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. 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.

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

\[
\tilde{a} = (x, y, z) \iff a = \left( \frac{x}{z}, \frac{y}{z} \right)
\]
Similarly, if $\hat{A}$ is the homogeneous representation of the three dimensional point $A$, then

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

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 $\beta$ about the $y$-axis and then an angle $\alpha$ about the $x$-axis. If the center of the image plane is not located on the $z_c$-axis, we model the displacement as an image displacement matrix $C$. Finally, the perspective projection matrix $T$ projects the resulting CCS coordinate onto the 2-D image plane:

$$\hat{a} = TCRG \hat{A}$$

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

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

III. TARGET AND OBSERVATION MODELS

We denote the target state at $k$ as $x_k$, and define it to be the target’s three-dimensional position and velocity in the WCS:

$$x_k = \begin{bmatrix} A_{w_{k+1}} & \hat{A}_{w_{k+1}} \end{bmatrix}^T$$

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

$$x_{k+1} = F \cdot x_k + w_k$$

where $F = \begin{bmatrix} I_3 & \Delta t \cdot I_3 \\ 0_3 & I_3 \end{bmatrix}$, $\Delta t$ is the time between measurements and $w_k \sim \mathcal{N}(0, Q_k)$.

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

$$\hat{a}_{jk} = TC_j R_j G_j \hat{A}_{w_k}$$

The measurement model is

$$Z_k = \begin{bmatrix} Z_k^1 \\ Z_k^2 \end{bmatrix} = \begin{bmatrix} a_{1k} \\ a_{2k} \end{bmatrix} + v_k$$

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

$$p(Z_k | A_k) = p(Z_k^1 | A_k) \cdot p(Z_k^2 | A_k)$$
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 predicts 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 $\lambda_{k-1}$. The particle filter predicts the particles ahead from time $k - 1$ to $k$, creating a set of particles $\{x_i^k\}_{i=1}^N$; the particles are then projected onto the image plane using $\lambda_{k-1}$. 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 $\lambda'_k$. The adjusted focal-length is then weighted by a confidence factor $f_c$ to set the camera focal-length $\lambda_k$ that will be used to obtain the observation $Z_k$.

We choose $\lambda'_k$ as follows. Let $A^{k}_{w_k}$ be the position component of particle $x^k_i$ and let $A^{i}_{c_k} = [X^i_{c_k} Y^i_{c_k} Z^i_{c_k}]^T$ be the projection of $A^{k}_{w_k}$ onto the CCS. From (1), for a given $\lambda_{k-1}$, the projection of $A^{i}_{c_k}$ onto the image plane is

$$x^k_i = \lambda_{k-1} \frac{X^i_{c_k}}{Z^i_{c_k}}, \quad y^k_i = \lambda_{k-1} \frac{Y^i_{c_k}}{Z^i_{c_k}}$$

Define $r^i_k = \sqrt{(x^k_i)^2 + (y^k_i)^2}$, and let $\tilde{r}$ be the index of the $k$-percentile (with $k$ ranging from 90% to 95%) value of $r^i_k$. Note that $\tilde{r}$ is not a function of $\lambda_{k-1}$. We choose $\lambda'_k$ as the largest value that satisfies both

$$\omega \geq \lambda'_{k} \left| \frac{X^i_{c_k}}{Z^i_{c_k}} \right| \quad \text{and} \quad \omega \geq \lambda'_{k} \left| \frac{Y^i_{c_k}}{Z^i_{c_k}} \right| \quad (7)$$

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

The updated focal-length $\lambda_k$ which is used to acquire $Z_k$ is the product

$$\lambda_k = \lambda'_k f_{c_{k-1}} \quad (8)$$

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

$$f_{c_{k-1}} = e^{-\gamma \sigma_{k-1}} \quad (9)$$

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

The focal-length $\lambda_k$ is used to obtain $Z_k$ 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 $\gamma$ 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 \( \{x^i_0\}_{i=1}^N \) from \( p(x_0) \) and set \( \{u^i\}_{i=1}^N = \frac{1}{N} \).
2. Choose initial values for the confidence factor \( f_{c_0} \) and focal-length \( \lambda_0 \).
3. Set \( k = 1 \).
4. Predict \( x^i_k \sim p(x_k|x^i_{k-1}), \{i = 1, 2, \ldots, N\} \) using RBPF [6].
5. Point the cameras to the predicted target position.
6. For each \( x^i_k \) project \( A^i_k \) (the position component of \( x^i_k \)) onto the image plane of the camera using (3), then compute \( \tilde{r} \).
7. Compute \( \lambda_k \) using (7) & (8).
8. Re-project \( A^i_k \) onto the image plane using \( \lambda_k \) to obtain \( \alpha^i_k \), and compute \( \sigma^2_k \).
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.

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


