AN ADAPTIVE ZOOM ALGORITHM FOR TRACKING TARGETS USING PAN-TILT-ZOOM CAMERAS

Himanshu Shah and Darryl Morrell

Department of Electrical Engineering
Arizona State University
Himanshu.Shah@asu.edu, morrell@asu.edu

ABSTRACT

We address the problem of configuring pan-tilt-zoom cameras to track a target maneuvering in three dimensions; in particular, we propose an adaptive zoom algorithm that minimizes target localization errors by adaptively changing the camera focal-length. The target tracker is implemented using a Rao-Blackwellized particle filter; the camera focal-length is adjusted so that the images of a given percentage of particles fall onto the camera image plane. The focal-length adjustment is also modified by a confidence factor that reflects the accuracy of the target position estimate. We evaluate the performance of the adaptive zoom algorithm using Monte Carlo simulations. These simulations demonstrate that the adaptive zoom algorithm has a smaller average squared position estimate error than a comparable fixed zoom algorithm.

I. INTRODUCTION

Sensor configuration is currently an area of significant research interest; development of agile sensors, coupled with considerable increases in available computing power, have significantly increased the performance impact of configuration strategies. This is evident in recent work [1, 2, 3] involving the configuration of one or more foveal sensors to track a moving target. In this paper, we adapt these foveal sensor configuration algorithms to the problem of configuring pan-tilt-zoom cameras to track a target maneuvering in three dimensions. In particular, we propose an adaptive zoom algorithm that minimizes localization errors by adaptively changing the focal-length of the camera. Zooming in onto a target enhances the localization accuracy. However excess 'zoom-in' can result in a target 'loss' (when the target image falls off the image plane). An excess 'zoom-out' can inhibit the ability of the camera to provide accurate estimates.

The configuration algorithm uses a particle filter that is based on a constant-velocity target dynamics model and on a simple three-dimensional camera geometry model. The adaptive zoom algorithm adjusts the camera focal-length until a given percentage of projected particles fall onto the image plane; the focal-length is also adjusted by a confidence factor that influences and is influenced by whether the zooming algorithm is aggressive or conservative. The proposed algorithm, which we call the Adaptive Zoom Technique for Enhanced Capture (AZTEC), uses two cameras to track a point target and is discussed in detail in the following sections. The 3-D camera geometry is elucidated in Section II. Section III provides the dynamic motion and observation models used to implement the recursive Bayesian filter. Section IV describes the proposed algorithm. Section V illustrates the potential benefits of this algorithm relative to constant zoom through simulation results. Conclusions are made in Section VI.

This work supported by AFOSR under grant F49620-03-1-0117.

II. 3-D CAMERA GEOMETRY

We now consider the relationship between the target position and the location of its image when projected onto a camera image plane [4, 5]. We use a pin-hole camera model, and do not consider distortion or other issues that arise in real optical systems. The target state (position and velocity) is formulated in a three-dimensional cartesian coordinate system denoted the World Coordinate System (WCS). The camera imaging geometry is formulated in terms of the camera's reference frame called the Camera Coordinate System (CCS); the relationship between the CCS and the WCS is defined in terms of several transformation matrices.

A point target located at $A_w = (X_w, Y_w, Z_w)$ in the WCS and $A_c = (X_c, Y_c, Z_c)$ in the CCS is projected onto a point $a = (x, y)$ on the camera image plane. We first consider the projection from A_c to a and then the relationship between A_c and A_w. The projection from A_c to a is a perspective transformation which can be expressed using linear transformations and *homogeneous coordinates*. If $\tilde{a}$ is the homogeneous representation of a 2-D point a, then

$$\tilde{a} = (x, y, z) \Leftrightarrow a = (\frac{x}{z}, \frac{x}{z})$$

Similarly, if $\tilde{A}$ is the homogeneous representation of the three dimensional point A, then

$$\tilde{A} = (X, Y, Z, U) \Leftrightarrow A = \left(\frac{X}{U}, \frac{Y}{U}, \frac{Z}{U}\right)$$

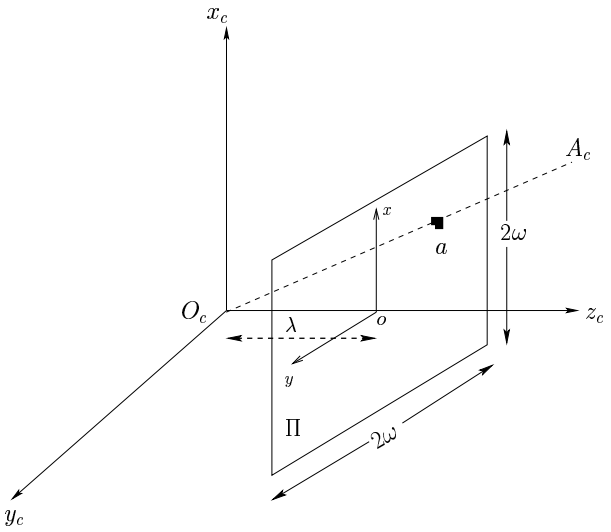

Fig. 1. Standard Perspective Projection

Let O_c be the center of projection which is at the origin $(0, 0)$. The image plane Π of the camera has dimensions $2\omega \times 2\omega$; it is parallel to the xy-plane of the CCS and at a distance λ along the camera's principal axis (the z_c-axis) (Fig. 1). λ is the focal-length of the camera. A point $A_c = (X_c, Y_c, Z_c)$ in the CCS is projected to a point $a = (x, y)$ on the image plane. The relationship between a and A_c is given by the Thales theorem:

$$x = \frac{\lambda X_c}{Z_c}, \qquad y = \frac{\lambda Y_c}{Z_c} \tag{1}$$

Using homogeneous coordinates, (1) can be written as —

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1/\lambda & 0 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} \tag{2}$$

or in matrix notation, $\quad \tilde{a} = T\,\tilde{A}$.

- $\tilde{a}$ is the homogeneous representation of a.
- $\tilde{A_c}$ is the homogeneous representation of A_c.
- T is the camera projection matrix (also called the intrinsic parameter matrix).

In general, the CCS is not aligned with the WCS. [5] explains how a point in the WCS is projected into the CCS. This requires the application of a translation matrix G that translates the origin of the WCS to that of the CCS located at $(g_x,\, g_y,\, g_z)$ followed by a rotation matrix R to align the two coordinate systems. Rotation of the coordinate system is performed by first rotating by an angle β about the y-axis and then an angle α about the x-axis. If the center of the image plane is not located on the z_c-axis, we model the displacement as as an image displacement matrix C. Finally, the perspective projection matrix T projects the resulting CCS coordinate onto the 2-D image plane:

$$\tilde{a} = TCRG\tilde{A} \tag{3}$$

Here, $G = \begin{bmatrix} 1 & 0 & 0 & g_x \\ 0 & 1 & 0 & g_y \\ 0 & 0 & 1 & g_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$, $C = \begin{bmatrix} 1 & 0 & 0 & c_x \\ 0 & 1 & 0 & c_y \\ 0 & 0 & 1 & c_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$,

$$R = R_\alpha R_\beta = \begin{bmatrix} \cos\beta & 0 & -\sin\beta & 0 \\ \sin\alpha\sin\beta & \cos\alpha & \sin\alpha\cos\beta & 0 \\ \cos\alpha\sin\beta & -\sin\alpha & \cos\alpha\cos\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

III. TARGET AND OBSERVATION MODELS

We denote the target state at k as $\mathbf{x}_k$, and define it to be the target's three-dimensional position and velocity in the WCS:

$$\mathbf{x}_k = \begin{bmatrix} A_{w_{k+1}} & \dot{A}_{w_{k+1}} \end{bmatrix}^T$$

Target dynamics are modeled by a discrete-time constant-velocity state equation of the form

$$\mathbf{x}_{k+1} = F \cdot \mathbf{x}_k + \mathbf{w}_k \tag{4}$$

where $F = \left[\begin{array}{c|c} I_3 & \Delta t \cdot I_3 \\ \hline \mathbf{0}_3 & I_3 \end{array}\right]$, Δt is the time between measurements and $\mathbf{w}_k \sim \mathcal{N}(0, Q_k)$.

Let a_{jk} be the projection of A_{w_k} (the target location at time k) onto the image plane of camera j, $j = 1, 2$:

$$\tilde{a}_{jk} = TC_jR_jG_j\tilde{A}_{w_k}$$

The measurement model is

$$\mathbf{Z}_k = \begin{bmatrix} Z_k^1 \\ Z_k^2 \end{bmatrix} = \begin{bmatrix} a_{1k} \\ a_{2k} \end{bmatrix} + \mathbf{v}_k \tag{5}$$

Here $\mathbf{v}_k \sim \mathcal{N}(0, R_k)$ and models pixelation noise as well as the errors in the parameters of the camera calibration matrix. Note that the measurements $\mathbf{Z}_k$, $k = 1, 2, \ldots$ are a function only of the target position. We assume that the observation errors for the 2 cameras are independent:

$$p(\mathbf{Z}_k | A_k) = p(Z_k^1 | A_k) \cdot p(Z_k^2 | A_k) \tag{6}$$

IV. CAMERA CONFIGURATION ALGORITHM

We implement the tracker with a Rao-Blackwellized Particle Filter (RBPF) [6]. To configure the camera at time k, the particle filter provides a predicted target location and (through the spread of the particles) a measure of the uncertainty of this predicted target location. Both cameras are pointed at the predicted target location; the focal-lengths of the cameras (and hence the cameras' zooms) are set so that the camera image plane contains a set percentage of the particles' images. In this paper, we adapt the zooms of both cameras based on computations performed for only one camera; thus, in this section we drop the explicit enumeration of cameras. The camera configuration algorithm could be extended to adapt the zoom of each camera independently.

The algorithm that configures the camera focal-length at time k begins with the previous camera focal-length λ_{k-1}. The particle filter predicts the particles ahead from $k-1$ to k, creating a set of particles $\{\mathbf{x}_k^i\}_{i=1}^N$; the particles are then projected onto the image plane using λ_{k-1}. The focal-length is adjusted so that approximately $\kappa\%$ of the projected particles lie within the bounds of the image plane; the adjusted focal-length is denoted λ_k'. The adjusted focal-length is then weighted by a confidence factor f_c to set the camera focal-length λ_k that will be used to obtain the observation $\mathbf{Z}_k$.

We choose λ_k' as follows. Let $A_{w_k}^i$ be the position component of particle $\mathbf{x}_k^i$ and let $A_{c_k}^i = \begin{bmatrix} X_{c_k}^i & Y_{c_k}^i & Z_{c_k}^i \end{bmatrix}^T$ be the projection of $A_{w_k}^i$ onto the CCS. From (1), for a given λ_{k-1}, the projection of $A_{c_k}^i$ onto the image plane is

$$x_k^i = \lambda_{k-1}\frac{X_{c_k}^i}{Z_{c_k}^i}, \qquad y_k^i = \lambda_{k-1}\frac{Y_{c_k}^i}{Z_{c_k}^i}$$

Define $r_k^i = \sqrt{(x_k^i)^2 + (y_k^i)^2}$, and let $\bar{i}$ be the index of the κ-percentile (with κ ranging from 90% to 95%) value of r_k^i. Note that $\bar{i}$ is not a function of λ_{k-1}. We choose λ_k' as the largest value that satisfies both

$$\omega \geq \lambda'_k \left|\frac{X_{c_k}^{\bar{i}}}{Z_{c_k}^{\bar{i}}}\right| \quad \text{and} \quad \omega \geq \lambda'_k \left|\frac{Y_{c_k}^{\bar{i}}}{Z_{c_k}^{\bar{i}}}\right| \tag{7}$$

This is to avoid the target from moving out of the field of view of the camera.

The updated focal-length λ_k which is used to acquire $\mathbf{Z}_k$ is the product

$$\lambda_k = \lambda_k' \; f_{c_{k-1}} \tag{8}$$

Here $f_{c_{k-1}}$ is the *confidence factor* that reflects our belief that the target was imaged at time $k-1$. The confidence factor is determined using

$$f_{c_{k-1}} = e^{-\gamma \cdot \sigma_{k-1}} \tag{9}$$

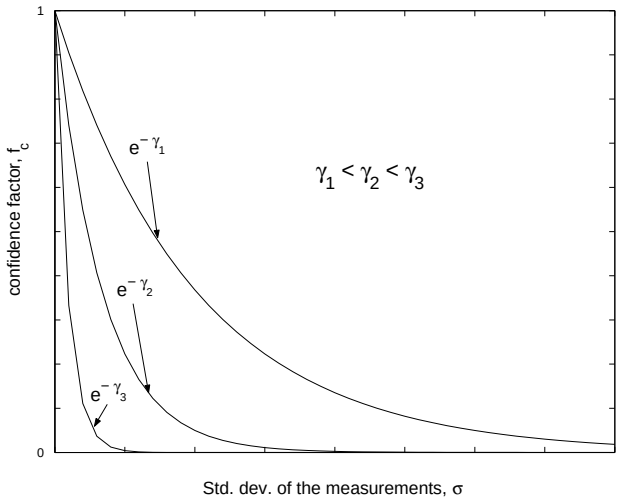

Fig. 2. Plot showing f_{c_k} as a function of σ_k for various values of γ

where γ is a settable parameter that determines how 'conservative' or 'aggressive' the zooming will be. σ_{k-1}^2 is the trace of the empirical covariance matrix of the particles $\{\mathbf{x}_{k-1}^i\}$ projected onto the observation plane with focal-length λ_{k-1}. Fig. 2 shows that smaller γ values give larger $f_{c_{k-1}}$ values and consequently more aggressive zooming.

The focal-length λ_k is used to obtain $\mathbf{Z}_k$ using (5). The weights are computed using (6). If the target is 'lost' (i. e. the target image does not fall within the image plane), then the particles are re-weighted; the weights of the particles whose projections fall on the image plane are set to zero. The algorithm is summarized in Table 1.

V. SIMULATION RESULTS

We evaluated the performance of the AZTEC algorithm using Monte Carlo simulations. In these simulations, the two cameras were positioned at $(50, 25, -25)$ and $(50, -50, 300)$. Simulations were run for two cases: (i) using a constant focal-length and (ii) adaptively changing the focal-length using the AZTEC algorithm. Different values of γ were used to investigate the effect of aggressiveness in zooming. Other chosen parameters include $\Delta t = 2$, $Q = \begin{bmatrix} (\Delta t^3/3) I_3 & (\Delta t^2/2) I_3 \\ (\Delta t^2/2) I_3 & (\Delta t) I_3 \end{bmatrix}$, $R = I_2$, $\omega = 6$ and $N = 300$. 100 Monte Carlo iterations were performed. The simulation results (Fig. 3 and Fig. 4) show that AZTEC performs significantly better than the constant zoom method. Since the target is chosen to move away from the two cameras, highly aggressive zooming i. e. $\gamma = 0.1$ gives the best performance while highly conservative zooming i. e. $\gamma = 1$ gives an average performance that is still better than when the zoom is kept constant.

Table 1. Target Tracking Algorithm

1. Generate $\{\mathbf{x}_0^i\}_{i=1}^N$ from $p(\mathbf{x}_0)$ and set $\{w_0^i\}_{i=1}^N = \frac{1}{N}$.
2. Choose initial values for the confidence factor f_{c_0} and focal-length λ_0.
3. Set $k = 1$.
4. Predict $\mathbf{x}_k^i \sim p(\mathbf{x}_k|\mathbf{x}_{k-1}^i), \{i = 1, 2, \ldots, N\}$ using RBPF [6].
5. Point the cameras to the predicted target position.
6. For each $\mathbf{x}_k^i$ project A_k^i (the position component of $\mathbf{x}_k^i$) on to the image plane of the camera using (3), then compute $\bar{i}$.
7. Compute λ_k using (7) & (8).
8. Re-project A_k^i onto the image plane using λ_k to obtain a_k^i, and compute σ_k^2.
9. Compute f_{c_k} using (9).
10. Obtain the measurement Z_k.
11. Compute the importance weights using (6). Compute the estimated target state.
12. Perform re-sampling using [7].
13. Set $k \leftarrow k + 1$ and go to step 4.

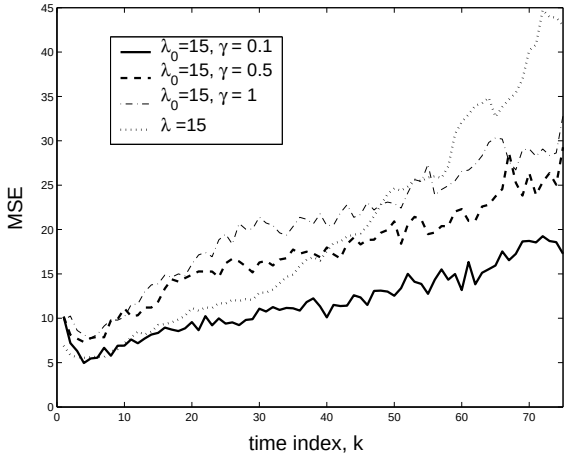

Fig. 3. MSE plot for constant and adaptive zooms

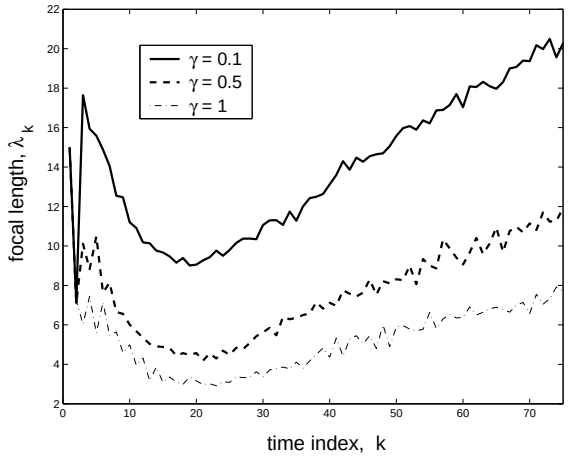

Fig. 4. Plot showing focal-length as a function of time

VI. CONCLUSIONS

In this paper, we have proposed the AZTEC algorithm that adjusts the zoom of two cameras to track a target with a Rao-Blackwellized Particle Filter. The AZTEC algorithm estimates the target state with lower average squared error than constant zoom.

VII. REFERENCES

[1] Y. Xue and D. Morrell, "Traget Tracking and Data Fusion using Multiple Adaptive Foveal Sensors," *International Conference on Information Fusion*, July 2003.

[2] Y. Xue and D. Morrell, "Adaptive Foveal Sensor for Target Tracking," *36th Asilomar Conference on Signals, Systems and Computers*, pp. 848–852, Nov. 2002.

[3] L. Li, D. Cochran, and R. Martin, "Target tracking with an attentive foveal sensor," *34th Asilomar Conference on Signals, Systems and Computers*, pp. 182–185, Oct. 2000.

[4] Stan Birchfield, "An introduction to projective geometry," for computer vision appl., Apr. 1998.

[5] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Addison-Wesley, 1992.

[6] P.-J. Nordlund and F. Gustafsson, "Sequential monte carlo filtering techniques applied to integrated navigation systems," *American Control Conference*, vol. 5, pp. 4375–4380, June 2001.

[7] M. Sanjeev Arulampalam, Simon Maskell, Neil Gordon, and Tim Clapp, "A Tutorial on Particle Filters for Online Non-linear/Non-Gaussian Bayesian Tracking," *IEEE Transactions on Signal Processing*, vol. 50, pp. 174–188, Feb. 2002.