Due to the measured error, a fuzzy estimator will be applied to describe the motion trajectory of the target. In this simulate results, the error diminishes effectively and the estimated trajectory can become smoother than measured one. Next, several different motion trajectories and feature points are used to evaluate tracking system. In general, the trajectories are tracked correctly. When a large of feature points intersect densely simultaneously, it is difficult to avoid misclassified trajectories. When some feature points are missing, the predicted points will be employed so as to sustain the same number of feature points in every frame.

In this simulation, the fuzzy system is used to estimate the trajectory and predict the position. Nevertheless, fuzzy modelling has some interdependent problems such as fuzzy partition of data space, identification of membership functions and consequence models, etc. In this simulation, these problems are solved by trials and errors, and the parameters are manually adjusted by the operator. How to develop an adaptive fuzzy rules which has the ability of self-organizing for dealing with any kind of possible problems will be the next challenge in the future.

In most researches, the cost function will be calculated by the three adjacent frames. Therefore, the cost function is composed of two factors, speed and direction without regard to acceleration. In the future, more than three frames may be used in the more advanced tracking systems to provide the cost function including the factor of acceleration. Such new systems can be applied in several motion states of multi-target and produce better performance than the current ones.

Reference

[1] S. S. Blackman, Multiple Target Tracking with Rader Applications. Norwood, MA: Artech House, 1986.

[2] K. S. Fu, R. C. Gonzalez and C.S.G. Lee, Robotics: Control, Sensing, Vision, and Intelligence. New York: McGraw-Hill, 1987.

[3] R. C. Gonzalez and R. E. Woods, Digital Image Processing. Reading, MA: Addision Wesley, 1993.

[4] T. I. Hentea, "Algorithm for automatic threshold determination for image segmentation," Canadian Conf. On Electrical and Computer Engineering, Vancouver, Canada, vol. 1, pp. 535-538, 1993.

[5] V. S. Hwang, "Tracking feature points in time-varying images using an opportunistic selection approach," Pattern Recognition, vol. 22, no. 3, pp.247-256, 1989.

[6] A. K. Jain, Fundamentals of Digital Image Processing. Englewood Cilhs, N.J: Prentice-Hall, 1989.

[7] C. T. Lin and C. S. Lee, Neural Fuzzy System. New Work: Prentice-Hall, 1996.

[8] R. Mehrotra, "Establishing motion-based feature point correspondence," Pattern Recognition, vol. 31, no. 1, pp. 23-30, 1998.

[9] D. Murray and A. Basu, "Motion tracking with an active camera, " IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 16, no. 5, pp. 449-459, 1994.

[10] M. A. Rahgozar and J. P. Allebach, "Motion estimation based on time-sequentially sampled imagery, " IEEE Trans. Image Processing, vol. 4, no. 1, pp. 48-65, 1995.

[11] K. Rangarajan and M. Shah, "Establishing motion correspondence, " Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 56-73, Lahaina, Maui, Hawaii, 1991.

[12] V. Salari and I. K. Sethi, "Feature point correspondence in the presence of occlusion, " IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 12, no. 1, pp. 87-91, 1990.

[13] I. K. Sethi and R. Jain, "Finding trajectories of feature points in a monocular image sequence, " IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 9, no. 1, pp. 56-73, 1987.

[14] J.-Y. Shieh, H. Zhuang, and R. Sudhakar, "Motion estimation from a sequence of stereo images: a direct method, " IEEE Trans. System, Man, Cybern, vol. 24, no. 7, pp. 1044-1053, 1994.

[15] S. B. Xu, "Qualitative depth from monoscopic cues, " Image Processing and its Applications, pp. 437-440, 1992.