Human Behavior-Based Target Tracking With an Omni-Directional Thermal Camera

Emrah Benli, *Student Member, IEEE*, Yuichi Motai, *Senior Member, IEEE*, and John Rogers, *Senior Member, IEEE*

***Abstract*—We investigate human behavior-based target tracking from omni-directional (O-D) thermal images for intelligent perception in unmanned systems. Current target tracking approaches are primarily focused on perspective visual and infrared (IR) band, as well as O-D visual band tracking. The target tracking from O-D images and the use of O-D thermal vision have not been adequately addressed. Thermal O-D images provide a number of advantages over other passive sensor modalities such as illumination invariance, wide field-of-view, ease of identifying heat-emitting objects, and long term tracking without interruption. Unfortunately, thermal O-D sensors have not yet been widely used due to the following disadvantages: low resolution, low frame rates, high cost, sensor noise, and an increase in tracking time. This paper outlines a spectrum of approaches which mitigate these disadvantages to enable an O-D thermal IR camera equipped with a mobile robot to track a human in a variety of environments and conditions. The curve matched Kalman filter is used for tracking a human target based on the behavioral movement of the human and maximum *a posteriori* (MAP)-based estimation is extended for the human tracking as long term which provides a faster prediction. The benefits to using our MAP-based method are decreasing the prediction time of a target's position and increasing the accuracy of prediction of the next target position based on the target's previous behavior while increasing the tracking view and lighting conditions via the view from O-D IR camera.**

***Index Terms*—Behavior-based target tracking, far infrared (IR) camera, human behavior understanding, mobile robot, omni-directional (O-D) camera, thermal vision.**

I. Introduction

TRACKING targets with path prediction from omni-directional (O-D) infrared (IR) sensor is proposed to increase the accuracy of tracking for unmanned systems. Current approaches are primarily focused on perspective visual band and O-D visual band for 3-D reconstruction, mapping, and tracking; furthermore, a literature search found no significant research in our area of interest, the O-D IR images and human behavior-based tracking prediction. O-D IR sensors provide numerous advantages to predict the human target's next position for precise tracking. O-D images provide 360° vision for the robot so the target can be tracked continuously. The thermal imaging feature of the O-D IR sensor provides a light independent observation of the target scene. The light independent observation and the 360° vision present an advantage for the robot by providing nonstop tracking of a target. Utilizing a human body heat signature is another advantage of O-D IR sensor which yields an easily obtained human target oriented detection. Since the other objects can provide an inconsistent heat pattern, the tracking of nonhuman targets delivers unsuccessful results in the complex texture of thermal images. Thus, the O-D IR sensor can continuously detect and track the human target in this complex texture of thermal images. However, O-D IR sensors have not yet been widely used because they have several disadvantages as well; O-D IR sensors have noise, low frame rates, and are quite costly. Since the IR sensors provide low resolution images, noise, and low frame rate combined causes high prediction as well as tracking error. In order to increase the accuracy of the predicted target position, we use a maximum *a posteriori* (MAP)-based curve matched Kalman filter (CMKF). CMKF improves the prediction accuracy for short term tracking; however, the advantage of continuous tracking with the O-D IR sensor becomes a disadvantage with its increasing computational time of CMKF approach. The low resolution and frame rate of the O-D IR camera causes a higher prediction error with the current visible band-based methods since they cannot handle O-D IR images. Thus our method is necessary to overcome the higher error. MAP-based CMKF helps to improve the prediction results and decreases the increasing computational time for long term tracking. We will look at the O-D IR view to predict the precise target position and tracking it in a faster manner by MAP-based CMKF.

The 360° view of thermal images for target tracking are obtained by a single O-D IR sensor, mounted on top of the mobile robot, as shown in Fig. 1(a). O-D images for the tracking process are taken while the target is in motion. Fig. 1(b) shows a closer look at the O-D camera used in our application and the configuration of its parts. Fig. 1(b) also shows an O-D IR image of the target in the scene. The spherical mirror reflects rays from objects toward the camera sensor and camera sees itself in the middle of the O-D IR image.

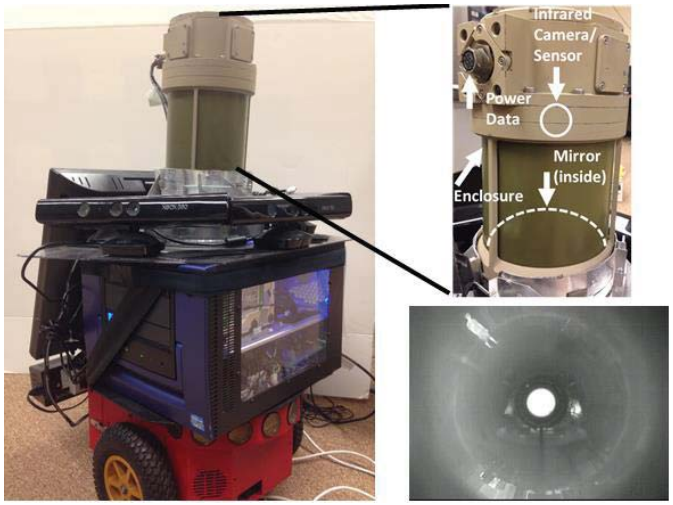
Manuscript received December 2, 2016; revised April 17, 2017 and June 13, 2017; accepted July 6, 2017. This work was supported in part by the U.S. Navy, Naval Surface Warfare Center Dahlgren, in part by the U.S. Army Research Laboratory under Grant CRADA15-18-01, in part by the Ministry of National Education of Turkey, and in part by the National Science Foundation under Grant 1054333. *(Corresponding author: Yuichi Motai.)*

E. Benli and Y. Motai are with the Department of Electrical and Computer Engineering, Virginia Commonwealth University, Richmond, VA 23284-3068 USA (e-mail: benlie@vcu.edu; ymotai@vcu.edu).

J. Rogers is with the U.S. Army Research Laboratory, Adelphi, MD, USA (e-mail: john.g.rogers59.civ@mail.mil).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCDS.2017.2726356



(a) (b)

Fig. 1. O-D IR camera. (a) Mobile robot equipped with O-D IR camera. (b) O-D camera with the placement map, and an O-D IR image.

The camera sensor is placed on the mirror's z-coordinate axis.

This paper shows that the fusion of target tracking for short and long term modified for the far IR view of targets with the curve matching (CM)-based tracking minimizes the tracking error and time. Behavior-based tracking for the human target is examined and the target position is estimated by CM and by Kalman filter (KF) for short term target movement. The human target's behavioral movement is considered so that we can analyze the human target's trajectory history and its walking behavior. In this period, analyzing movement helps our method to decide what kind of possible path the target can travel. For the recent short term trajectory, the walking behavior of the human target is continuously modeled by the CMKF method during the entire path progression of the target. The combination of these methods is updated by an MAP estimation-based statistical analysis for long term tracking. The contributions of our human behavior-based method are decreasing the prediction time of target position and increasing the accuracy of prediction for the next target position while increasing the tracking view and lighting conditions via the view from O-D IR camera.

The organization of this paper is as follows. In Section II, related works are discussed. Section III explains omni-IR camera data. Target tracking via O-D thermal images is proposed in Section IV. Then, the experiments are given in Section V. Finally, Section VI presents the conclusion and future work.

II. Related Works

We cover relevant studies on target tracking for mobile robots with a 360° thermal imager on a mobile robot. The following sections will first describe target tracking through mobile robots in Section II-A by comparing representative prediction methods. Then we explore, in Section II-B, human tracking methods based on their algorithms in order to get better results in terms of tracking than those discussed in previous methods [1]–[4].

TABLE I
Target Tracking Methods

Method	Camera Sensor	Sensor Requirements	Prediction Error
Curve Matching [18, 27-33, 37-44]	IR or Color	Camera + Range	Moderate
Kalman Filter (KF) [3, 7-13, 45-49]	IR or Color	Camera + Range	High
Maximum A Posteriori (MAP) [1, 2, 4]	Color	Camera + Range	Low

A. Target Tracking via Mobile Robots

For various years, human behavior-based target tracking has had an enormous increase in research and popularity [5], [6]. Prevalent target tracking interest has been for the human body; there are several existing methods: processing 2-D or 3-D reconstructed images by normal KF-based methods [7]–[13], or by manipulating several sensors, or laser-based via on-board laser range finder [10], [14], or by utilizing an ordered KF [9], or by quaternions [15]. Several human features may be employed [16], [17], for tracking, likewise the robot's placement in the environment can be used for tracking as well. Some other works have been applied: for detection and classification of abnormal movement [18]. The ability to recognize a human from afar through motion energy mapping [19]. In order to detect movement, extraction features are applied spatially and through chronological templates [20]. Target activity recognition is attained by a sequential silhouette analysis based on human behavior [21], [22]. But, not one algorithm implements a technique such that the robot is trained to mark and learn the target's trajectory.

Target trajectory application in some studies has shown that robots may be trained to learn from a target's motion as in [8], [23], and [24] and then tracking the target's path [12]. Other research for unfamiliar environments are modeled by a robot as in [25] and [26]. However, any prediction and estimation algorithms of a target's trajectory have not been applied to those methods aimed at target behavior analysis. Table I displays the details of the three different tracking methods that we will focus on for our target tracking methodology; CM, KF, and MAP-based method in conjunction with sensors and prediction accuracy.

B. Human Tracking Algorithms

CM-based tracking has various applications in the field of robotics. The most common methods in CM is recognition of some known curve from images, and tracking these corresponding paths [27]–[31], whereas [32] does not utilize any parameters in order for it to record images by the elasticity theory. Also, [33]–[37] are responsible for seeking contours in the images to detect as well as categorize the targets. CM can be achieved with splines, another extensive area of study [38]–[40]. Other CM studies utilize the polygonal arc methods [41], while curve characteristics matching is conducted in a curve measurement of unary and

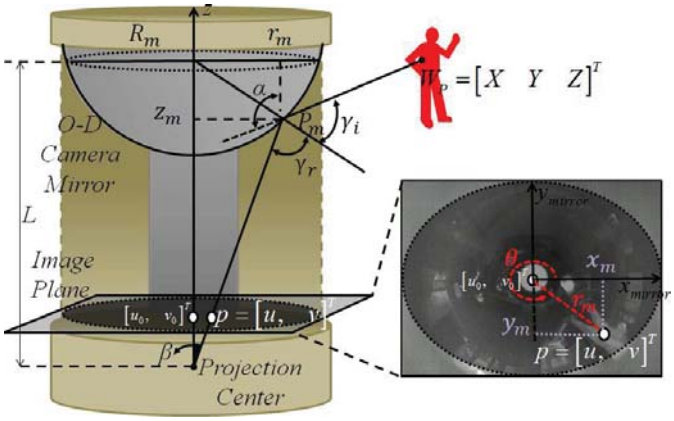


Fig. 2. Projections of spherical mirror. Unified model for projection of a space point $W_P = [X \quad Y \quad Z]^T$ to $p = [u \quad v]^T$ image point. The image shows the view of the coordinates from the top of the camera.

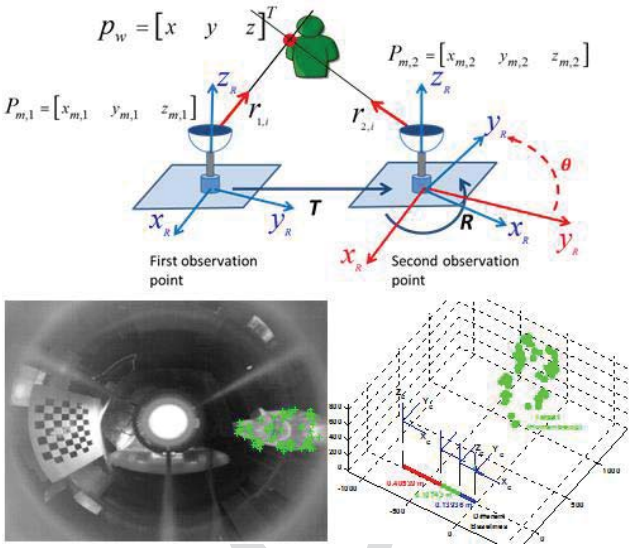


Fig. 3. 3-D reconstruction from moving stereo.

binary [42], fuzzy logic in [43], and Sethian's fast marching method in [44].

The KF consists of an equation pack which keeps the system state up to date. The system state is derived from its latter state which is being extensively used in target and trajectory tracking [45], [46]. The KF method has access to a universal solution to a quadratic mean estimation problem in terms of a rectilinear estimator [47], [48]. The CMKF is an another method that checks the reappearance in the target's movement behaviors and sections in a certain extent with the CM techniques of [3] and [49]. CMKF is a low computational cost method in comparison to previous methods.

MAP estimation has been applied to KF-based tracking in recent studies [1], [2]. The MAP estimation method helps to resolve the problem of target tracking by using the extended KF (EKF) approach such that it selects the most probable local hypotheses. A bank of MAP estimation tracking is proposed in [1]. The method introduces a solution to the linearization problem caused by EKF tracking. The least probable hypotheses are pruned to control the computational cost. The method proposed in [2] is another multitarget tracking method from a moving camera. A range sensor and particle filter is used to detect the moving targets and distances then the tracking problem was solved by MAP estimation. In this method, both target and robot positions were estimated by MAP. An IR camera for tracking small target's is used with another MAP application in [4].

III. Omni-Infrared Camera Data

$W_P = [X \quad Y \quad Z]^T$ is a real-world point in space and $p = [u \quad v]^T$ is the projection of said 3-D point from the spherical mirror onto an image, as shown in Fig. 2. The point's reflection on the spherical mirror has a ray vector of $P_m = [x_m \quad y_m \quad z_m]^T$ and this ray vector is converted into another point, $p = [u, \quad v]^T$, on the image. Equation (1) calculates the projected image point $p = [u, \quad v]^T$ given by [50]

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_u & 0 & u_0 \\ 0 & f_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \tan\beta & \cos\theta \\ \tan\beta & \sin\theta \\ & 1 \end{bmatrix} \tag{1}$$

where the first term of the right hand side, known as the camera intrinsic matrix, consists of focal lengths, f_u, f_v, and the coordinates of principle point, u_0, v_0. The angles, β and θ, are calculated by utilizing cylindrical mirror coordinates $[\theta \quad r_m \quad z_m]^T$ which include $r_m = \sqrt{x_m^2 + y_m^2}$ and $z_m = \sqrt{R_m^2 - r_m^2}$. The parameter L, the distance from the sphere center to projection center, is used to find the angle β from (2). The angle, θ, is derived by using the real-world point coordinates from $\tan^{-1}(Y/X)$. It can also be found by using the mirror coordinates of x_m, y_m since the direction of the ray vector to the world point and the mirror point is the same

$$\beta = \tan^{-1}\big(r_m / (L - z_m)\big). \tag{2}$$

The angle, α, is obtained by using the radial distance from the optical axis to a real-world point, $r = \sqrt{X^2 + Y^2}$, and the Z-coordinate of the real-world point (3). Since the reflection angles, γ_i and γ_r, on the spherical mirror surface are equal, another relationship between the angles, α and β, is expressed as $(\beta - \alpha)/2 = \tan^{-1}(r_m/z_m)$

$$\alpha = \tan^{-1}\big((Z - z_m)/(r - r_m)\big) + \pi/2. \tag{3}$$

Finding α, β, and θ offers us the transition between the mirror coordinates, P_m, and image coordinates, p, which is our target's position from experimental data set.

The calculation of 3-D coordinates of real points based on the moving stereo is illustrated in Fig. 3. The improved information of low resolution IR images helps to enhance the tracking of objects. The 3-D target coordinates for precise target tracking is obtained from our previous work, n observation points are used to calculate the target coordinates. The number of observation points, n, is adjusted with respect to the experimental results.

The solution of (4) gives two depth values (z_1, z_2) by utilizing ray vectors of the mirror coordinates, $P_{m,1} = [x_{m,1} \quad y_{m,1} \quad z_{m,1}]^T$ and $P_{m,2} = [x_{m,2} \quad y_{m,2} \quad z_{m,2}]^T$ from both images. The calculated world coordinates of the feature

points, $p_{W_1} = [\,x_1 \quad y_1 \quad z_1\,]^T$ and $p_{W_2} = [\,x_2 \quad y_2 \quad z_2\,]^T$ are found from the middle point of those depth values. If the middle point depth value is calculated by x_m and y_m coordinates of the reference image, 3-D coordinates can be obtained for each triangulation step. The reconstructed points are plotted in a form of a point cloud in Fig. 3. Any change to the target's 3-D coordinates can be tracked by our proposed tracking algorithm more precisely

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \left(\begin{bmatrix} \|P_{m,1}\|^2 & \langle P_{m,1}, -RP_{m,2} \rangle \\ \langle -RP_{m,2}, P_{m,1} \rangle & \|-RP_{m,2}\|^2 \end{bmatrix} \right)^{-1} \times \begin{bmatrix} \langle P_{m,1}, T \rangle \\ \langle T, -RP_{m,2} \rangle \end{bmatrix} \tag{4}$$

where R is a 3×3 rotation matrix and T is a 3×1 translation vector between two camera positions. The dot product operator, $\langle 3 \times 1, 3 \times 1 \rangle$, utilizes two 3×1 column vectors and a 2×1 column vector is obtained for the depth values of two feature points. Mirror coordinates are normalized so that they do not have units and rotation matrix and translation vector provides the units for real-world coordinate

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} \frac{\|P_{m,1}\|^2 \langle T, -RP_{m,2} \rangle - \langle -RP_{m,2},\, P_{m,1} \rangle \langle P_{m,1},\, T \rangle}{\|-RP_{m,2}\|^2 \|P_{m,1}\|^2 - \langle P_{m,1}, -RP_{m,2} \rangle^2} \\ \frac{\|-RP_{m,2}\|^2 \langle P_{m,1},\, T \rangle - \langle T, -RP_{m,2} \rangle \langle -RP_{m,2},\, P_{m,1} \rangle}{\|-RP_{m,2}\|^2 \|P_{m,1}\|^2 - \langle P_{m,1}, -RP_{m,2} \rangle^2} \end{bmatrix}. \tag{5}$$

After the depth coordinates are obtained from (5) by utilizing the transformation information, the coordinates of the target are attained as $p_W = [\,x \quad y \quad z\,]^T$ from the average triangulation results.

IV. Human Behavior-Based Target Tracking via Omni-Directional Thermal Images

A new method of tracking targets from images from an O-D IR camera is implemented using the behavior analysis-based tracking. Nonlinear movement of the targets makes the requirement for the IR O-D images significant for behavior-based and long term tracking in order to make a precise prediction in addition to providing tracking results. The advantage of an O-D IR is that it provides for an easier way to detect a target as well as tracking for a long period of time; however, visual band sensor-based methods have high prediction error and increasing computational time problems that make tracking difficult, making our method necessary. We are developing a new tracking algorithm for O-D thermal distribution in the following three steps. First, behavior learning by using CM in Section IV-A. Second, Section IV-B, prediction for short term tracking. Then, long term tracking with MAP estimation in Section IV-C, and finally, criteria to follow a human target from a mobile robot in Section IV-D.

A. Behavior Analysis

For human targets, we considered the data sequences of longitudinal trajectory rather than a static scene. For the first behavior analysis, a nearby target is detected and separated from the background (Fig. 4). The human position trajectories in the video sequences are determined by assembling the

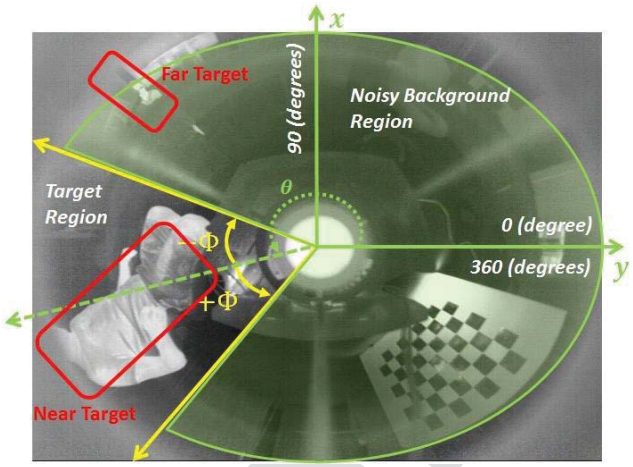


Fig. 4. Near target position is selected for trajectory history analyses.

extracted features described in a static scene of O-D reconstruction (as new cases are accrued over time into multiple frames). Behavior learning is acquired from the patterns of target's trajectories in an O-D thermal distribution. The human target's behavioral movement is the algorithm's main criteria for making a decision. The next movement of the human target considers both its present movement, and the prior walking behaviors of the target's entire trajectory in the short term prediction window.

The target's position on the O-D image is recorded for the bearing tracking first with respect to the O-D geometry coordinates which is given by the angle, θ. The starting point, θ_1, is the target's position angle from the first image, θ_c is the starting point of the current motion of the target, and θ_t is the last position of the target on the last image seen in Fig. 5(a). We converted an O-D curve to an unwrapped image trajectory line given in Fig. 5(b). O-D image angles, θ_t, are calculated from the horizontal, x_m, and vertical, y_m, coordinates of P_m mirror coordinates for the angle prediction process (6). The image trajectory positions are obtained from the horizontal coordinates of $P_W(t)$ until, $p_W(t)$, the last pixel position of the entire movement, at time t, and $p_C(m)$ is the last position of the current motion trajectory

$$\theta = \tan^{-1}\left(y_m / x_m\right). \tag{6}$$

Target distance tracking is applied to a current position to find the best recent distance of the target, in Fig. 6, when we have the distance value from each reconstruction step. The calculated distance coordinates, $z(t)$, are used from $P_W(t)$ and $P_C(m)$ strings for the entire distance and the current distance values of the target. Distance behavior analysis results will be combined with the bearing behavior results and will be sent to $P_W(t)$ and $P_C(m)$ strings, then (7) and (8), for the CM process.

The CM method [3] acquires the motion characteristics of a human target by modeling the movements of a human through the mathematical CM equation in order to relate the target's movement at the precise instant it happened to some similar motion in the past (Fig. 7). Table II shows the learning algorithm CM. For modeling the behavioral movement of the human target, we consider two curves, the first curve is

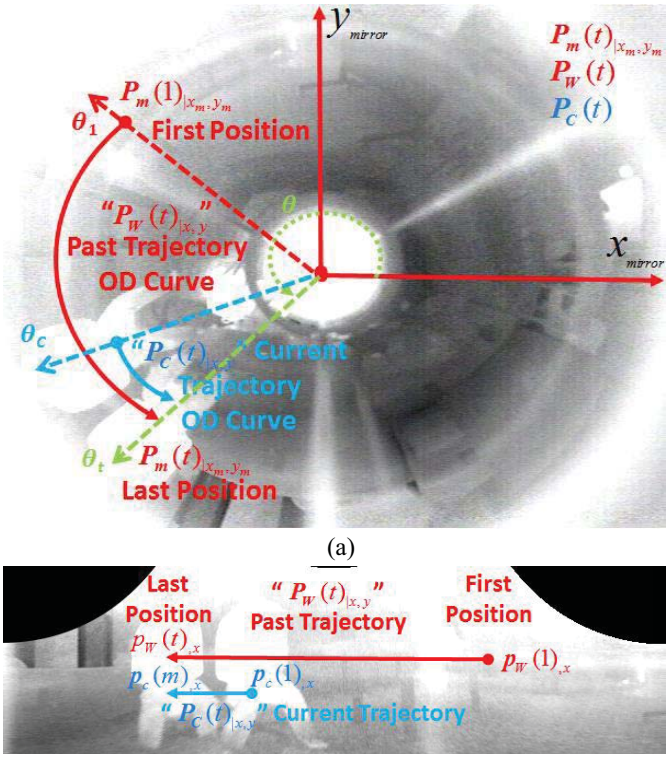


Fig. 5. Bearing tracking. (a) O-D angle trajectory curve of the target. (b) Unwrapped image trajectory of long term and current trajectory of the target.

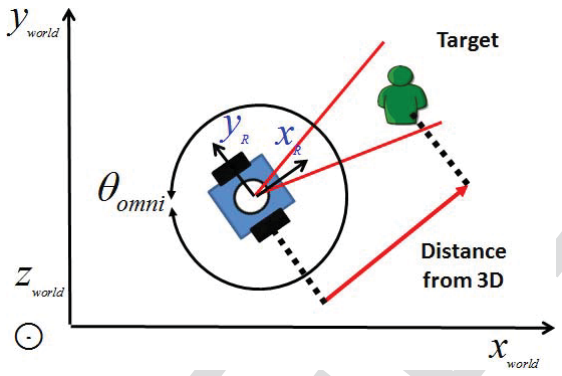


Fig. 6. Distance behavior analysis from 3-D reconstruction results.

used for the target's whole trajectory from the beginning of its motion or the target's past trajectory curve, and the second curve is used for the target's current trajectory or the target's current trajectory curve. The current target's curve in the trajectory represents the recent walking behavior of the human target. Similarity detection of a current target curve is searched in the target's whole trajectory. In order to use these trajectory curves for a similarity search, we constructed two trajectory strings from the curves, the current curve string $P_C(m)$ and the whole trajectory string $P_W(t)$, respectively. The matched part of the whole trajectory string, $P_W(t)$, with the current string, $P_C(m)$, provides the time length of similar movement characteristics of the target. The matched part of these strings are taken from the last position of the target, the last element of the

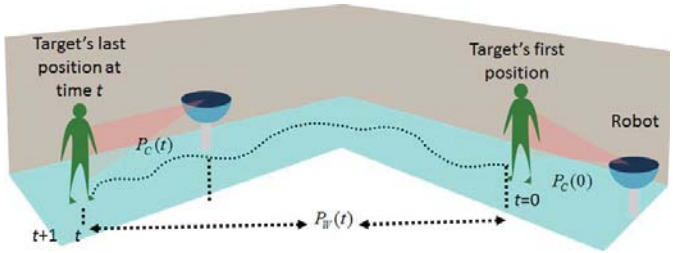


Fig. 7. CM for behavior analysis.

TABLE II
CM ALGORITHM

For human behavior modeling via curve fitting

$P_W(t)$ ← Past trajectories;

$P_C(m)$ ← Current trajectories;

Match ← Find $P_C(m)$ in $P_W(t)$;

Assign the best curve modeling to evaluate $P_C(m)$ in $P_W(t)$

Length ← Find the best length of the match;

Frequency ← Frequency of the match;

Learns the curve string sequences of the target behaviors for prediction.

string $P_W(t)$. The two empty strings are beginning to be filled with the target's positions from when we start the tracking mission of the robot. The target's entire trajectory is constructed as the past curve in string $P_W(t)$. The specific length of the entire trajectory string $P_W(t)$ is matched with the recent movement string $P_C(m)$, and this is considered a matched curve. If the CM algorithm cannot find any matched part in these strings, the current curve string $P_C(m)$ is restarted as an empty current curve string. The strings $P_W(t)$ and $P_C(m)$ are represented by

$$P_W(t) = \begin{bmatrix} p_W(1) & p_W(2) & \dots & p_W(t) \end{bmatrix}^T \quad (7)$$

$$P_C(m) = \begin{bmatrix} p_C(1) & p_C(2) & \dots & p_C(m) \end{bmatrix}^T. \quad (8)$$

The human target behavior-based modeling utilizes the targets recent walking behavior in the interested trajectory. The matched parts and the number of matches during this period models the target behavior to make a decision for the future position of the human target according to this walking behavior model. The weight of CM represents the recent behavioral walking of the human target in the trajectory. The position coordinates in the strings are searched and a weight for CM is determined by two criteria; the length of the match, and the frequency of the match in its history. The CM method uses the weight from this search with the given criteria and the equation for the weight of the CM framework is shown as follows:

$$\eta_{\mathrm{CM}} = k \frac{l_{C,\mathrm{matched}}}{l_{C,\mathrm{string}}} f\left(l_{C,\mathrm{string}}\right)\left(1 - \frac{1}{l_{C,\mathrm{string}}(s+1)}\right). \quad (9)$$

$l_{C,\mathrm{matched}}$ is the length of the exact matched part in the strings, $l_{C,\mathrm{string}}$ is the length of the current curve string, while

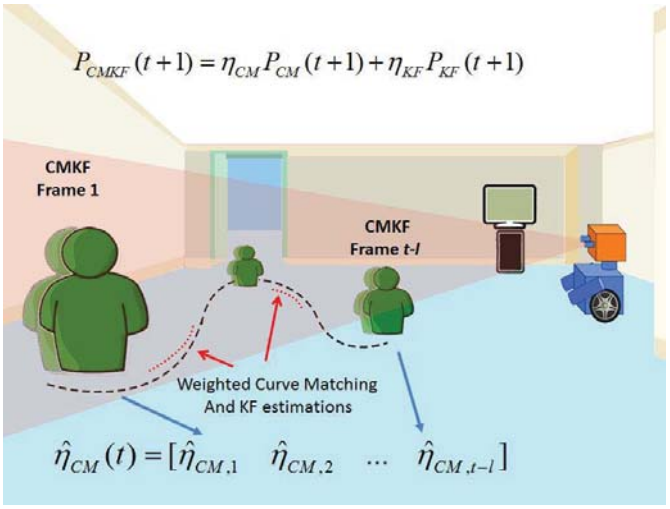


Fig. 8. CMKF estimation of adjusting each short term period tracking of the target.

f is the sigmoid function to keep the weight bounded, and the amount of matched strings is s. The final weight of the CM algorithm is obtained from (9).

The position of the target from the CM quadratic curve fitting and the weight of the CM is multiplied to find the next target's coordinates. The most recent target coordinates are used for tracking by CM and estimation of the next position using the CM weight.

B. Prediction for Short Term Tracking Using Curve Matched Kalman Filter

In step 2, the next position of the trajectories are estimated by the trained patterns in step 1. The learning space along with longitudinal analysis is now time-indexed as a possible expansion of the prediction window size. After obtaining the recent position of the target, the trajectory is analyzed for its previous path history to estimate the future position information of the target. This prediction is used to update the target's position which allows for the possibility to forecast the subsequent behavior of a human target from data sequences with time by comparing incoming online data sequences $[t, t+1]$ and the existing data sequence $[0, t]$. CMKF [3] is used in our proposed method as illustrated in Fig. 8. The KF method aims to track the motion of a target by accumulating the amount of CM states that depend on the extrapolation. Table III is representative of the modules on how the CM makes relation between the target's prompt movement and any related movement in the entire trajectory, and this makes an improved extrapolation for the KF framework over the time sequences. Module 1 calculates the probabilities of each method and module 2 forms the weights of CM as well as KF. Module 3 creates the matched curve from the target's trajectory and module 4 updates the matched curve probability. Module 5 estimates the next position information of the target. The flowchart of the CMKF is obtained through MAP by using n images for target positions and tracking from those reconstructed view can be seen in Fig. 9.

TABLE III
CMKF MODULES

Module1	Calculate the mixing probabilities.
Module2	Form the weighted average using the mixing probabilities.
Module3	Create CM state matched filtering.
Module4	Update CM state probability.
Module5	Estimate and covariance combination.

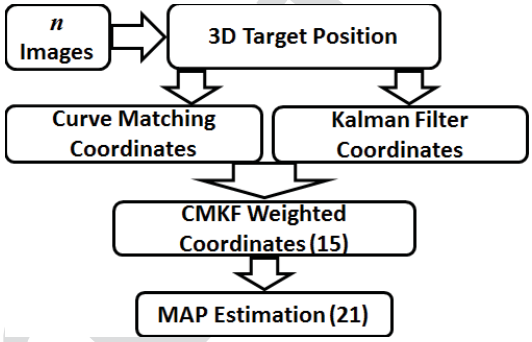


Fig. 9. Flowchart of CMKF via MAP.

KF predicts the next position at time $(t+1)$ for the system model with the following equations:

$$x_{t+1} = \begin{bmatrix} 1 & dt \\ 0 & 1 \end{bmatrix} x_t + \begin{bmatrix} dt^2/2 \\ dt \end{bmatrix} u_t + \varepsilon_t \quad (10)$$

$$y_t = \begin{bmatrix} 1 & 0 \end{bmatrix} x_t + v_t \quad (11)$$

where x_t is the velocity and the position vector at time t, u_t is the acceleration, and y_t is the measured position of the target. ε_t is the process noise and v_t is the measurement noise. If we define transition matrix $A = [1 \quad dt; \quad 0 \quad 1]$, input matrix $B = [dt^2/2; \quad dt]$, measurement matrix $J = [1 \quad 0]$ when the measurement of the position is done every dt seconds. KF equations K_t, the Kalman gain, $\hat{x}_t$, an estimation of next position can be given as

$$K_t = G_t J^T \left(J G_t J^T + W\right)^{-1} \quad (12)$$

$$\hat{x}_{t+1} = \left(A\hat{x}_t + Bu_t\right) + K_t\left(y_t + J\hat{u}_t\right) \quad (13)$$

$$G_{t+1} = G_t J^T (I - K_t J) G_t A^T + Q \quad (14)$$

where G_t is the estimation-error covariance for the system, W is the measurement noise covariance, and Q is the process noise covariance of the system. Exploiting the greater predictive power of the higher feature space, online data sequences effectively update the CM described in step 1. The update is used to identify the future curve parameters at time $t+1$, by minimizing the prediction error, called the normalized root mean squared error, between the predicted and actual position trajectories. The weight of KF for the curve matched KF process is calculated from the CM algorithm weight by $\eta_{\mathrm{KF}} = 1 - \eta_{\mathrm{CM}}$. The target coordinates are applied to KF and the next position of the target is obtained by (13). KF weight

η_{KF} is multiplied by the estimated KF coordinates to find the weighted KF coordinates for the short term tracking process

$$P_{\mathrm{CMKF}}(t+1) = \eta_{\mathrm{CM}} P_{\mathrm{CM}}(t+1) + \eta_{\mathrm{KF}} P_{\mathrm{KF}}(t+1). \quad (15)$$

After finding the accurate estimation of CMKF, the MAP is applied to the trajectory of the target. In order to find the next position of the targets, the MAP estimation will become more reliable and is then applied proportionally to the CM weight. This proportion is calculated from the length of the CMKF frame. Thus we can predict the lost data depending mostly on long term prediction than short term CMKF.

C. Prediction for Long Term Tracking Using Maximum a Posteriori Estimation

The MAP estimation method is a probabilistic estimation for the next target's position for their entire trajectory path, which we called long term tracking. CMKF prediction has an error problem and an increasing computational time problem, since this approach only considers the limited or short term trajectories. MAP estimation method helps to solve this error and time problems for long term trajectories. We would like to increase the horizontal time window by considering a longer window size.

The combination of MAP and CMKF is used in our proposed method to find the minimum error and the best estimation of the target's position. Next target position is first predicted by the KF method with respect to the CM for the last matched trajectory part. Then the MAP estimation is applied to the whole trajectory with all the matched trajectory curves in the past. Thus, the next position is predicted by a probabilistic estimation approach rather than solely searching for the last matched curve. The entire trajectory is represented by $P_W(t)$ and the characteristics, such as weights and probability, of CM for all CMKF frames. The CMKF frames are represented by $P_C(t)$ for every current trajectory in that instant with the prior of matched curves, $P_C(0)$. The entire and the current human target trajectories were substituted for all MAP estimation processes with the equation $\mathbb{P}(P_C(t)|P_W(t)) \propto \mathbb{P}(P_C(0))\mathbb{P}(P_W(t)|P_C(t))$. We want to maximize the left hand side to get the best weight of a matched curves in the entire trajectory $\arg\max_{P_C(t)} \mathbb{P}(P_C(t)|P_W(t))$. The right hand side of the equation is also maximized similarly $\arg\max_{P_C(t)} \mathbb{P}(P_C(0))\mathbb{P}(P_W(t)|P_C(t))$. The approximations to obtain the weight of CM for the next position of the target are calculated by (16), while we maximize the product of the entire trajectory and the current trajectory with the prior, $P_C(0)$, in the short term frame. The optimum frame length will be specified when the minimum error is satisfied by (17)

$$\mathbb{P}(P_C(t)|P_W(t)) = \mathbb{P}(P_C(0)) \prod_{i \in t} \mathbb{P}(P_W(i)|P_C(t)). \quad (16)$$

The size of the CM and the KF frame for short term tracking is determined by the prediction error when it is less than the predefined threshold value as the estimation process is initiated from the beginning. The time length of short term CMKF tracking frames is calculated by (17) in order to keep the short term frame on the most recent part of the trajectory.

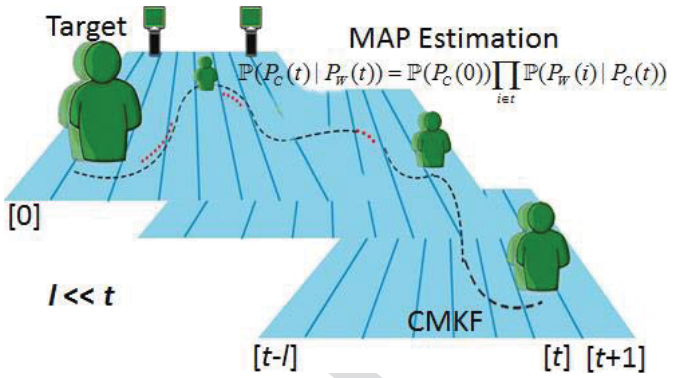


Fig. 10. MAP estimation scan the whole trajectory and CMKF frames moving through target trajectory to estimate next position for best weighted update for CM algorithm and KF.

The running average prediction error during the frame is calculated from the difference between the CMKF prediction and the actual position. This short term frame helps to keep the prediction error minimum while the prediction time is bounded

$$l = \underset{l_{\mathrm{frame}} \in [0,t]}{\arg\min} \frac{\sum_{l=t-l_{\mathrm{frame}}}^{t} (P_{\mathrm{CMKF}}(l) - P_w(l))}{l_{\mathrm{frame}}}. \quad (17)$$

Then the weights of each frame are stored from $t-l$ to the end of the last predicted target position corresponding to the sample of time t (Fig. 10). The stored weights of CM and KF aid in finding the posterior probability of the subsequent weight. Thus, in addition to the prediction of the following target position, we also predict the weight of the combination of two estimation methods one step prior to its application. The process of the moving CMKF frame along the trajectory and the MAP estimation applied on the whole trajectory is given in Fig. 10. The MAP estimation is given by the Bayes' rule equation while MAP estimates the best application of CM to the KF frame by maximizing $\mathbb{P}(P_C(t)|P_W(t)) = \mathbb{P}(P_C(0))\mathbb{P}(P_W(t)|P_C(t))/\mathbb{P}(P_W(t))$. The posterior equation involves prior of matched curves in the trajectory history, which is the first term of the numerator in equation, and the maximum likelihood, the second term of the numerator. The denominator is evidence for the change in weight of the whole trajectory.

The weight of CM, $\hat{\eta}_{\mathrm{CM}}(t)$, is updated according to the maximum result of the posterior probability by the following equation:

$$\begin{aligned} \hat{\eta}_{\mathrm{CM}}(t) &= \underset{P_C(t)}{\arg\max}\, \mathbb{P}(P_C(t)|P_W(t)) \\ &= \underset{P_C(t)}{\arg\max} \frac{\mathbb{P}(P_C(0))\mathbb{P}(P_W(t)|P_C(t))}{\mathbb{P}(P_W(t))}. \end{aligned} \quad (18)$$

Since the evidence portion is not dependent on $\hat{\eta}_{\mathrm{CM}}(t)$ as well as the result of the maximization process, we then can drop the denominator portion

$$\begin{aligned} \hat{\eta}_{\mathrm{CM}}(t) &= \underset{P_C(t)}{\arg\max}\, \mathbb{P}(P_C(0))\mathbb{P}(P_W(t)|P_C(t)) \\ &= \underset{P_C(t)}{\arg\max}\, \mathbb{P}(P_C(0)) \prod_{i=t-l}^{t} \mathbb{P}(P_W(i)|P_C(t)). \end{aligned} \quad (19)$$

The logarithm of the posteriors can be used to make the problem easier, so we can rearrange in the following form:

$$\hat{\eta}_{\mathrm{CM}}(t) = \arg\max_{P_C(t)} \left(\sum_{i=t-l}^{t} \log \mathbb{P}(P_W(i)|P_C(t)) + \log \mathbb{P}(P_C(0)) \right). \tag{20}$$

Predictive target positioning may sync with the position of moving target, while minimizing the time delay of tracing the target. Long term tracking and reliable target following are desired to maintain an acceptable prediction accuracy so that the feature space can dynamically adjust to changes in the learning phase for the duration of the longitudinal sequence. The weight of CM is adjusted by MAP for long term tracking results and is applied to CM and KF separately by using (21). The estimated coordinates, $P_{\mathrm{MAP}}(t+1)$, of the target is obtained more precisely

$$P_{\mathrm{MAP}}(t+1) = \hat{\eta}_{\mathrm{CM}}(t) P_{\mathrm{CM}}(t+1) + \left(1 - \hat{\eta}_{\mathrm{CM}}(t)\right) P_{\mathrm{KF}}(t+1). \tag{21}$$

We propose a simultaneously iterative learning and prediction for consistent field testing. The prediction criterion for simultaneous learning provides the error variations for the time horizon window of the data sequences. We determine the applicable prediction range for the long term data that will minimize the prediction error. The horizontal prediction window will be iteratively applied to the updated human behavior (described in Section IV-A) for the incoming online sequences for predicting the future positioning of robot trajectories (described in Section IV-B).

D. Criteria to Follow Human Target From Mobile Robot

Target motion analysis helps to extract the target's pixels in the image sequence for the reconstruction of the target view. The robot's movement will cause the positions of feature points to change all along the images as well as the pixels in each image. In order to track the target and the distance information of the target, target's feature points and feature points of the other objects must to be sorted. We will use three criteria to obtain more detailed information about the target and identify the target in an O-D IR view. Table IV shows the procedures of the target's tracking steps.

Distance tracking is one of three criteria needed in order to track the target's trajectory. The coordinates of the target provides the depth information of the target. The current target's position, P_W, and robot's current position, P_R, provide the actual distance from the target. X_T, the location of the target and, X_R, the location of the robot are used for the distance in x, y axes, $|X_{T,R}| = |P_{W-R}| = \sqrt{(x_t - x_r)^2 - (y_t - y_r)^2}$. The recent coordinates of the target are used for inputting the distance and is then tracked by MAP estimation (Fig. 10). Predicted and actual distances between robot and target are used for validation of the prediction for target distance tracking.

The second criterion is angle tracking of the targets such that the direction of the robot is based on the targets. The angle of the target, θ_T, in the O-D image provides a value from 0° to 360° of the target position, $\Delta\theta_T = \theta_t - \theta_0$. Those angle values are tracked and then are predicted in order to find the minimum error of V_θ, angular velocity of the target, for prediction steps (Fig. 11). Angular velocity for that instant can be calculated from the angle position difference of the first and last position of the target, $V_\theta = \Delta\theta_T/\Delta t = (\theta_t - \theta_0)/(t - t_0)$.

The third criteria is the velocity of the targets from the predicted positions. Fig. 11 shows the target velocity, V_T, calculated by using the actual position of the target at the first and at its final movement moment, $|X_{t,0}| = |P_{W,t-0}| = \sqrt{(x_t - x_0)^2 - (y_t - y_0)^2}$. Predicting the target's velocity will provide an acceptable way to approach the target and having the robot follow the target while simultaneously adjusting the robot's velocity to track its movement, $V_T = \Delta X_T/\Delta t = (X_t - X_0)/(t - t_0)$.

TABLE IV
CRITERIA OF TARGET TRACKING

For target motion variables

$P_R(t) = [\mathrm{x}_r\ \mathrm{y}_r\ \mathrm{z}_r]$, $\theta_R \leftarrow$ Robot's motion;

$P_W(t) = [\mathrm{x}_t\ \mathrm{y}_t\ \mathrm{z}_t]^T$, $\theta_T \leftarrow$ Target's motion;

Extraction$\leftarrow$ Find P_W in P_R;

Assign the best target 3D position P_W ;

Evaluate target's motion characteristics;

$\min\left(\left|X_{T,R}\right|\right) = \min\left(\left|P_{W-R}\right|\right) \leftarrow$ Find the distance of the target;

$V_\theta = \Delta\theta_T/\Delta t = \left(\theta_t - \theta_0\right)/\left(t - t_0\right) \leftarrow$ Find the angular velocity of the target;

$V_T = \Delta X_T/\Delta t = \left(X_t - X_0\right)/\left(t - t_0\right) \leftarrow$ Find the velocity of the target;

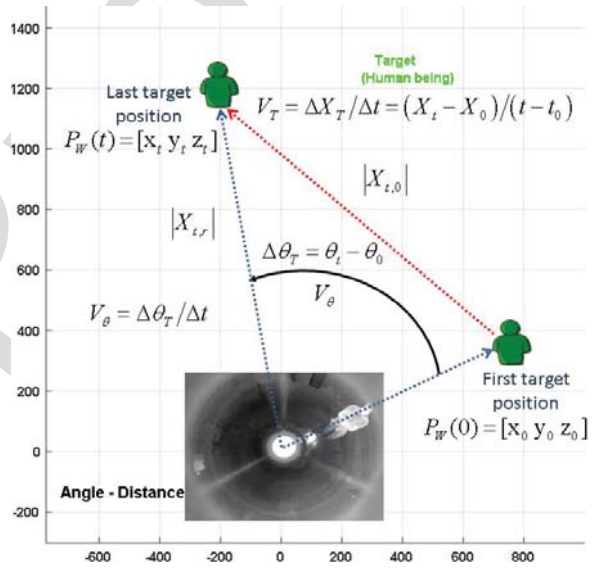


Fig. 11. Velocity change criteria of angle and target position during the target motion.

V. Experiments

The experiment section is organized in the form of six sections. First, in Section V-A, behavior analysis and hardware information with the data sets are used for tracking is given. Second, prediction for short term target tracking is applied and compared with respect to the time length of the prediction period in Section V-B. Third, prediction for long term target tracking using the MAP estimation method is given in Section V-C. Then, target motion analysis for evaluating various human target behaviors is in Section V-D. And, an analytical comparison is done to other studies in Section V-E.

A. Behavior Analysis and Hardware

The hardware for target tracking consists of a mobile robot platform with three main components. First component is the mobile robot Pioneer 3-DX, equipped with a PC for autonomous control. Second component is a Windows 7 computer with an Intel i7 processor used to perform image processing and tracking algorithms with MATLAB. Third component is the RemoteReality O-D IR camera sensor for the acquisition of 360° O-D IR images with 65° of vertical field of view (FOV). The O-D IR sensor provides gray level images with a resolution of 640×480 pixels and the images are taken by a 30-Hz imaging frequency of an O-D IR sensor. The mobile robot platform equipped with a windows computer and an O-D IR sensor is shown in Fig. 1 in Section I. The O-D IR camera was calibrated by the O-D camera calibration toolbox [51] using a heated grid pattern since the calibration toolbox was only created for visible-band cameras. The grid pattern is constructed by a silver aluminum tape in order to utilize heat reflection, dependent specifically on the temperature of the tracked objects. The detection of feature points and human target in the images was done by corner detection in MATLAB's computer vision toolbox with respect to Eigen values of the feature points.

The imaging conditions included have four different types: 1) short term; 2) long term; 3) slow move; and 4) fast move. As shown in Fig. 12, the first data set was taken for short term tracking, 20 images, in a relatively small area and was over a short distance. The second data set had a longer time observation with a slow moving target, with 475 images, in a larger area. The third image sequence was also taken in a large area, about 80 m^2, with a short term recording period for a faster moving target. The last data set includes a fast target with long term tracking purposes, 384 images. The characteristics of the imaging condition for the data sets of short term, and long term along with slow and fast movement, can be seen from Table V. The sampling of the video recording time was set to five images per second.

The CM method for trajectory estimation was used to improve the tracking error of a human target by a mobile robot. The O-D IR images provide additional information to select the human regions based on a thermal signature of the human body. The pixel values corresponding to the human body temperature are specified in order to make a decision for the detection threshold of the human body regions. The feature points of the human target were calculated by the SURFfeature

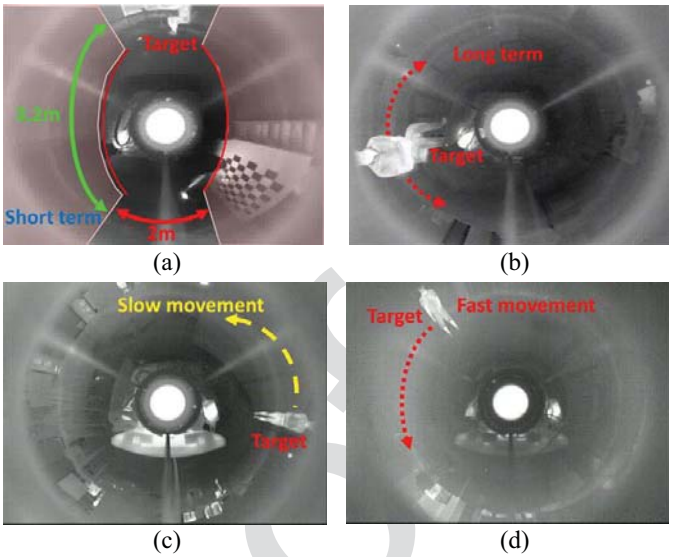


Fig. 12. Representative imaging conditions. (a) Short term. (b) Long term. (c) Slow movement. (d) Fast movement.

TABLE V
Imaging Conditions

Imaging	Covered Area (m^2)	Total Frame Size (images)	Average Target Velocity (m/sec)
Short Term	6.4	20	1.8445
Long Term	80	475	1.432
Fast Movement	80	384	1.867

MATLAB function in the unwrapped O-D images. Those images were also converted to binary images from a gray level image, based on the thermal signature of a human, to get the human target's position information rather than the coordinates of the noisy feature points in the background. In order to avoid losing the human target during the tracking process, the five strongest features of the target are selected. Those feature points were detected with respected to the difference in two consecutive images. The detection process for the strongest feature points was applied for the moving human target region. Thus, we avoid the detection of the objects which have similar thermal signatures to the human body while the static objects were ignored from the scene. In case of losing any of the selected features, the remaining feature points overcome the misdetection problem of the target's position. Since our method is based on the human motion behavior, any unexpected error or mistracking is predicted correctly in the next process. For evaluation of our method, we selected only one target which is closest and giving better features for detection. The first detected and tracked features of the corresponding target is considered as the main target. However, we can apply our method to multiple targets after detection and proper identifying each target in the O-D images. The position of the target in the frames 7, 225, and 428, while it is moving is shown in Fig. 13. The target's positions from every image frame are given on each image with the corresponding color to show the entire trajectory of the target. The color bar

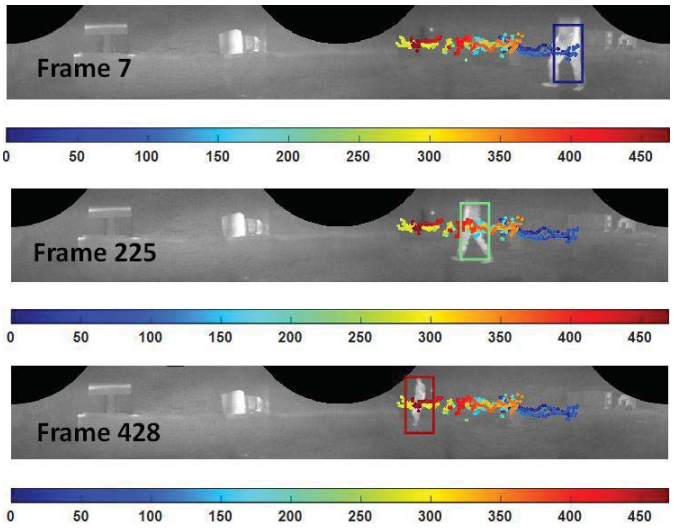


Fig. 13. O-D data set used for tracking in unwrapped (panoramic) O-D image format. The detected target is shown in frames 7, 225, and 428. The color shows the trajectory of the target according to frame number in the color bar.

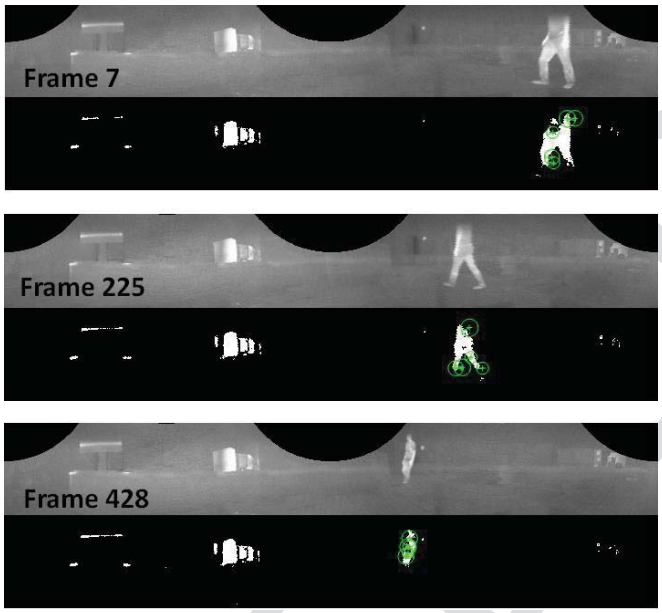


Fig. 14. Original unwrapped images from frames 7, 225, and 428. Detected target features in the binary unwrapped O-D images during the targets movement.

represents the corresponding color of the specific frame number. Dark blue represents the first position of the target while the dark red is the last target position in the trajectory. In order to detect these target positions, the moving human target has been detected in the original unwrapped O-D images, shown in Fig. 14. The strongest feature points of those images are also shown by green circles on the binary image in Fig. 14 for frames 7, 225, and 428 from the image sequence. The human region was detected by utilizing the thermal signature of the human body. Objects with the similar temperature were also eliminated by applying the detection algorithm on the regions that have changes from a previous frame. Table VI shows the data set information in greater detail.

TABLE VI
CAPTURED DATA SETS

Imaging	Covered Area (degree/m^2)	Total Frame Size (images)	Average Target Velocity (m/sec)
Omni-directional Data Set	360 / 80	475	2.102
Perspective Data Set	60 / 13.4	400	1.432

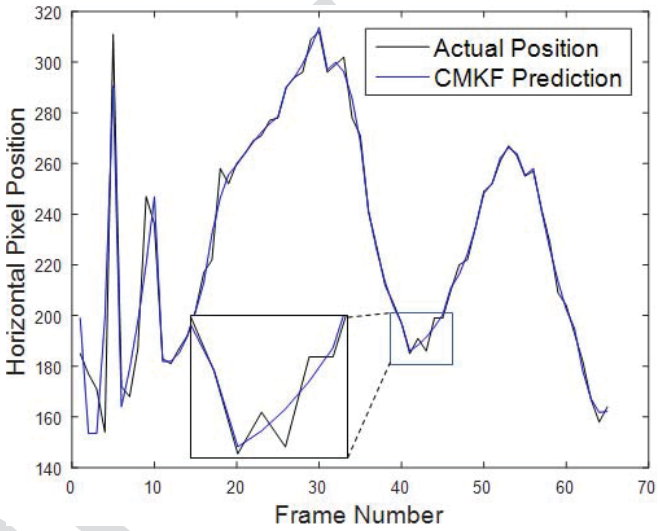


Fig. 15. Trajectory prediction from CMKF for O-D IR data set, short term data set.

The coordinates of the target in each images of the data set was calculated by the strongest feature points and those coordinates for the image sequence were used to estimate the trajectory of the target while in motion. Those detected target coordinates in the images were considered as the ground truth of the target image position. After the prediction process, the prediction result and those detected target positions were used to calculate the prediction error. The ground truth of the target was done by physical measurements from the target to the robot for the evaluation of our method. Linear regression based on the CM algorithm was applied to those coordinates and the estimated curve was obtained as the result. The actual data is given by a black line in Fig. 15 and estimated values of this position are given by a blue line as the CMKF prediction in Fig. 15. The position of the target is given in the horizontal pixel coordinates of the human target in the unwrapped O-D images. During a specific term, the target's current curve matches a prior curve as seen from the error between an actual and an estimated curve. The result was the motion of the target includes the position with respect to beginning coordinates, this portion of the curve was predicted accurately as we can see in Fig. 15. The computational time for estimating the subsequent target coordinates using CMKF is given in Fig. 17.

The result was the motion of the target including the position's angle with respect to the beginning coordinates, this portion of the curve was predicted accurately as we can see in Fig. 15. CMKF had an average error of 0.7357 pixel with respect to the position information on the trajectory path.

TABLE VII
TRACKING ERROR USING CMKF

Imaging Dataset	Angular Velocity (degree/sec)	Angle Error (degree)	Target Velocity Error (m/sec)	Prediction Position Error (pixel)
Short Term	2.4217	0.3905	0.5772	1.0149
Long Term	1.4794	0.1891	0.0474	0.7357
Fast Movement	1.9591	0.2232	0.0924	0.8682

This error was tolerable for the targets that have nonlinear motion characteristics. However, computational time was getting higher for the later portion of the trajectory. The computational time for estimating the next target coordinates by CMKF is given in Table IX.

B. Prediction for Short Term Target Tracking

In this section, prediction for the tracking method was applied to short and long periods with a slow as well as fast target images from the O-D IR camera. First, the robot was driven in a laboratory condition to take images of the target with a slow target imaging type setting. Then, the robot recording for fast movement of the target for long and short periods aimed at tracking the target in a larger area as a more complex data set.

The tracking results of the target from short and long term data sets is compared and the average prediction errors are given in Table VII. When the target was moving very fast, the prediction error was increasing. The best result was obtained from the slow movement with a long term data set, 0.7357 pixel for the CMKF tracking method. However, the prediction error remained stable for much longer times. The average prediction error increased to 1.0149 pixel for the target's short term observation with an actual speed of 1.8445 m/s shown in Table V and a predicted speed of 2.4217 m/s, given in Table VII. The average prediction error provided a moderate value of 0.8682 pixel for the fast moving target of the long term data set from the observed images. The prediction of the target's velocity was obtained as 1.9591 m/s with an acceptable velocity error of 0.0924 m/s while the target moving fast for a long period of time.

The KF method was applied to the same data set without the CM algorithm. The results of the KF method provided a higher prediction error than the CMKF method with a stable average for computational time. Also, the standard deviation of error was high since the KF method contributed significantly higher prediction error. CMKF had around 98 percent better standard deviation, when the nonlinearity of the target's movement was high. Computational time was stable and higher for KF, but CMKF computational time was increasing as the tracking duration was increasing. CMKF gave improved prediction results with around 25% lower prediction error than the KF method.

C. Prediction for Long Term Target Tracking

We used three different data sets, short term, long term, and fast movement, and they were compared to find the minimum prediction error for the tracking the target in a large area and small area. First, short term tracking used the MAP estimation-based prediction method and the tracking time has decreased; for long term tracking by applying the MAP estimation to see the differences for those two different tracking results. The data sets with different target velocities were utilized for the long term tracking process to obtain the minimum prediction error. The short term data set with the fast target movement in a 6.4 m^2 area while the long term data set area with slow and fast target movement in a 80 m^2 area were used, given in Fig. 12, respectively.

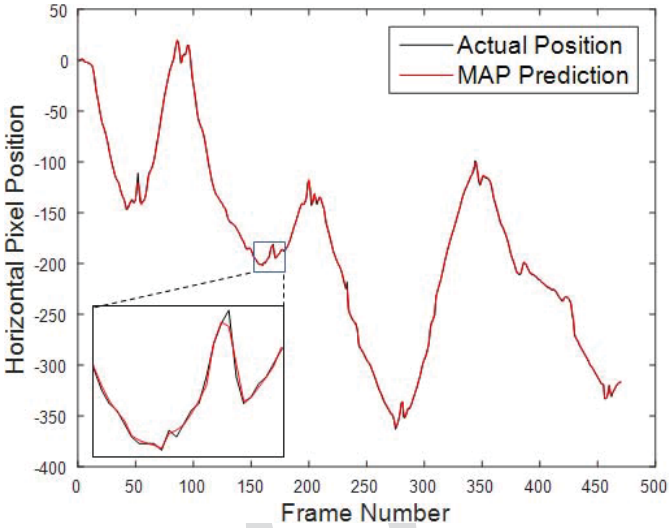


Fig. 16. Trajectory prediction from MAP with O-D IR data set, long term data set.

CMKF increased the prediction accuracy for next target position with respect to KF by around 60%, however, the computational time was increasing continuously and also after a specific time period prediction error remained stable. MAP estimation made it possible to adjust the weights of CM and KF along with the predicted weight from MAP, the next target position was calculated with higher accuracy and faster computational time.

MAP estimation predicted the next target position calculated with 10.48% higher accuracy and faster (Fig. 16). The human behavior-based method predicted the possible unexpected values from the sudden movement of the target and separates them from the human's walking behavior in the long term trajectory. Those sudden movements of the human and possible changes in the target's movement direction in the trajectory are given in the zoomed window in detail. We used the actual position of the target from the detected image points in the data set images. The given input, actual position, and the predicted position in the next image frame are used to calculate the prediction error. It can be seen that our method predicted a closer target position to a possible human movement. The prediction difference from the CMKF prediction can be seen from Tables VII and VIII.

In the fast target case the average prediction error was 0.1723 pixel higher than the slow target case. Thus, this higher prediction error increased the average tracking error

TABLE VIII
TRACKING ERROR MAP ESTIMATION

Imaging Dataset	Angular Velocity (degree/sec)	Angle Error (degree)	Target Velocity Error (m/sec)	Prediction Position Error (pixel)
Short Term	2.0869	0.1993	0.2425	0.7075
Long Term	1.4198	0.1693	0.0122	0.6586
Fast Movement	1.8934	0.2136	0.0268	0.8309

TABLE IX
COMPUTATIONAL COMPLEXITY OF PREDICTION

Duration	Slow Movement Time (ms)	Fast Movement Time (ms)	Average Prediction Time (ms)
CMKF	0.1767	0.1467	0.1617
MAP	0.0974	0.0979	0.0976
Difference (%)	44.88	33.26	39.64

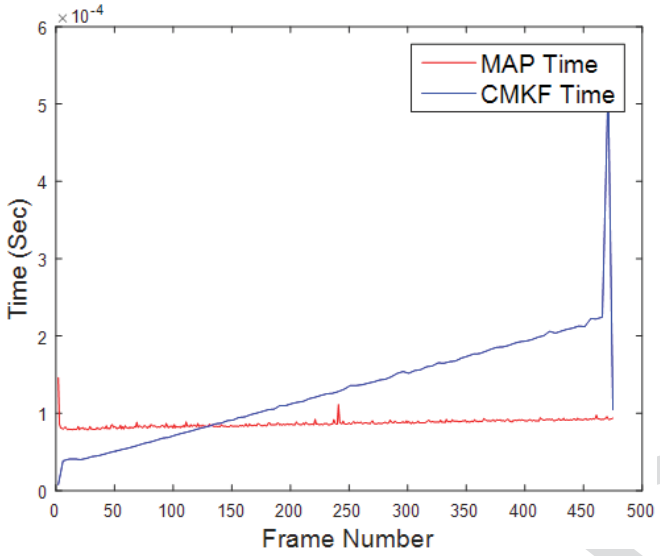


Fig. 17. Computational time for long term tracking by using MAP estimation (red) and CMKF (blue).

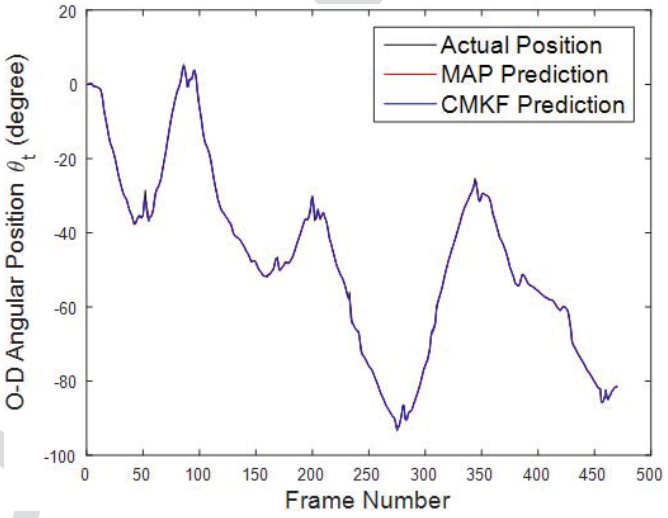


Fig. 18. Angular position θ_t of the target by CMKF and MAP estimation.

over 20.13% for the fast movement data set, since the target's trajectory had more sudden moves and changes than the slow movement data set's trajectory. The best imaging conditions were obtained from the long term data set with the slow movement of the target, 1.432 m/s shown in Table V and Fig. 12(c). The lowest prediction error for the long term data set by the MAP prediction case with 100 to 500 images by 0.6586 average value.

The computational time for short term and long term tracking of different data sets are given by Fig. 17. The time was considered to track the target's coordinates from each image. Fig. 17 gives the slow movement data set's computational time along with the short term method, (blue), and long term method (red) of computational times. The computational time curve shows that our proposed MAP-based method took longer to predict the target's position for the short time tracking, up to the 127th frame. The computational time for CMKF increased and passed the time spent in our method when the frame number is higher than 127. The computational time of those reported periods were increasing almost linearly for the CMKF method. We can see that the time for CMKF nearly doubled the time consumed from the 200th frame to the 400th frame. In case of multiple target tracking, our method can be applied to each target separately after identification of each target properly. The number of targets will increase the computational time similar to other methods. The second target doubles the computational time; however, our method provides sufficient time to track multiple targets since the computational time is low enough to finalize the process. Our proposed method gave a computational time approximately stable for the same time period. We used the advantage of lower prediction time from CMKF for a shorter term tracking and kept the prediction time steady. The use of CMKF in short term tracking windows as part of the MAP-based continuous prediction maintains the computational time with a minimum increase.

Long term tracking by MAP estimation process gave the best value for tracking with the CMKF frame length of 20 images and this value was decided to be the optimal length for a CMKF subset. The CMKF prediction accuracy remained stable but the computational time was still increasing after the 20th image. We limited the CMKF short term tracking window at this point and decided the final prediction by the MAP prediction to increase the accuracy and decrease the computational time.

The average computational time for a short term CMKF frame was 0.1617 ms, and 0.0976 ms for long term tracking by MAP estimation given in Table IX. Our proposed method gave 39.64% lower computational time for the prediction of the target's next position with higher accuracy. The MAP estimation method has an advantage of a lower computational time while the computational time of the CMKF method is increasing around 50% for every additional 200 images. The prediction time will be much higher in case of using

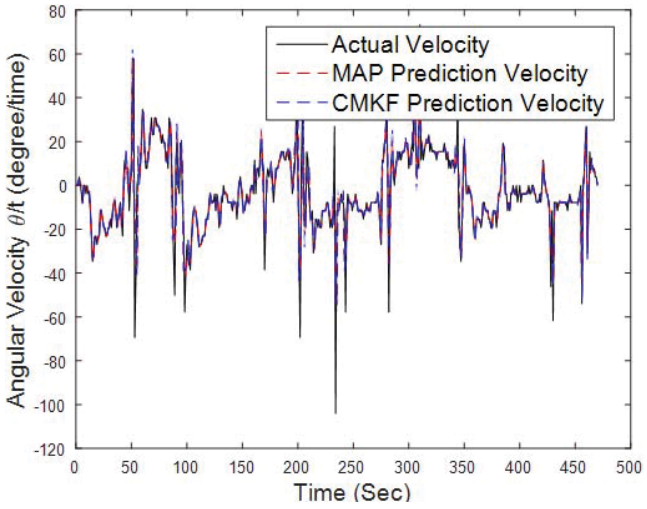


Fig. 19. Angular velocity $\dot{\theta}_t$ of target by CMKF and MAP estimation.

TABLE X
EVALUATION OF FOLLOW HUMAN TARGETS

Motion	Average Prediction Error (Fast)	Average Prediction Error (Slow)	Average Prediction Time (ms) (Fast/Slow)
Distance (m)	0.8309	0.6586	0.0974 / 0.0979
Angle (degree)	0.2136	0.1693	0.0974 / 0.0979
Velocity (m/sec)	0.0268	0.0122	0.1298 / 0.1305

TABLE XI
COMPARISON WITH OTHER METHODS

Methods	Average Prediction Error (pixel)	Velocity Prediction Error (m/sec)	Standard Deviation of Error (pixel)	Average Prediction Time (ms)
Proposed MAP based tracking	**0.6586**	**0.0938**	**2.2874**	**0.0976**
MAP Bank [1]	N/A	9.9527	N/A	0.3160
BVT [5]	35.620	N/A	N/A	125.0
CMKF [3]	0.7357	0.2390	0.0662	0.1617
EKF [3]	1.0121	N/A	5.2748	0.4255

only CMKF prediction for continuous human target tracking. Fig. 17 shows that the computational time of CMKF was 2.5 times higher than our method at the 450th frame, the linear increase shows that the computational time difference between two methods will increase for a higher number of frames. For a longer tracking process, our method delivers increasing performance.

D. Evaluation of Various Human Target Behaviors

Distance tracking is applied as the first step. The O-D view aids the depth acquisition for any orientation of the target with respect to the robot's position. The most recent coordinates of the target are used for inputting the distance tracking process with the depth value, $|X_{t,r}|$, of those coordinates.

Angle tracking the target is the second step of the process. The angle of the target, θ_t, is derived from the detected target position in the O-D image. The O-D image provides from a 0°

TABLE XII
APPENDIX

Symbol	Definition
$W_P = [X \quad Y \quad Z]^T$	3D Real world point in space
$p = [u \quad v]^T$	2D image point on O-D image
$[u_0 \quad v_0]$	Principle point of the camera
L	Distance between mirror center and projection center
R_m	Radius of the mirror
$P_m = [x_m \quad y_m \quad z_m]^T$	Reflected mirror coordinates of a world point
$P_{m,1} = [x_{m,1} \quad y_{m,1} \quad z_{m,1}]^T$	Ray vector of the mirror coordinates from first observation point
$P_{m,2} = [x_{m,2} \quad y_{m,2} \quad z_{m,2}]^T$	Ray vector of the mirror coordinates from first observation point
$r_m = \sqrt{x_m^{\,2} + y_m^{\,2}}$	Radial distance of the mirror point to optical axis
$r = \sqrt{X^2 + Y^2}$	Radial distance of a real world point to optical axis
f_u, f_v	Focal lengths of the O-D camera
β	Angle between the rays from the projection center to principle point and image point
o	Angle of the image point on the image *x, y* coordinates
α	Angle between the vertical axis and the ray vector from the world point to the mirror surface
$\gamma_i,\ \gamma_r$	Reflection angle on the mirror
R	3x3 Rotation matrix
T	3x1 Translation vector
z_1, z_2	Calculated target distance from the first and second observation point
$p_W = [x \quad y \quad z]^T$	Triangulation result, target's position.
$P_W(t)$	Entire positions of target in trajectory.
$P_C(m)$	Current trajectory of the target in the time length of m
$P_C(0)$	Prior of matched curve
$\eta_{CM},\ \eta_{KF},\ \hat{\eta}_{CM}(t)$	Weight of CM, KF, and MAP-based CM weight
$P_{CMKF},\ P_{CM},\ P_{KF},\ P_{MAP}$	Predicted positions by CMKF, CM, KF, and MAP
$l_{C,string},\ l_{C,matched}$	Length of entire trajectory and matched curve
$l,\ l_{frame}$	Final and test length of CMKF frame
$k,\ s,\ f$	Constant to adjust CM weight, amount of match string, and sigmoid function
K_t	Kalman gain
J, W	Measurement matrix and noise covariance
A	Transition matrix
B	Input matrix
G_t	Estimation error covariance
Q	Measurement noise covariance
$u_t,\ v_t$	Acceleration, measurement noise
$\hat{x}_t$	Estimation of KF
$P_R,\ X_R$	Robot's coordinates and 2D location
V_T, V_θ	Target's velocity and angular velocity
θ_R, θ_T	Angle of robot and target.

to a 360° angle position of the target and those angle values are stored for the prediction steps (Fig. 18).

The actual velocity of the targets was calculated from the target's stored positions during the movement of the target. The predicted Target velocities from the CMKF and the MAP methods showed that the MAP had predicted the target's velocity with 71.0% lower error than the CMKF method, shown in Tables VII and VIII and Fig. 19. Tracking of the angle provides improved tracking results of 10.48% compared to the short term tracking results (Fig. 18). Table X shows the tracking error of the distance, the angle and angular velocity of the target, $\dot{\theta}_t$ in terms of slow or fast moving target. For the slow target, the velocity prediction provided 54.47% lower error than the fast target.

E. Comparison to Other Studies

Comparison between the MAP and the other four tracking methods is given by the average prediction error, the velocity prediction error, the standard deviation of error, and the average prediction time for this section. The benefits of the O-D IR sensor are advantageous, but does have lower accuracy as well as a higher prediction time problem that makes tracking difficult with the visual band and perspective camera-based methods. We overcame these problems to make tracking work better with the proposed method. The comparison between the proposed MAP and the other four methods: a bank of MAP [1], the backward model validation-based visual tracking (BVT) [5], the CMKF [3], and the EKF method is shown in Table XI. These methods are calculated with respect to the average prediction error, the velocity prediction error, the standard deviation of the error, and the average prediction time.

The results of all five methods indicated that our proposed method improved the prediction of the next target position. The standard deviation of error has improved by 56.63% with respect to the EKF method. The velocity prediction error for our method was 60.75% lower than CMKF prediction the target velocity and significantly lower than velocity error of a bank of MAP. The standard deviation of error was higher than CMKF's standard deviation of error; however, our proposed method provided a 39.64% faster computational speed than the CMKF method. Also, our proposed method gives an additional computational time advantage for a considerably longer tracking processes. The average prediction time of our MAP-based tracking method was lower and was significantly faster than the average prediction time of the BVT method. The long term behavior analysis prediction error was lower than both CMKF and EKF methods while it outperformed BVT in terms of average prediction error.

VI. Conclusion

We proposed behavior learning and MAP estimation-based target tracking methods to improve the tracking accuracy and to reduce the computational time. An O-D IR camera was used to maximize the target tracking timeframe with the freedom of lighting and to extend the tracking view by 360° FOV of O-D sensor. The disadvantages of the other methods for long term tracking, low accuracy and high computational time, was solved by our proposed MAP-based target tracking method by using an O-D IR camera and utilizing human behavior and human body features. We evaluated our method by tracking only one target from one mobile robot and this method can be used for multiple robots with proper identification of each target.

For our future work, we plan to utilize the MAP-based tracking with multiple human targets by increasing the number of mobile robots. The low computational time of our method will provide more sufficient time for multiple target tracking. Employment of multiple robots will be used for predicting the multiple targets' positions to improve the tracking performance of the mobile robots.

Acknowledgment

The authors would like to thank F. Fanary for the help to improve this paper by proofreading.

References

[1] G. Huang, K. Zhou, N. Trawny, and S. I. Roumeliotis, "A bank of maximum a posteriori (MAP) estimators for target tracking," *IEEE Trans. Robot.*, vol. 31, no. 1, pp. 85–103, Feb. 2015.

[2] W. Choi, C. Pantofaru, and S. Savarese, "A general framework for tracking multiple people from a moving camera," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 7, pp. 1577–1591, Jul. 2013.

[3] S. J. Lee, G. Shah, A. A. Bhattacharya, and Y. Motai, "Human tracking with an infrared camera using a curve matching framework," *EURASIP J. Adv. Signal Process.*, vol. 2012, no. 1, pp. 1–15, 2012.

[4] K. Bai, Y. Wang, Y. Yan, and Q. Song, "Infrared small target tracking based on target and interference behaviors model," *Infrared Phys. Technol.*, vol. 67, pp. 256–265, Nov. 2014.

[5] Y. Yuan, S. Emmanuel, Y. Fang, and W. Lin, "Visual object tracking based on backward model validation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 11, pp. 1898–1910, Nov. 2014.

[6] R. Liu and X. Zhang, "Understanding human behaviors with an object functional role perspective for robotics," *IEEE Trans. Cogn. Develop. Syst.*, vol. 8, no. 2, pp. 115–127, Jun. 2016.

[7] I. Mikić, M. Trivedi, E. Hunter, and P. Cosman, "Human body model acquisition and tracking using voxel data," *Int. J. Comput. Vis.*, vol. 53, no. 3, pp. 199–223, 2003.

[8] M. Yeasin and S. Chaudhuri, "Toward automatic robot programming: Learning human skill from visual data," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 30, no. 1, pp. 180–185, Feb. 2000.

[9] S. Jung and K. Wohn, "Tracking and motion estimation of the articulated object: A hierarchical Kalman filter approach," *Real Time Imag.*, vol. 3, no. 6, pp. 415–432, 1997.

[10] N. Bellotto and H. Hu, "Multisensor-based human detection and tracking for mobile service robots," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 39, no. 1, pp. 167–181, Feb. 2009.

[11] D.-S. Jang, S.-W. Jang, and H.-I. Choi, "2D human body tracking with structural Kalman filter," *Pattern Recognit.*, vol. 35, no. 10, pp. 2041–2049, 2002.

[12] M. Yeasin and S. Chaudhuri, "Development of an automated image processing system for kinematic analysis of human gait," *Real Time Imag.*, vol. 6, no. 1, pp. 55–67, 2000.

[13] A. J. B. Trevor, J. G. Rogers, and H. I. Christensen, "Planar surface SLAM with 3D and 2D sensors," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2012, pp. 3041–3048.

[14] F. Ferland and F. Michaud, "Selective attention by perceptual filtering in a robot control architecture," *IEEE Trans. Cogn. Develop. Syst.*, vol. 8, no. 4, pp. 256–270, Dec. 2016.

[15] X. Yun and E. R. Bachmann, "Design, Implementation, and experimental results of a quaternion-based Kalman filter for human body motion tracking," *IEEE Trans. Robot.*, vol. 22, no. 6, pp. 1216–1227, Dec. 2006.

[16] A. Zaraki *et al.*, "Design and evaluation of a unique social perception system for human-robot interaction," *IEEE Trans. Cogn. Develop. Syst.*, to be published.

[17] M.-T. Yang, S.-C. Wang, and Y.-Y. Lin, "A multimodal fusion system for people detection and tracking," *Int. J. Imag. Syst. Technol.*, vol. 15, no. 2, pp. 131–142, 2005.
[18] C. Bauckhage, J. K. Tsotsos, and F. E. Bunn, "Automatic detection of abnormal gait," *Image Vis. Comput.*, vol. 27, nos. 1–2, pp. 108–115, 2009.
[19] X. Zhou and B. Bhanu, "Integrating face and gait for human recognition at a distance in video," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 37, no. 5, pp. 1119–1137, Oct. 2007.
[20] P. S. Huang, "Automatic gait recognition via statistical approaches for extended template features," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 31, no. 5, pp. 818–824, Oct. 2001.
[21] D. K. Vishwakarma and K. Singh, "Human activity recognition based on spatial distribution of gradients at sub-levels of average energy silhouette images," *IEEE Trans. Cogn. Develop. Syst.*, to be published.
[22] L. Wang, T. Tan, H. Ning, and W. Hu, "Silhouette analysis-based gait recognition for human identification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 12, pp. 1505–1518, Dec. 2003.
[23] F. Lerasle, G. Rives, and M. Dhome, "Tracking of human limbs by multiocular vision," *Comput. Vis. Image Understanding*, vol. 75, no. 3, pp. 229–246, 1999.
[24] J. G. Rogers, A. J. B. Trevor, C. Nieto-Granda, and H. I. Christensen, "Simultaneous localization and mapping with learned object recognition and semantic data association," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, 2011, pp. 1264–1270.
[25] P. Weckesser and R. Dillmann, "Modeling unknown environments with a mobile robot," *Robot. Auton. Syst.*, vol. 23, no. 4, pp. 293–300, 1998.
[26] C. Hu, F. Arvin, C. Xiong, and S. Yue, "A bio-inspired embedded vision system for autonomous micro-robots: The LGMD case," *IEEE Trans. Cogn. Develop. Syst.*, to be published.
[27] D. W. Paglieroni, G. E. Ford, and E. M. Tsujimoto, "The position-orientation masking approach to parametric search for template matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 7, pp. 740–747, Jul. 1994.
[28] Y.-P. Wang, S. L. Lee, and K. Toraichi, "Multiscale curvature-based shape representation using B-spline wavelets," *IEEE Trans. Image Process.*, vol. 8, no. 11, pp. 1586–1592, Nov. 1999.
[29] J. Bigun, T. Bigun, and K. Nilsson, "Recognition by symmetry derivatives and the generalized structure tensor," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 12, pp. 1590–1605, Dec. 2004.
[30] J.-Y. Wang and F. S. Cohen, "Part II: 3-D object recognition and shape estimation from image contours using B-splines, shape invariant matching, and neural network," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 1, pp. 13–23, Jan. 1994.
[31] J. Porrill and S. Pollard, "Curve matching and stereo calibration," *Image Vis. Comput.*, vol. 9, no. 1, pp. 45–50, Feb. 1991.
[32] L. D. Cohen, "Auxiliary variables and two-step iterative algorithms in computer vision problems," *J. Math. Imag. Vis.*, vol. 6, no. 1, pp. 59–83, 1996.
[33] C. Orrite and J. E. Herrero, "Shape matching of partially occluded curves invariant under projective transformation," *Comput. Vis. Image Understanding*, vol. 93, no. 1, pp. 34–64, 2004.
[34] A. Ross, S. C. Dass, and A. K. Jain, "Fingerprint warping using ridge curve correspondences," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 1, pp. 19–30, Jan. 2006.
[35] C. Gope, N. Kehtarnavaz, G. Hillman, and B. Würsig, "An affine invariant curve matching method for photo-identification of marine mammals," *Pattern Recognit.*, vol. 38, no. 1, pp. 125–132, 2005.
[36] T. B. Sebastian and B. B. Kimia, "Curves vs. skeletons in object recognition," *Signal Process.*, vol. 85, no. 2, pp. 247–263, 2005.
[37] B. J. Super, "Fast correspondence-based system for shape retrieval," *Pattern Recognit. Lett.*, vol. 25, no. 2, pp. 217–225, 2004.
[38] Y.-H. Gu and T. Tjahjadi, "Coarse-to-fine planar object identification using invariant curve features and B-spline modeling," *Pattern Recognit.*, vol. 33, no. 9, pp. 1411–1422, 2000.
[39] Z. Huang and F. S. Cohen, "Affine-invariant B-spline moments for curve matching," *IEEE Trans. Image Process.*, vol. 5, no. 10, pp. 1473–1480, Oct. 1996.
[40] Y. Avrithis, Y. Xirouhakis, and S. Kollias, "Affine-invariant curve normalization for object shape representation, classification, and retrieval," *Mach. Vis. Appl.*, vol. 13, no. 2, pp. 80–94, 2001.
[41] B. Kamgar-Parsi and B. Kamgar-Parsi, "Matching sets of 3D line segments with application to polygonal arc matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 10, pp. 1090–1099, Oct. 1997.
[42] Y. Shan and Z. Zhang, "New measurements and corner-guidance for curve matching with probabilistic relaxation," *Int. J. Comput. Vis.*, vol. 46, no. 2, pp. 157–171, 2002.
[43] J. S. Marques, "A fuzzy algorithm for curve and surface alignment," *Pattern Recognit. Lett.*, vol. 19, no. 9, pp. 797–803, 1998.
[44] M. Frenkel and R. Basri, "Curve matching using the fast marching method," in *Energy Minimization Methods in Computer Vision and Pattern Recognition*, A. Rangarajan, M. Figueiredo, and J. Zerubia, Eds. Berlin, Germany: Springer, 2003, pp. 35–51.
[45] S. Wu and L. Hong, "Hand tracking in a natural conversational environment by the interacting multiple model and probabilistic data association (IMM-PDA) algorithm," *Pattern Recognit.*, vol. 38, no. 11, pp. 2143–2158, 2005.
[46] R. Rosales and S. Sclaroff, "A framework for heading-guided recognition of human activity," *Comput. Vis. Image Understanding*, vol. 91, no. 3, pp. 335–367, 2003.
[47] P. S. Maybeck, *Stochastic Models, Estimation and Control*. New York, NY, USA: Academic Press, 1982.
[48] Y. Bar-Shalom, *Estimation With Applications to Tracking and Navigation*. New York, NY, USA: Wiley, 2001.
[49] H. J. Wolfson, "On curve matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 5, pp. 483–489, May 1990.
[50] J. Gaspar, N. Winters, and J. Santos-Victor, "Vision-based navigation and environmental representations with an omnidirectional camera," *IEEE Trans. Robot. Autom.*, vol. 16, no. 6, pp. 890–898, Dec. 2000.
[51] D. Scaramuzza, A. Martinelli, and R. Siegwart, "A toolbox for easily calibrating omnidirectional cameras," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Beijing, China, 2006, pp. 5695–5701.

Emrah Benli (S'15) received the B.Sc. degree in electronics and telecommunication engineering from Kocaeli University, Kocaeli, Turkey, in 2009 and the M.Sc. degree in electrical and computer engineering from Clemson University, Clemson, SC, USA, in 2013. He is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, Virginia Commonwealth University, Richmond, VA, USA.

His current research interests include intelligent systems, computer vision, image processing, pattern recognition, robotic system design and control, and human–robot interaction.

Yuichi Motai (S'00–M'03–SM'12) received the B.Eng. degree in instrumentation engineering from Keio University, Tokyo, Japan, in 1991, the M.Eng. degree in applied systems science from Kyoto University, Kyoto, Japan, in 1993, and the Ph.D. degree in electrical and computer engineering from Purdue University, West Lafayette, IN, USA, in 2002.

He is currently an Associate Professor of electrical and computer engineering with Virginia Commonwealth University, Richmond, VA, USA. His current research interests include the sensory intelligence; particularly in medical imaging, pattern recognition, computer vision, and sensory-based robotics.

John Rogers (S'09–M'14–SM'17) received the B.S. and M.S. degrees in electrical and computer engineering from Carnegie Mellon University, Pittsburgh, PA, USA, in 2002, the M.S. degree in computer science from Stanford University, Stanford, CA, USA, in 2006, and the Ph.D. degree from the Georgia Institute of Technology, Atlanta, GA, USA, in 2012.

He is a Research Scientist, specializing in autonomous mobile robotics, with the U.S. Army Research Laboratory's Computational and Information Sciences Directorate, Adelphi, MD, USA. His current research interests include automatic exploration and mapping of large-scale indoor and outdoor environments, place recognition in austere locations, and semantic scene understanding and probabilistic reasoning for autonomous mobile robots.