Quaternion-Based Tracking of Multiple Objects in Synchronized Videos*

Quming Zhou[1], Jihun Park[2], and J.K. Aggarwal[1]

[1] Department of Electrical and Computer Engineering
The University of Texas at Austin
Austin, TX 78712
zhou@ece.utexas.edu, aggarwaljk@mail.utexas.edu
[2] Department of Computer Engineering
Hongik University
Seoul, Korea
jhpark@hongik.ac.kr

Abstract. This paper presents a method for tracking multiple objects using multiple cameras that integrates spatial position, shape and color information to track object blobs. Given three known points on the ground, camera calibration is computed by solving a set of quaternion-based nonlinear functions rather than solving approximated linear functions. By using a quaternion-based method, we can avoid the singularity problem. Our method focuses on establishing correspondence between objects and templates as the objects come into view. We fuse the data from individual cameras using an Extended Kalman Filter (EKF) to resolve object occlusion. Results based on calibration via Tsai's method as well as our method are presented. Our results show that integrating simple features makes the tracking effective, and that EKF improves the tracking accuracy when long term or temporary occlusion occurs.

1 Introduction and Previous Work

The efficient tracking of multiple objects is a challenging and important task in computer vision, with applications in surveillance, video communication and human-computer interaction. Many factors such as lighting, weather, unexpected intruders or occlusion may affect the efficiency of tracking in an outdoor environment. One solution is to use multiple cameras [1–3].

In our previous paper [4], we presented a single camera module for tracking moving objects in an outdoor environment and classifying them into three categories: single person, people group or vehicle. Our method integrates motion, spatial position, shape and color information to track object blobs. Although we were encouraged by the results of our single camera tracker, several unresolved problems remain, mainly due to object occlusion.

* This work was partially supported by grant No. 2000-2-30400-011-1 from the Korea Science and Engineering Foundation. We thank Ms. Debi Prather for proofreading.

A. Yazici and C. Şener (Eds.): ISCIS 2003, LNCS 2869, pp. 430–438, 2003.

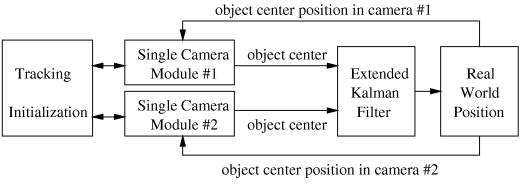


Fig. 1. Architecture of the proposed system using multiple cameras.

In this paper, we address the issue of 3D object trajectory [5]. The contributions of our paper are (1) data fusion from two cameras' observations using a simple Kalman filter(an approach that has not been explored in computer vision) and (2) a three point based camera calibration method that is developed and compared to Tsai's calibration method. Our tracking objective is to establish a correspondence between the image structures of consecutive frames over time to form persistent object trajectories. We focus on developing a methodology for tracking multiple objects in the view of two different fixed cameras, whose transformation matrix and focal length are estimated from coplanar control points. Real-world coordinates are used for multiple camera tracking. (Our single camera system [4] used 2D image coordinates.) We apply an Extended Kalman Filter to fuse the separate observations from the two cameras into a position and velocity in real world coordinates. If the target object becomes occluded from the view of one camera, our tracker switches to the other camera's observation. Tracking is initialized by solving a constrained linear least squares problem. The constraint is that a moving object in the real world always has some height. It helps us to remove the shadow on the ground plane, whose height is zero. The Kalman filter has been used in multi-sensor data fusion [6, 7]. Usually, the Kalman filter is applied to each sensor's data [8] or to combined data with additional logic [9]. We use the Kalman filter to fuse data based on the assumption that there is a mathematical relationship between the target object's image positions in the two synchronized cameras. Measurements from two synchronized cameras provide enough information to estimate the state variables of the system, the position and velocity in real world coordinates. Figure 1 shows the architecture of our proposed multiple camera system.

2 Multiple Camera Tracking

This section focuses on developing a methodology for tracking objects in the view of two fixed cameras. Compared to Tsai's five-point calibration method[10], we use only three planar control points. The theory is based on an image homography, the transformation from one plane to another. We consider the tracking problem as a dynamic target tracked by two cameras, each with different measurement dynamics and noise characteristics. We combine the multiple cameras to obtain a joint tracking that handles occlusion better than single camera-based

tracking. We fuse the individual camera observations to obtain combined measurements and then use a Kalman filter to obtain a final state estimate based on the fused measurements. Measurements are fused by increasing the dimension of the observation vector of the Kalman filter, based on the assumption that there is a mathematical relationship between the positions of an object in each camera view. Since the relationship is nonlinear, an Extended Kalman Filter is used to estimate the state vector.

2.1 Quaternion and Its Expansion to Translation

A quaternion needs four variables to represent a single rotation or orientation, which has 3 DOFs(degrees of freedom). Euler angles sometimes create problems such as the Gimbal lock problem, where we lose one degree of freedom and may get an infinite number of solutions, leading to singularity. But we avoid this by using a quaternion. A quaternion is also computationally faster since it does not require the computation of trigonometric equations. Because a quaternion can represent only rotation, we have extended the quaternion by adding a translation step[11]. The quaternion is defined by $q = (\cos\frac{\theta}{2}, \bar{n}\sin\frac{\theta}{2}) = (w, pi + qj + rk)$, where $\bar{n}$ represent axis of rotation, θ rotation angle around $\bar{n}$, and $\cos\frac{\theta}{2} = w$ represents the real part of a quaternion, while $\bar{n}\sin\frac{\theta}{2} = pi + qj + rk$ represents the imaginary part of a quaternion. We denote $\bar{v}$ to be a v vector. Let h be an extension of the quaternion, $h = (w, p, q, r, x, y, z)$. The rotational part, (w, p, q, r) where $w^2 + p^2 + q^2 + r^2 = 1$, is the same as that of a quaternion. The translational part is (x, y, z). Multiplication of two extended-quaternions is defined as follows.

$$\begin{aligned} h_1 * h_2 &= (w_1, v_1, t_1) * (w_2, v_2, t_2) \\ &= (w_1 * w_2 - \bar{v}_1 \cdot \bar{v}_2, w_1 * \bar{v}_2 + w_2 * \bar{v}_1 + \bar{v}_1 \times \bar{v}_2, \\ &\quad (w_1^2 - \bar{v}_1 \cdot \bar{v}_1)\bar{t}_2 + 2\bar{v}_1(\bar{v}_1 \cdot \bar{t}_2) + 2w_1(\bar{v}_1 \times \bar{t}_2) + \bar{t}_1) \qquad (1) \\ \text{where} \quad & \bar{v}_i = (p_i, q_i, r_i),\ \bar{t}_i = (x_i, y_i, z_i),\ |w_i|^2 + |v_i|^2 = 1, i = 1, 2 \end{aligned}$$

2.2 Camera Calibration with Three Known Planar Points

Let us assume we know three planar points $\bar{a}$, $\bar{b}$, and $\bar{u}$, where $\bar{a} = [a_x\ a_y\ a_z]^T$ etc. without any camera distortion. These points are represented in world coordinates and the same notation for these points is ${}^W\bar{a}$,${}^W\bar{b}$, and ${}^W\bar{u}$, respectively. Because we need to handle coordinate transformations, we denote W as world coordinate, C_1 and C_2 (for short, C) as coordinates of camera #1 and #2, respectively. ${}^W_C T$ is a homogeneous transformation matrix that transforms data represented in C coordinate to corresponding data in W coordinate. Let the three input points be ${}^W\bar{a}$,${}^W\bar{b}$, and ${}^W\bar{u}$, represented in real world coordinate, W. Let I_1 and I_2 (for short, I) denote the 2D image coordinates of camera #1 and #2, respectively. Input points in image coordinates that are projected points of three real world points are ${}^Ia_x$, ${}^Ia_y$, ${}^Ib_x$, ${}^Ib_y$, ${}^Iu_x$ and ${}^Iu_y$. The unknowns are transformation matrices between W and C_i (where i, $i = 1, 2$, is the

camera index) coordinates, and focal length, f_i. In order to convert points represented in world coordinates to camera #1 coordinates, we compute the following equations. $h_1 = (w_1, p_1, q_1, r_1, x_1, y_1, z_1)$ is camera #1 coordinate's quaternion-based representation in W. (w_1, p_1, q_1, r_1) is camera orientation representation in quaternion, while (x_1, y_1, z_1) is camera #1 coordinate origin represented in W. We solve these equations for each camera transformation. The following equations are scalar components of the translational vector part of equation 1.

$$\begin{aligned}
R_1 &= w_1 * w_1 - p_1 * p_1 - q_1 * q_1 - r_1 * r_1, \\
h_1 &= (w_1, p_1, q_1, r_1, x_1, y_1, z_1) \\
{}^{C_1}a_X &= (R_1 * {}^{W}a_x + 2. * (p_1 * {}^{W}a_x + q_1 * {}^{W}a_y + r_1 * {}^{W}a_z) * p_1 \\
&\quad +2. * w_1 * (q_1 * {}^{W}a_z - r_1 * {}^{W}a_y) + x_1) \\
{}^{C_1}a_Y &= (R_1 * {}^{W}a_y + 2. * (p_1 * {}^{W}a_x + q_1 * {}^{W}a_y + r_1 * {}^{W}a_z) * q_1 \\
&\quad +2. * w_1 * (r_1 * {}^{W}a_x - p_1 * {}^{W}a_z) + y_1) \\
{}^{C_1}a_Z &= -(R_1 * {}^{W}a_z + 2. * (p_1 * {}^{W}a_x + q_1 * {}^{W}a_y + r_1 * {}^{W}a_z) * r_1 \\
&\quad +2. * w_1 * (p_1 * {}^{W}a_y - q_1 * {}^{W}a_x) + z_1)
\end{aligned} \tag{2}$$

Similarly we can compute for ${}^{C_1}b_X$, ${}^{C_1}b_Y$, ${}^{C_1}b_Z$, ${}^{C_1}u_X$, ${}^{C_1}u_Y$, and ${}^{C_1}u_Z$. Then corresponding points on camera image #1 are computed as ${}^{I_1}a_X = \frac{{}^{C_1}a_X}{-{}^{C_1}a_Z} f_1$, ${}^{I_1}a_Y = \frac{{}^{C_1}a_Y}{-{}^{C_1}a_Z} f_1$, ${}^{I_1}b_X = \frac{{}^{C_1}b_X}{-{}^{C_1}b_Z} f_1$, ${}^{I_1}b_Y = \frac{{}^{C_1}b_Y}{-{}^{C_1}b_Z} f_1$, ${}^{I_1}u_X = \frac{{}^{C_1}u_X}{-{}^{C_1}u_Z} f_1$, and ${}^{I_1}u_Y = \frac{{}^{C_1}u_Y}{-{}^{C_1}u_Z} f_1$ as well as $w_1^2 + p_1^2 + q_1^2 + r_1^2 = 1$. There are seven equations to be solved. Our eight unknowns are w_1, p_1, q_1, r_1, x_1, y_1, z_1, and f_1. While moving the origin of camera coordinate along the gaze vector, we can get an infinite proper combination of x_1, y_1, z_1, f_1, satisfying the projection. For the numerical solution, we fix f_1 values and solve a set of nonlinear equations with seven equations and seven unknowns. Given a fourth known planar point, the final set of solutions was selected among the infinite set of possible solutions such that it gives minimum calibration error.

In order to determine the mathematical relationship between the observations from two different views, we calibrate the two cameras using three coplanar control points as explained above. After the calibration, we know the homogeneous transformation ${}^{W}_{C_1}T^{-1} = T$, ${}^{W}_{C_2}T^{-1} = S$, focal length, $f_1 = f_T$ and $f_2 = f_S$. If a point $(x,\ y,\ z)$ on a plane in the real world is projected on both cameras, a pixel of the *k-th* image from camera #1 is denoted as $(Q_{1x,k}, Q_{1y,k})$, while a pixel of the *k-th* image from camera #2 is denoted as $(Q_{2x,k}, Q_{2y,k})$.

2.3 Extended Kalman Filter (EKF)

Knowing the cameras' calibration, we are able to merge the object's tracking from two different views into a real world coordinate view. The 3D tracking data is the state vector, which cannot be measured directly. The object's positions in two camera views are the measurement . We assume a constant velocity between two consecutive frames.

The dynamic equation related to the state vector X_k is described as follows,

$$X_{k+1} = \Phi_k X_k + W_k \tag{3}$$

where $X_k = [x\ y\ z\ \dot{x}\ \dot{y}\ \dot{z}]^T$ is the spatial position and velocity in real world coordinates; W_k is a white noise with known covariance structure, $E(W_k, W_j) = R_k \delta_{k,j}$, t represents time, and

$$\Phi_k = \begin{bmatrix} 1 & 0 & 0 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 & \Delta t & 0 \\ 0 & 0 & 1 & 0 & 0 & \Delta t \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \text{ is the transition matrix.}$$

We derive the observation equation from the object's positions in two camera views. Let $(Q_{1x,k}, Q_{1y,k})$ and $(Q_{2x,k}, Q_{2y,k})$ be the corresponding image positions of object Q in two camera views. The observation vector

$$Z_k = [Q_{1x,k}, Q_{1y,k}, Q_{2x,k}]^T = h_k(X_k) + V_k \tag{4}$$

where V_k is a white noise with known covariance structure $E(V_k, V_j) = Q_k \delta_{k,j}$ and $E(V_k, V_j) = 0$, $h_k$is a non-linear function relating the state vector X_k to the measurement Z_k.

The dynamic equation is linear but the measurement equation is nonlinear, so the EKF is used to estimate the state vector X_k. Expanding in a Taylor series and neglecting higher order terms,

$$\begin{aligned} Z_k = h_k(X_k) + V_k &= h_k(\hat{X}_{k|k-1}) + H_k^{'}(X_k - \hat{X}_{k|k-1}) + V_k \\ &= H_k^{'} X_k + V_k + h_k(\hat{X}_{k|k-1}) - H_k^{'} \hat{X}_{k|k-1} \end{aligned} \tag{5}$$

where $H_k^{'}$is the Jacobian matrix of partial derivations of $h_k(X_k)$ with respect to X_k, $\hat{X}_{k|k-1}$ means the estimation of X_k from the X_{k-1}.

$$H_k^{'} = \frac{\partial h_k}{\partial X} = \begin{bmatrix} \frac{\partial Q_{1x,k}}{\partial x} & \frac{\partial Q_{1x,k}}{\partial y} & \frac{\partial Q_{1x,k}}{\partial z} & \frac{\partial Q_{1x,k}}{\partial \dot{x}} & \frac{\partial Q_{1x,k}}{\partial \dot{y}} & \frac{\partial Q_{1x,k}}{\partial \dot{z}} \\ \frac{\partial Q_{1y,k}}{\partial x} & \frac{\partial Q_{1y,k}}{\partial y} & \frac{\partial Q_{1y,k}}{\partial z} & \frac{\partial Q_{1y,k}}{\partial \dot{x}} & \frac{\partial Q_{1y,k}}{\partial \dot{y}} & \frac{\partial Q_{1y,k}}{\partial \dot{z}} \\ \frac{\partial Q_{2x,k}}{\partial x} & \frac{\partial Q_{2x,k}}{\partial y} & \frac{\partial Q_{2x,k}}{\partial z} & \frac{\partial Q_{2x,k}}{\partial \dot{x}} & \frac{\partial Q_{2x,k}}{\partial \dot{y}} & \frac{\partial Q_{2x,k}}{\partial \dot{z}} \end{bmatrix} \tag{6}$$

2.4 Tracking Initialization

Tracking initialization is a process of labeling a new object in one camera. If this new object has a correspondence in the other camera, it should be assigned the same tag number. The reason behind tracking initialization is that when we recover 3D coordinates of an object by computing $Q_{1x,k}$, $Q_{1y,k}$ and $Q_{2x,k}$, and project the estimated (x, y, z) using the transformation matrix computed using camera calibration, the estimated $Q_{2y,k}$ should be near the observed $Q_{2y,k}$. We

Fig. 2. Two input video streams.

initialized tracking by solving a constrained linear least squares problem, which is described as follows:

$$\min_{y} \quad \|GY - D\|_2^2 \quad \text{such that } z > z_0$$

$$\text{where } Y = [x\ y\ z]^T$$

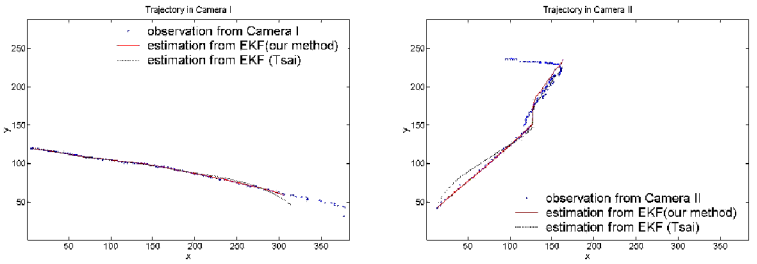
$$G = \begin{bmatrix} T_{11}f_T - Q_{1x}T_{13} & T_{21}f_T - Q_{1x}T_{23} & T_{31}f_T - Q_{1x}T_{33} \\ T_{12}f_T - Q_{1y}T_{13} & T_{22}f_T - Q_{1y}T_{23} & T_{32}f_T - Q_{1y}T_{33} \\ S_{11}f_S - Q_{2x}S_{13} & S_{21}f_S - Q_{2x}S_{23} & S_{31}f_S - Q_{2x}S_{33} \\ S_{12}f_S - Q_{2y}S_{13} & S_{22}f_S - Q_{2y}S_{23} & S_{32}f_S - Q_{2y}S_{33} \end{bmatrix}$$

$$D = \begin{bmatrix} Q_{1x}T_{43} - T_{41}f_T & Q_{1y}T_{43} - T_{42}f_T & Q_{2x}S_{43} - S_{41}f_S & Q_{2y}S_{43} - S_{42}f_S \end{bmatrix}^T.$$

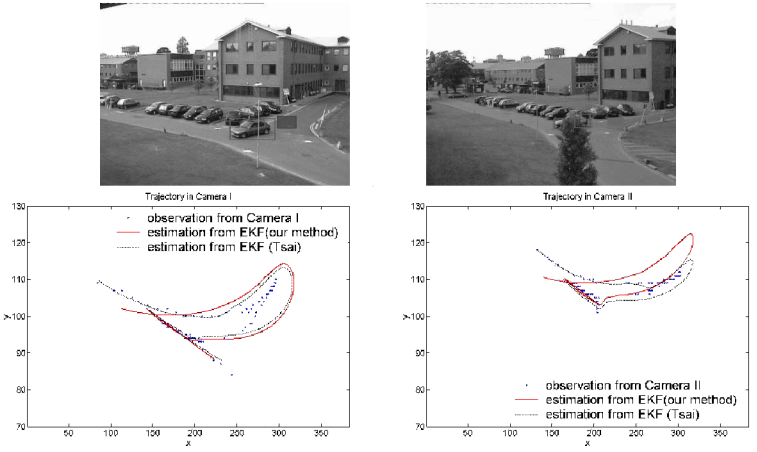
We add the constraint $z > z_0$ since a moving object in the real world always has some height. This helps us to remove the shadow on the ground plane, whose height is zero. We regard the object Q_1 in camera #1 and the object Q_2 in camera #2 as the same object Q if the sum from the above optimization problem is the minimum of all possible combinations.

3 Experiments in Multiple Camera Tracking

We use the data sets provided by the PETS2001 and other videos to evaluate the proposed tracking system. Figs. 2-3 give multiple camera examples. In Fig. 2, the first row frames are from camera #1 and the second row frames are from camera #2. In column (a), the van is still. Column (b) shows a new candidate template is created when the van's movement is detected by camera #2. (The van's motion is not detected by camera #1 because the motion is too small.) The frames in column (c) show the point at which the candidate template becomes a true template and tracking begins for both cameras. The van in Fig. 2(a)



(a)Trajectory of the two people in Fig. 2.



(b)Trajectory of the car.

Fig. 3. Trajectory of the two people and a car.

stays still for a long time; hence, it is merged into the background. It suddenly starts in the next frame. A 15-frame median filter updates the background, and a candidate new template is created, as shown by the rectangular blob in Fig. 2(b). This new candidate lasts more than three successive frames and thus begins to be tracked as a moving object in both camera as is seen in Fig. 2(c). When the object moves out of the overlapping views of cameras #1 and #2, it is tracked by the single camera module.

Fig. 3(a)-(b) shows two trajectory examples. The dots are the observations in each camera and the lines are estimates from the EKF using Tsai's 5-point calibration method and our 3-point method. In Fig. 3(a), the projections from the real world trajectory estimated by EKF fit the observations from the two cameras pretty well. It can be seen that our three point based method is better than Tsai's five point based method. In Fig. 3(b), there is some divergence between the projections and the observations. The divergence comes from the complex trajectory of the car. In this case Tsai's method gives better results than our method.

Our experiments demonstrate that the overall performance of multiple camera tracking is better than that of a single camera, especially when occlusion occurs. The EKF works well with most temporary occlusions, and the tracking initialization process can deal with long-term occlusion of say, more than 100 frames. Using multiple cameras, we can track the bicyclist in data set #2, which is occluded for a long time by the tree in camera #1 and thus would be difficult to track using only that camera. More accurate camera calibration will reduce the tracking error at the cost of having a more complex observation equation in the EKF. Multiple camera tracking relies on the objects remaining in two camera views most of time. When the target object moves out of the field of view of one camera permanently, the tracking returns to the single camera model.

4 Conclusions

In this paper, we have presented a system for tracking using synchronized multiple cameras in an outdoor environment. We combine spatial position, shape and color to achieve good performance in tracking people and vehicles. Our three point based calibration method is comparable to Tsai's five point based method. The Extended Kalman Filter fuses data from multiple cameras and performs quite well with occlusion. However, EKF may not work as well when long-term occlusion happens in both camera views. Based on our success of tracking multiple objects across multiple video streams, the reported research may be extended to recognizing moving object activities in multiple perspectives.

References

1. Cai, Q., Aggarwal, J.: Tracking human motion in structured environments using a distributed-camera system. IEEE Trans. on PAMI **21** (1999) 1241–1247
2. Khan, S., Javed, O., Rasheed, Z., Shah, M.: Human tracking in multiple cameras. In: Proc. of 8th Int. Conf. on Computer Vision. Volume Jan., Vancouver, Canada (2001) 331–336
3. S.L.Dockstsder, Tekalp, A.: Multiple camera tracking of interacting and occluded human motion. Proc. of the IEEE **89** (2001) 1441–1455
4. Zhou, Q., Aggarwal, J.: Tracking and classifying moving objects from video. In: Int. Workshop on Performance Evaluation of Tracking and Surveillance, Kauai, Hawaii (2001)
5. Bradshaw, K., Reid, I., Murray, D.: The active recovery of 3d motion trajectories and their use in prediction. IEEE Trans. on PAMI **19** (1997) 219–234
6. Gan, Q., Harris, C.: Comparison of two measurement fusion method for kalman-filter-based multisensor data fusion. IEEE Trans. on Aerospace and Electronic System **37** (2001) 273–279
7. Sasiadek, J., Hartana, P.: Sensor data fusion using kalman filter. In: Proc. of 3rd Int. Conf. on Information Fusion. Volume 2., Paris, France (2000) 19–25
8. Loffeld, O., Kramer, R.: Phase unwrapping for sar interferometry. a data fusion approach by kalman filtering. In: Proc. of Int. Symposium on Geosciences and Remote Sensing. Volume 3., Hamburg, Germany (1999) 1715–1717

9. Escamilla-Ambrosio, P., Mort, N.: A hybrid kalman filter-fuzzy logic architecture for multisensor data fusion. In: Proc. of Int. Symposium on Intelligent Control, Mexico City, Mexico (2001) 364–369
10. Tsai, R.: A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. IEEE Journal of Robotics and Automation **RA-3** (1987) 323–344
11. Park, J., Kim, S.: Kinematics and constrained joint design using quaternion. In: The 2002 International Conf. on Imaging Science, Systems, and Technology. (2002)